JGLUE: 日本語言語理解ベンチマーク

栗原健太郎 [1]　河原大輔 [1]　柴田知秀 [2]
[1] 早稲田大学理工学術院　[2] ヤフー株式会社
[1]{kkurihara@akane., dkw@}waseda.jp
[2]tomshiba@yahoo-corp.jp

概要

計算機による言語理解を目指して、複数種類の言語理解タスクを包括的に解くことによってモデルを評価、分析することが盛んに行われている。英語の言語理解ベンチマークである GLUE [1] を先駆けとして、中国語版の CLUE [2] やフランス語版の FLUE [3] などの英語以外の言語でもベンチマークの公開が進んでいる一方で、日本語においてはこのようなベンチマークは存在しない。本研究では、一般的な日本語理解能力を測ることを目的とし、翻訳を介することなく、日本語で一から言語理解ベンチマーク JGLUE を構築する。JGLUE によって日本語自然言語処理における言語理解研究の促進を図る。

1　はじめに

高性能な言語理解モデルの研究開発が目下、活発に行われている。言語理解モデルの改良には、言語理解の能力を様々な観点から評価し分析するためのベンチマーク (データセット群) が必要である。英語においては、GLUE (General Language Understanding Evaluation) [1] が構築、公開されている。GLUE である程度の高スコアを達成できる言語理解モデルが開発されると、より難易度の高いベンチマークとして SuperGLUE [4] などが構築され、ベンチマーク構築と言語理解モデル開発の好循環が生まれている。

このような英語における言語理解研究活性化の潮流に乗じて、中国語版の CLUE [2]、フランス語版の FLUE [3]、韓国語版の KLUE [5] などの各言語におけるベンチマークの構築や、XGLUE [6] などの多言語ベンチマークの構築が進んでいる。

日本語においては、現在のところ、GLUE のようなベンチマークは存在せず、多言語ベンチマークにおいても日本語データは少数しか含まれていない。個々の日本語データセットは構築されているが、翻訳を介した構築手法を用いているか、特定のドメインを対象にしているものが多い。例えば、JSNLI [7] や JSICK [8] などは、英語のデータセットからの機械翻訳あるいは人手翻訳によって構築されている。いずれの翻訳手法でも翻訳文の不自然さや日本との文化差が大きな問題となる。また、JRTE コーパス [9] や運転ドメイン QA データセット [10] は、ホテルレビューや運転行動を対象としたデータセットであり、一般的なドメインの言語理解能力を測るのには向かない。

表 1　JGLUE の構成

タスク	データセット	train	dev	test
文章分類	MARC-ja	187,528	5,654	5,639
	JCoLA	—	—	—
文ペア分類	JSTS	12,463	1,457	1,589
	JNLI	20,117	2,434	2,508
QA	JSQuAD	63,870	4,475	4,470
	JCommonsenseQA	9,012	1,126	1,126

本研究では、一般的な日本語理解能力を測ることを目的とし、翻訳を介することなく、日本語で一から言語理解ベンチマーク JGLUE を構築する。JGLUE は、文章分類、文ペア分類、QA の 3 種類のタスクから構成し、GLUE および SuperGLUE のタスクを幅広くカバーするように設計した (表 1)。本ベンチマークによって日本語における言語理解研究が活性化することを期待する。

2　JGLUE の構築

JGLUE は、表 1 のとおり、文章分類、文ペア分類、QA のタスクから構成する。以下では、各タスクのデータセットの構築方法について説明する。各データセットの構築はクラウドソーシング[1)]を用いて行う。ただし、文章分類タスクの一つである JCoLA (日本語容認性判断データセット) [11] は東京大学大関研究室から提供される予定であり、本稿では説明しない。

1)　Yahoo!クラウドソーシング (https://crowdsourcing.yahoo.co.jp/) を用いた。

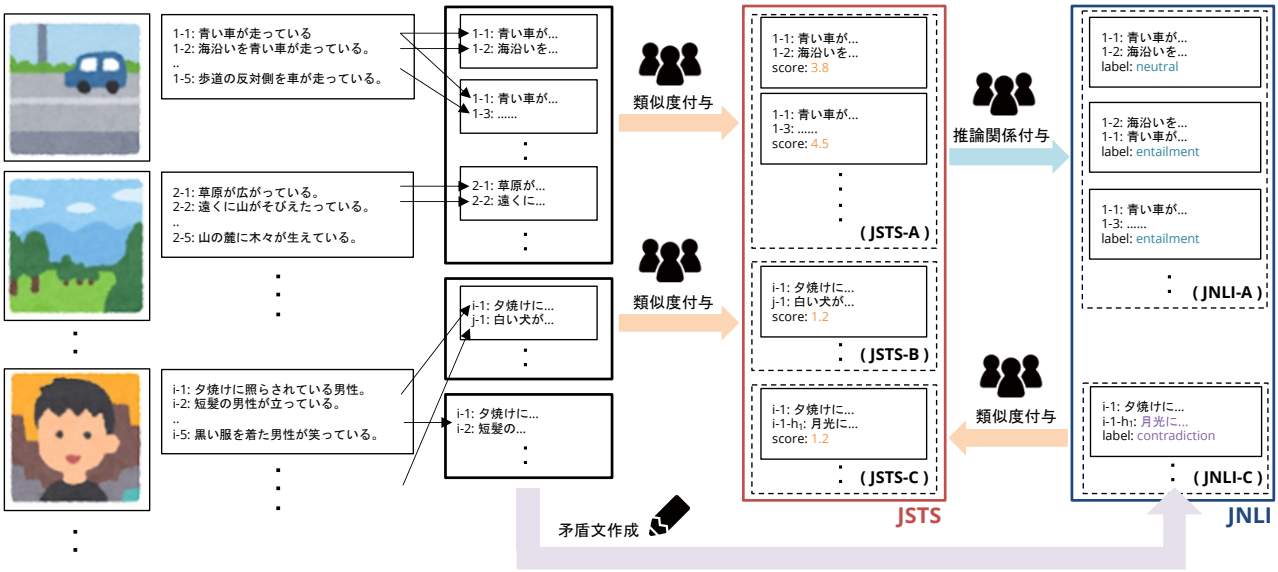


図 1　JSTS・JNLI の構築フロー (画像の出典: いらすとや (https://www.irasutoya.com/), ONWA イラスト (https://onwa-illust.com/))

表 2　JSTS・JNLI の例: 由来について、A, B, C はそれぞれ (JSTS-A, JNLI-A), (JSTS-B), (JSTS-C, JNLI-C) に含まれる文ペアであることを示す。

文 1 / 前提文	文 2 / 仮説文	類似度	推論関係	由来
街中の道路を大きなバスが走っています。	道路を大きなバスが走っています。	4.4	entailment	A
テーブルに料理がならべられています。	テーブルに食べかけの料理があります。	3.0	neutral	A
野球選手がバットをスイングしています。	野球選手がキャッチボールをしています。	2.0	contradiction	C
フリスビーをくわえた犬がいます。	建物の前にバスが一台停車しています。	0.0	—	B

2.1　MARC-ja

文章分類タスクの一つとして、多言語商品レビューコーパス MARC (Multilingual Amazon Reviews Corpus) [12] を使用してデータセットを構築する。

MARC は、通信販売サイト「アマゾン」における商品レビューとそれに対する 1〜5 の 5 段階の評価をまとめたコーパスであり、英語や日本語などの複数言語で公開されている。JGLUE においては、MARC の日本語部分を使用し、容易に判断可能な問題にするために、5 段階の評価のうち 3 を除く 4 つの評価について、1, 2 を “negative”、4, 5 を “positive” に変換して用いた 2 値分類タスクとする。

MARC における問題点として、positive な内容のレビューに対して低評価がついている場合など、内容と評価が乖離したデータが含まれている場合があることが挙げられる。これらのデータには、レビュー内容と明らかに異なるラベルが割り振られるため、データセットの品質を低下させる。

評価に用いる dev/test データについては高品質なものにするために、positive, negative 判定タスクをクラウドソーシングで実施する。多数決により正解ラベルを振り直すとともに、票が割れる事例については除去する。

評価指標には精度 (acc) を用いる。

2.2　JSTS・JNLI

文ペア分類タスクについては、意味的類似度計算 (Semantic Textual Similarity, STS) データセット JSTS および自然言語推論 (Natural Language Inference, NLI) データセット JNLI を構築する。

STS は文ペアの意味的な類似度を推定するタスクである。正解の類似度は、0 (意味が完全に異なる) 〜 5 (意味が等価) の間の値として付与されるのが一般的である。NLI は、前提文と仮説文の文ペアが与えられたときに、前提文が仮説文に対してもつ推論関係を認識するタスクである。推論関係としては「含意 (entailment)」「矛盾 (contradiction)」「中立 (neutral)」の 3 値で定義されるのが一般的である。

STS, NLI タスクは GLUE において、それぞれ STS-B [13], MultiNLI [14] データセットが含まれている。日本語では、NLI データセット SNLI (Stanford NLI) [15] を機械翻訳した JSNLI [7]、STS/NLI データセット SICK [16] を人手翻訳した JSICK [8] がある。

しかし、1 節で述べたように、これらには翻訳に由来する問題があるため、本研究では日本語で一から構築する。

JSTS と JNLI の文ペアは基本的に、YJ Captions Dataset [17] (以下、YJ Captions と呼ぶ) から抽出する[2]。SICK や JSICK と同様に、JSTS と JNLI を構成する文ペアの大部分は重複しており、同じ文ペアに対する類似度と推論関係の関係を分析することができる。JSTS における類似度は STS-B と同様に 0〜5 の実数値とし、JNLI における推論関係は MultiNLI などと同様に上記の 3 値とする。

JSTS と JNLI の構築フローを図 1 に示す。基本的には、YJ Captions のある画像に対する 2 つのキャプションを文ペアとし、クラウドソーシングによって類似度 (図 1 の JSTS-A) および entailment と neutral の NLI 事例 (図 1 の JNLI-A) を得る。しかし、ある画像に対する 2 つのキャプションからは類似度の低い文ペアと contradiction 関係をもつ文ペアを収集することが難しいという問題がある。そこで、異なる画像に対するキャプションから類似度の低い文ペアを収集し (図 1 の JSTS-B)、contradiction 関係については、あるキャプションに対して矛盾する文をワーカに作文してもらうことによって収集する (図 1 の JNLI-C)。作文で収集した contradiction 関係の事例についてもワーカに類似度を付与してもらう (図 1 の JSTS-C)。

以上の手続きによって獲得した JSTS-A, B, C の 3 つで JSTS、また JNLI-A, C の 2 つで JNLI を構築した。表 2 に JSTS と JNLI の例を示す。

JSTS の評価指標には、STS-B と同様に Pearson および Spearman 相関係数を用いる。JNLI の評価指標には、MultiNLI と同様に精度を用いる。

2.3 JSQuAD

QA タスクとして機械読解タスクの一つである SQuAD の日本語版と、次節で説明する CommonsenseQA の日本語版を構築する。

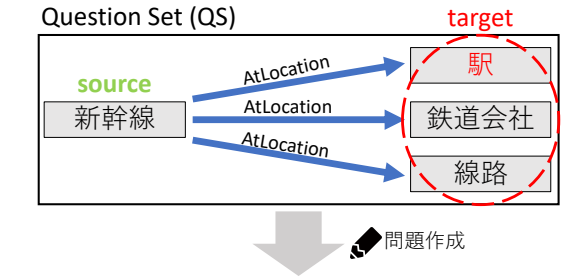
機械読解タスクは文書を読み、それに関する質問に対して答えるというタスクである。多くの機械読解評価セットは英語で構築されているが、その他の言語での機械読解評価セット ([20] など) や多言語の評価セット ([21] など) が構築されている。

2) YJ Captions は MS COCO Caption Dataset [18] の日本語版で、MS COCO [19] に含まれる画像に日本語のキャプションを 5 文ずつクラウドソーシングで付与することによって構築されている。

[タイトル] 東海道新幹線
1987 年（昭和 62 年）4 月 1 日の国鉄分割民営化により、JR 東海が運営を継承した。西日本旅客鉄道（JR 西日本）が継承した山陽新幹線とは相互乗り入れが行われており、...。2020 年（令和 2 年）3 月現在、東京駅 - 新大阪駅間の所要時間は最速 2 時間 21 分、最高速度 285km/h で運行されている。

質問: 2020 年、東京〜新大阪間の最速の所要時間は
答え: 2 時間 21 分
質問: 東海道新幹線と相互乗り入れがされている路線はどこか？
答え: 山陽新幹線

図 2 JSQuAD の例



問題: 電車に人が乗り降りする場所を何という？
選択肢: 駅, 鉄道会社, 線路, 空港, 港

誤り選択肢追加

図 3 JCommonsenseQA の構築フロー

日本語ではクイズを対象にした機械読解評価セット [22] や運転ドメインの評価セット [10] が構築されているが、一般ドメインのものはない。そこで、Wikipedia を用いて一般ドメインの評価セットを構築する。構築は基本的に SQuAD 1.1 [23] にならう。

まず、Nayuki[3] を用いて、高品質な記事 10,000 記事を選出し、そこからランダムに 822 記事を選ぶ[4]。次に、記事を段落に分割し、ワーカに各段落を提示し、段落を理解できれば答えられるような質問とその答えを書いてもらう。図 2 に JSQuAD の例を示す。

評価指標は SQuAD にならい、Exact match (EM) と F1[5] を用いる。

2.4 JCommonsenseQA

JCommonsenseQA は、CommonsenseQA [24] の日本語版データセットであり、常識推論能力を評価するための 5 択 QA 問題で構成する。JCommonsenseQA は、CommonsenseQA と同様に、知識ベース Concept-

3) Nayuki は Wikipedia 内のハイパーリンクに基づき、記事の品質を推定するものである。https://www.nayuki.io/
4) 例えば「熊本県」「フランス料理」などの記事がある。
5) 英語では F1 は単語単位で計算されているが、日本語で形態素単位で計算すると採用する形態素解析器によって値が異なってしまうので文字単位で計算する。

表 3　JGLUE による各種モデルの評価結果

モデル	MARC-ja acc		JSTS Pearson/Spearman		JNLI acc		JSQuAD EM/F1		JCommonsenseQA acc	
	dev	test	dev	test	dev	test	dev	test	dev	test
Human	0.989	0.990	0.899/0.861	0.909/0.872	0.925	0.917	0.870/0.943	0.874/0.947	0.988	0.988
東北大 $\text{BERT}_{\text{BASE}}$	0.958	0.957	0.908/0.868	0.908/0.865	0.901	0.882	0.871/0.940	0.874/0.945	0.806	0.798
東北大 $\text{BERT}_{\text{BASE}}$(文字)	0.956	0.957	0.893/0.851	0.901/0.858	0.875	0.868	0.864/0.934	0.868/0.938	0.739	0.732
東北大 $\text{BERT}_{\text{LARGE}}$	0.955	0.961	**0.915**/0.875	0.912/0.869	0.910	0.884	0.881/**0.947**	0.881/**0.950**	0.816	0.810
NICT $\text{BERT}_{\text{BASE}}$	0.958	0.960	0.911/0.874	0.910/0.866	0.904	0.889	**0.893/0.947**	**0.902/0.950**	0.821	0.809
早稲田大 $\text{RoBERTa}_{\text{BASE}}$	0.962	0.962	0.911/0.870	0.910/0.867	0.903	0.887	0.863/0.926	0.862/0.922	0.838	**0.852**
$\text{XLM-RoBERTa}_{\text{BASE}}$	0.961	0.962	0.879/0.832	0.888/0.836	0.894	0.865	-	-	0.701	0.721
$\text{XLM-RoBERTa}_{\text{LARGE}}$	**0.964**	**0.965**	0.914/**0.878**	**0.918/0.879**	**0.926**	**0.906**	-	-	**0.839**	0.830

Net [25] をシードとし、クラウドソーシングを用いて構築する。ConceptNet は、2 つの概念 (concept) と、その間の関係 (relation) を表す 3 つ組からなる多言語知識ベースである。3 つ組は方向性を持ち、例えば (新幹線, AtLocation, 駅) のように、(source concept, relation, target concept) として表される。

JCommonsenseQA の構築フローを図 3 に示す。まず、source と、それに対して同じ relation を持つ target3 つからなる集合 (Question Set, QS) を ConceptNet から収集する。次に、各 QS に対して、1 つの target のみが解答となる問題文の作成と、2 つの誤り選択肢の追加をクラウドソーシングで行う。

評価指標には CommonsenseQA と同様に精度を用いる。

3　JGLUE を用いたモデル評価

3.1　実験設定

実験に用いた事前学習モデルを付録の C に示す。ファインチューニングはタスク/データセットに応じて以下のように行った[6]。

- 文章分類タスクと文ペア分類タスク: [CLS] トークンに対する分類/回帰問題を解く。
- JSQuAD: 各トークンに対して答えのスパンの開始/終了となるかどうかの分類問題を解く[7]。
- JCommonsenseQA: 質問と各選択肢を連結し、多肢選択式問題を解く。

dev セットで最適なハイパーパラメータを探索し、最適なハイパーパラメータで test セットで性能を算出した。実験に用いたハイパーパラメータを付録の D に示す。

3.2　結果

表 3 に各種モデルのスコアならびにヒューマンスコアを示す。ヒューマンスコアはデータ構築と同様、クラウドソーシングを用いて算出した。モデルの性能の比較は以下のようにまとめることができる。

- 全般的には $\text{XLM-RoBERTa}_{\text{LARGE}}$ が最もよい。これは LARGE サイズであることと、事前学習のテキストとして Wikipedia よりも大規模な Common Crawl を使っていることが考えられる。
- 基本単位について、サブワード単位と文字単位を比較すると一貫してサブワード単位の方が精度が高い。
- JCommonsenseQA は Wikipedia には記載されにくい常識的な知識を要求することから、Common Crawl を用いたモデルの精度が高い。
- JCommonsenseQA 以外についてはベストなモデルは人間のスコアと同等または超えている。

公開されているモデルは事前学習のテキスト、学習時間、トークナイザなどの条件が異なるため、精度向上にどの要素が効いているか、厳密には判断できない。今後、他の条件を揃えて例えばトークナイザのみを変えた場合の比較などを行う予定である。

4　おわりに

本論文では日本語における言語理解ベンチマーク JGLUE の構築について述べた。JGLUE v1 は 2022 年 3 月に公開する予定である。JGLUE を用いて、事前学習モデルの包括的な評価や、より難しい評価データの構築が進むことを期待している。今後は GLGE [26] のような生成系タスクや FLEX [27] のような Few-shot タスクのデータセットなどを構築する予定である。

6)　Hugging Face 社の transformers を用いた。https://github.com/huggingface/transformers

7)　$\text{XLM-RoBERTa}_{\text{BASE}}$ と $\text{XLM-RoBERTa}_{\text{LARGE}}$ はトークナイザとして Unigram 言語モデルを利用しており、トークンの区切りと答えのスパンの開始/終了が一致しないことが多く、性能が出ないため対象から除いた。

謝辞

本研究はヤフー株式会社と早稲田大学の共同研究により実施した。

参考文献

[1] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In **EMNLP2018 Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP**, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[2] Liang Xu, Hai Hu, Xuanwei Zhang, Lu Li, Chenjie Cao, Yudong Li, Yechen Xu, Kai Sun, Dian Yu, Cong Yu, Yin Tian, Qianqian Dong, Weitang Liu, Bo Shi, Yiming Cui, Junyi Li, Jun Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Zhe Zhao, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhengliang Yang, Kyle Richardson, and Zhenzhong Lan. CLUE: A Chinese language understanding evaluation benchmark. In **COLING2020**, pp. 4762–4772, Barcelona, Spain (Online), December 2020.

[3] Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. FlauBERT: Unsupervised language model pre-training for French. In **LREC2020**, pp. 2479–2490, Marseille, France, May 2020. European Language Resources Association.

[4] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, **Advances in NeurIPS**, Vol. 32. Curran Associates, Inc., 2019.

[5] Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Ji Yoon Han, Jangwon Park, Chisung Song, Junseong Kim, Youngsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice Oh, Jung-Woo Ha, and Kyunghyun Cho. KLUE: Korean language understanding evaluation. In **Thirty-fifth Conference on NeurIPS Datasets and Benchmarks Track (Round 2)**, 2021.

[6] Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. Xglue: A new benchmark dataset for cross-lingual pre-training, understanding and generation, 2020.

[7] 吉越卓見, 河原大輔, 黒橋禎夫. 機械翻訳を用いた自然言語推論データセットの多言語化. 情報処理学会研究報告 第244回自然言語処理研究会, pp. 1–8, 2020.

[8] 谷中瞳, 峯島宏次. JSICK: 日本語構成的推論・類似度データセットの構築. 人工知能学会全国大会論文集, Vol. JSAI2021, pp. 4J3GS6f02–4J3GS6f02, 2021.

[9] Yuta Hayashibe. Japanese realistic textual entailment corpus. In **LREC2020**, pp. 6827–6834, Marseille, France, May 2020. European Language Resources Association.

[10] Norio Takahashi, Tomohide Shibata, Daisuke Kawahara, and Sadao Kurohashi. Machine comprehension improves domain-specific Japanese predicate-argument structure analysis. In **MRQA2019**, pp. 98–104, Hong Kong, China, November 2019.

[11] 染谷大河, 大関洋平. 日本語版 CoLA の構築. 言語処理学会第 28 回年次大会, 2022.

[12] Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. The multilingual Amazon reviews corpus. In **EMNLP2020**, pp. 4563–4568, Online, November 2020.

[13] Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. **In SemEval-2017**, 2017.

[14] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In **NAACL2018**, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[15] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In **EMNLP2015**, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[16] Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. A SICK cure for the evaluation of compositional distributional semantic models. In **LREC2014**, pp. 216–223, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA).

[17] Takashi Miyazaki and Nobuyuki Shimizu. Cross-lingual image caption generation. In **ACL2016**, pp. 1780–1790, 2016.

[18] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. Microsoft COCO captions: Data collection and evaluation server, 2015.

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common objects in context, 2015.

[20] Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. A span-extraction dataset for Chinese machine reading comprehension. In **EMNLP-IJCNLP2019**, Hong Kong, China, November 2019.

[21] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. MLQA: Evaluating cross-lingual extractive question answering. In **ACL2020**, Online, July 2020.

[22] 鈴木正敏, 松田耕史, 岡崎直観, 乾健太郎. 読解による解答可能性を付与した質問応答データセットの構築. 言語処理学会第 24 回年次大会, 2018.

[23] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In **EMNLP2016**, pp. 2383–2392, Austin, Texas, November 2016.

[24] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In **NAACL2019**, Minneapolis, Minnesota, June 2019.

[25] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. **AAAI2017**, Vol. 31, No. 1, Feb. 2017.

[26] Dayiheng Liu, Yu Yan, Yeyun Gong, Weizhen Qi, Hang Zhang, Jian Jiao, Weizhu Chen, Jie Fu, Linjun Shou, Ming Gong, Pengcheng Wang, Jiusheng Chen, Daxin Jiang, Jiancheng Lv, Ruofei Zhang, Winnie Wu, Ming Zhou, and Nan Duan. GLGE: A new general language generation evaluation benchmark. In **ACL-IJCNLP 2021 Findings**, Online, August 2021.

[27] Jonathan Bragg, Arman Cohan, Kyle Lo, and Iz Beltagy. FLEX: Unifying evaluation for few-shot NLP. In **NeurIPS2021**, 2021.

表 4　事前学習モデルの詳細
(モデル名の括弧内は huggingface models での名称を示す。東北大 $\text{BERT}_{\text{BASE}}$ と $\text{XLM-RoBERTa}_{\text{BASE}}$ はそれぞれに対応する LARGE モデルも利用する。事前学習テキストの CC は Common Crawl を表す。)

モデル名	基本単位	事前学習テキスト
東北大 $\text{BERT}_{\text{BASE}}$ (cl-tohoku/bert-base-japanese-v2)	サブワード (MeCab + BPE)	日本語 Wikipedia
東北大 $\text{BERT}_{\text{BASE}}$(文字) (cl-tohoku/bert-base-japanese-char-v2)	文字	日本語 Wikipedia
NICT $\text{BERT}_{\text{BASE}}$	サブワード (MeCab + BPE)	日本語 Wikipedia
早稲田大 $\text{RoBERTa}_{\text{BASE}}$ (nlp-waseda/roberta-base-japanese)	サブワード (Juman++ + Unigram LM)	日本語 Wikipedia + CC
$\text{XLM-RoBERTa}_{\text{BASE}}$ (xlm-roberta-base)	サブワード (Unigram LM)	多言語 CC

A　JCommonsenseQA の例

JCommonsenseQA の例を図 4 に示す。

問題: 会社の最高責任者を何というか？
選択肢: 教師, 部長, 社長, 部下, バイト

問題: スープを飲む時に使う道具は何？
選択肢:スプーン, メニュー, 皿, フォーク, はし

図 4　JCommonsenseQA の例

B　各データセットのラベル分布

MARC-ja, JSTS, JNLI のラベルの分布を表 5 - 7 に示す。

表 5　MARC-ja のラベル分布

ラベル	train	dev	test	Total
positive	165,477	4,832	4,895	175,204
negative	22,051	822	744	23,617
Overall	187,528	5,654	5,639	198,821

表 6　JSTS のラベル分布

類似度レンジ	train	dev	test	Total
0 - 1	2,847	353	405	3,605
1 - 2	1,753	184	160	2,097
2 - 3	2,784	308	355	3,447
3 - 4	3,720	466	488	4,674
4 - 5	1,359	146	181	1,686
Overall	12,463	1,457	1,589	15,509

表 7　JNLI のラベル分布

ラベル	train	dev	test	Total
entailment	2,876	353	367	3,596
neutral	11,193	1,347	1,365	13,905
contradiction	6,048	734	776	7,558
Overall	20,117	2,434	2,508	25,059

C　事前学習モデル

実験に用いた事前学習モデルの詳細を表 4 に示す。

D　ハイパーパラメータ

実験に用いたハイパーパラメータを表 8 に示す。

表 8　実験に用いたハイパーパラメータ
(中括弧内の数字はハイパーパラメータサーチをして最適なものを選んだことを示す)

ハイパーパラメータ名	値
learning rate	{5e-5, 3e-5, 2e-5}
epoch	{3, 4}
warmup ratio	0.1
max seq length	512 (MARC-ja), 128 (JSTS, JNLI), 384 (JSQuAD), 64 (JCommonsenseQA)

E　学習データ量を変えた時の精度変化

学習データ量を 0.75 倍、0.5 倍に変えて精度がどのように変わるかを調べた。モデルは最も精度がよかったモデルを利用した。結果を図 5 に示す。いずれのデータセットでも精度はほぼ飽和しており、構築した学習データ量が十分であることがわかる。

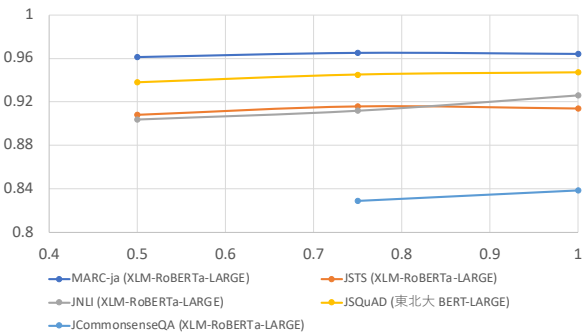


図 5　学習データ量を変えた時の精度変化 (dev セット)