



非構造化データの構造化における情報抽出

2020/12/2

情報処理学会 第246回自然言語処理研究会
株式会社リーディング・エッジ社 研究開発部 川崎拳人



概要

- ・ 弊社リーディング・エッジ社では、IT人材の派遣事業を主にやっており、求人情報を含むメール文書を扱う機会が多い
- ・ このようなメール文書はフォーマットがバラバラであり、営業戦略において、価値あるデータとして扱うことが難しい状況にある
- ・ そこで、機械翻訳手法（Transformer）によりメール文書の構造化を試みた
- ・ しかしながら、「特定個所の抽出」においては課題が見られたため、本研究では機械読解による手法（ALBERT）を導入し、情報抽出手法の検討を行った



目次

- ・ 非構造化データとは
- ・ 構造化における課題
- ・ 機械翻訳による構造化と課題
- ・ 構造化手法の検討
- ・ 関連研究
- ・ 実験
- ・ 評価指標と結果
- ・ まとめ
- ・ 考察

非構造化データとは

非構造化データは構造化データに比べ、AIやデータサイエンスの分野では生かし切れていない

	非構造化データ	構造化データ
特徴	<ul style="list-style-type: none">・フォーマットが決まっていない・データ量が多い分、ストレージコストがかかる・データの検索や更新に手間がかかる・活用するためには、データに前処理を施す必要がある	<ul style="list-style-type: none">・フォーマットが決まっている・管理しやすい・検索や更新に手間がかからない・行列で管理されているのですぐに活用できる状態にある
例	文書、画像、音声など	DB、CSVなど



構造化における課題

メールは大量にあり、そのフォーマットもバラバラ

メール1

【案件】大規模システム開発
医療向け、損保向け、システム開発系向け
各業種それぞれ大規模システム開発経験のある方を募集しております。

【場所】新宿

【期間】7月～長期

【人数】複数名

【単価】スキル見合い

【精算】140-180h

【スキル】

<必須>

- ・Java3年以上の開発経験
- ・JavaScript1年以上の開発経験

<尚可>

- ・大規模システム開発経験者
- ・ネットワーク経験者

【面談】2回

【外国籍】不可

【備考】コントロール問題ない方

メール2

案件名：Kubernetes環境の基盤構築

作業概要：

- ・インフラ・運用・自動化
- ・WEBサーバの基盤構築

必須スキル：

- ・Linuxに関して構築経験3年以上
- ・サーバ基盤の設計構築経験3年以上

歓迎スキル：

- ・Python
- ・Chef、Jenkins
- ・Zabbix
- ・Kubernetesを使用した経験

期間：即日～長期 ※3月からでも可能

場所：渋谷

募集：1～2名

単価：65万～80万（スキル見合い）

精算：あり

面談：1回（場合によっては2回）

年齢：制限なし

時間：9:00～18:00

メール3

◆作業内容：外資企業向け会計システムの
エンハンス業務

・調査、問合せ依頼に対して、システムの
調査

・既存システムの改修に伴う、設計、
オフショアへの展開、受入れ検証

・本番リリース

◆現場環境：UNIX、Oracle、COBOL、
JCL

◆必須：UNIXの操作経験

K-shellの改修経験

◆尚可：COBOL経験

◆作業場所：大井町駅徒歩5分/JR高崎駅
徒歩15分

◆期間：即日～長期

◆勤務時間 9:30～18:30

◆予算：Max50万（140-200）

◆人数：1名（40歳までを希望）

◆面談：Max2回（1回目弊社同席）

構造化における課題

構造化するためには、様々な処理が必要

メール1

項目名の判定

【案件】大規模システム開発
医療向け、損保向け、システム開発系向け
各業種それぞれ大規模システム開発経験のある方を募集しております。

【場所】新宿

【期間】7月～長期

【人数】複数名

【単価】スキル見合い

【精算】140-180h

【スキル】スキル要素の抽出

<必須>

- ・Java3年以上の開発経験
- ・JavaScript1年以上の開発経験

<尚可>

- ・大規模システム開発経験者
- ・ネットワーク経験者

【面談】2回 可/不可をフラグで表す

【外国籍】不可

【備考】コントロール問題ない方

抜き出す要素の判定

メール2

案件名：Kubernetes環境の基盤構築

作業概要：

- ・インフラ・運用・自動化
- ・WEBサーバの基盤構築

必須スキル：

- ・Linuxに関して構築経験3年以上
- ・サーバ基盤の設計構築経験3年以上

歓迎スキル：

- ・Python
- ・Chef、Jenkins
- ・Zabbix
- ・Kubernetesを使用した経験

期間：即日～長期 ※3月からでも可能

場所：渋谷

募集：1～2名

単価：65万～80万（スキル見合い）

精算：あり

面談：1回（場合によっては2回）

年齢：制限なし

時間：9:00～18:00

工程の抽出

メール3

◆作業内容：外資企業向け会計システムの
エンハンス業務

・調査、問合せ依頼に対して、システムの
調査

・既存システムとの連携、データ
移行、設計、
オフショア開発、テスト、
工程の抽出
検証

・本番リリース

◆現場環境：UNIX、Oracle、COBOL、
JCL

◆必須：UNIXの操作経験

K-shellの改修経験 最寄り駅の抽出

◆尚可：COBOL経験

◆作業場所：大井町駅徒歩5分/JR高崎駅
徒歩15分

◆期間：即日～長期

◆勤務時間 9:30～18:30

◆予算：Max50万（140-200）

◆人数：1名（40歳までを希望）

◆面談：Max2回（1回自弊社同席）

項目と対になっていない要素の抽出

構造化における課題

構造化するためには、様々な処理が必要

メール1

【案件】大規模システム開発
医療向け、損保向け、システム開発系向け
各業種それぞれ大規模システム開発経験のある方を募集しております。

【場所】新宿
【期間】7月～長期
【人数】複数名
【単価】スキル見合い
【精算】140-180h
【スキル】

<必須>

- ・Java3年以上の開発経験
- ・JavaScript1年以上の開発経験

<尚可>

- ・大規模システム開発経験者
- ・ネットワーク経験者

【面談】2回
【外国籍】不可
【備考】コントロール問題ない方

定量化する

スキルを抽出して
カテゴリ別に
振り分ける

可/不可をフラグで表す

メール2

案件名：Kubernetes環境の基盤構築
作業概要：

- ・インフラ・運用・自動化
- ・WEBサーバの基盤構築

必須スキル：

- ・Linuxに関して構築経験3年以上
- ・サーバ基盤の設計構築経験3年以上

歓迎スキル：

- ・Python
- ・Chef、Jenkins
- ・Zabbix
- ・Kubernetesを使用した経験

期間：即日～長期 ※3月からでも可能
場所：渋谷
募集：1～2名
単価：65万～80万（スキル見合い）
精算：あり
面談：1回（場合によっては2回）
年齢：制限なし
時間：9:00～1

様々な表現を統一する

数値に変換する

定量化する

メール3

◆作業内容：外資企業向け会計システムの
エンハンス業務

- ・調査、問合せ依頼に対して、システムの調査
- ・既存システムの改修に伴う、設計、オフショアへの展開、受入れ検証
- ・本番リリース

◆現場環境：UNIX、Oracle、COBOL、
K-shellの改修経験

◆尚可：COBOL経験
◆作業場所：大井町駅徒歩5分/JR高崎駅徒歩15分
◆期間：即日～長期
◆勤務時間 9：30～18：30
◆予算：Max50万（140-200）
◆人数：1名（40歳までを希望）
◆面談：Max2回（1回目弊社同席）

大文字小文字を揃える



構造化タスク

課題を構造化タスクに集約し、それを実現することによって課題を解決する

1. 特定情報の抽出

抽出すべき項目名とその内容の関連性を判断して、特定個所を抽出する

2. 表記変換

同一の意味を持つ様々な表記を統一する

3. 表現の統一

抽象的な表現を定量的な表現に統一する

機械翻訳による構造化

Transformerによりメール文書からJSONに変換する

表1 メール文

【案件名】 AI Webアプリ開発案件
【内容】 チャットボットを用いたWebアプリの設計・開発に携わっていただきます
【勤務地】 茅場町（東京メトロ東西線・日比谷線）
【期間】 10月～12月 ※延長の可能性あり
【必須スキル】 Python, C#を使用した開発経験 自然言語処理に関する知識
【推奨スキル】 javascript, django Webアプリ開発の経験
【単価】 50～60万前後
【年齢】 40歳はじめまでを希望
【人数】 1名
【面談回数】 2回（弊社同席）
【その他】 勤怠，コミュニケーションに問題のない方

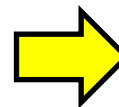


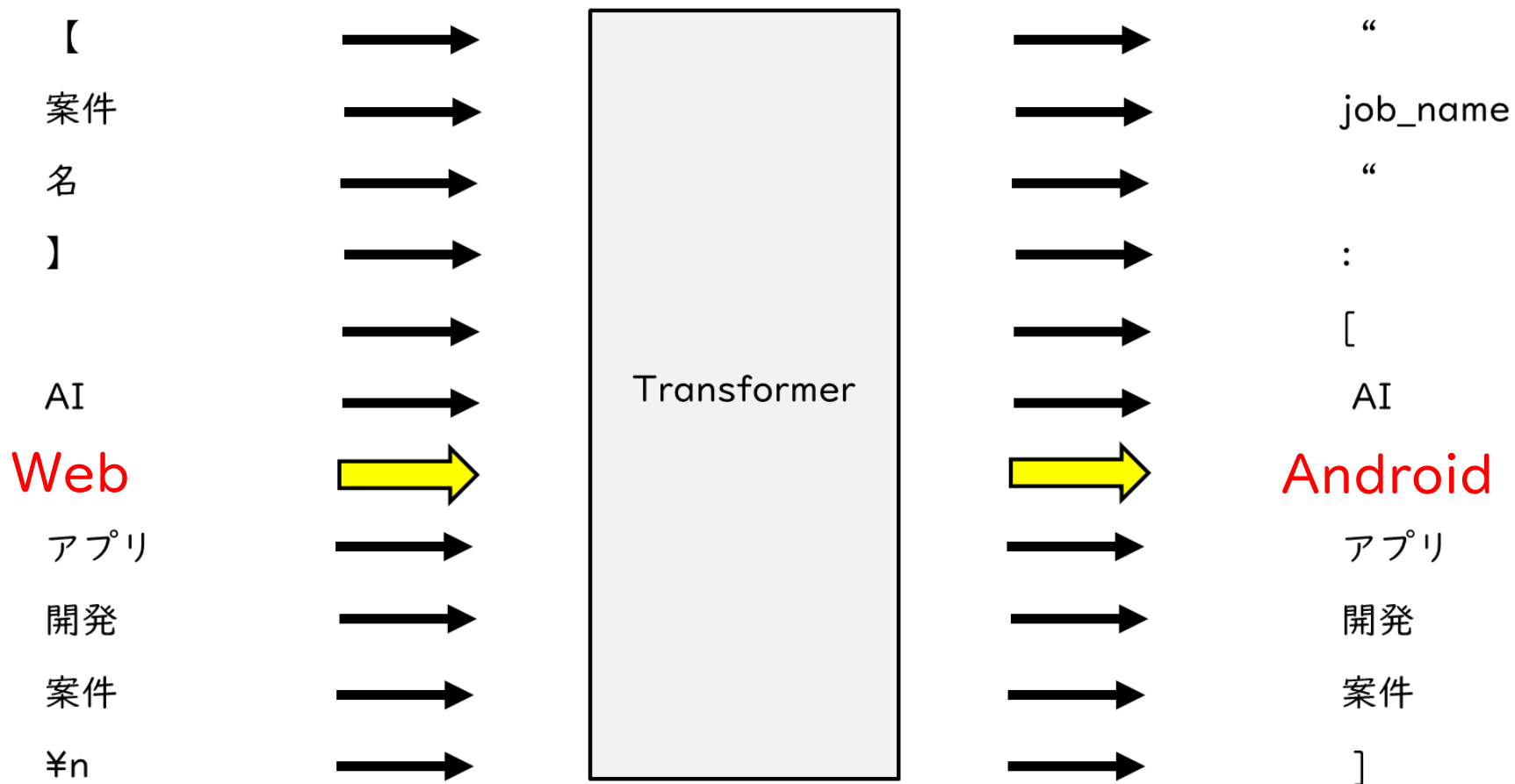
表2 JSON

```
{"job_name": ["AI Webアプリ開発案件"],  
"contents": ["チャットボットを用いたWebアプリの設計・開発に携わっていただきます"],  
"sites_station": ["茅場町"],  
"durings": ["10月～12月"],  
"price_min": [500000],  
"price_max": [600000],  
"age_min": [-1],  
"age_max": [43],  
"skill_required_os": [],  
"skill_required_lang": ["Python", "C#"],  
...,  
"skill_recommended_web": ["JavaScript", "Django"],  
"required_numbers": ["1名"],  
"counts_for_interview": ["2回（弊社同席）"],  
"etc": ["勤怠，コミュニケーションに問題のない方"]}
```

課題

翻訳の副作用

同一単語でも英語から日本語に翻訳するような作業をしているため、時々翻訳ミスが発生する



構造化手法の検討

機械読解により、Start/Endでの抽出を行うことで、構造化の精度を高める

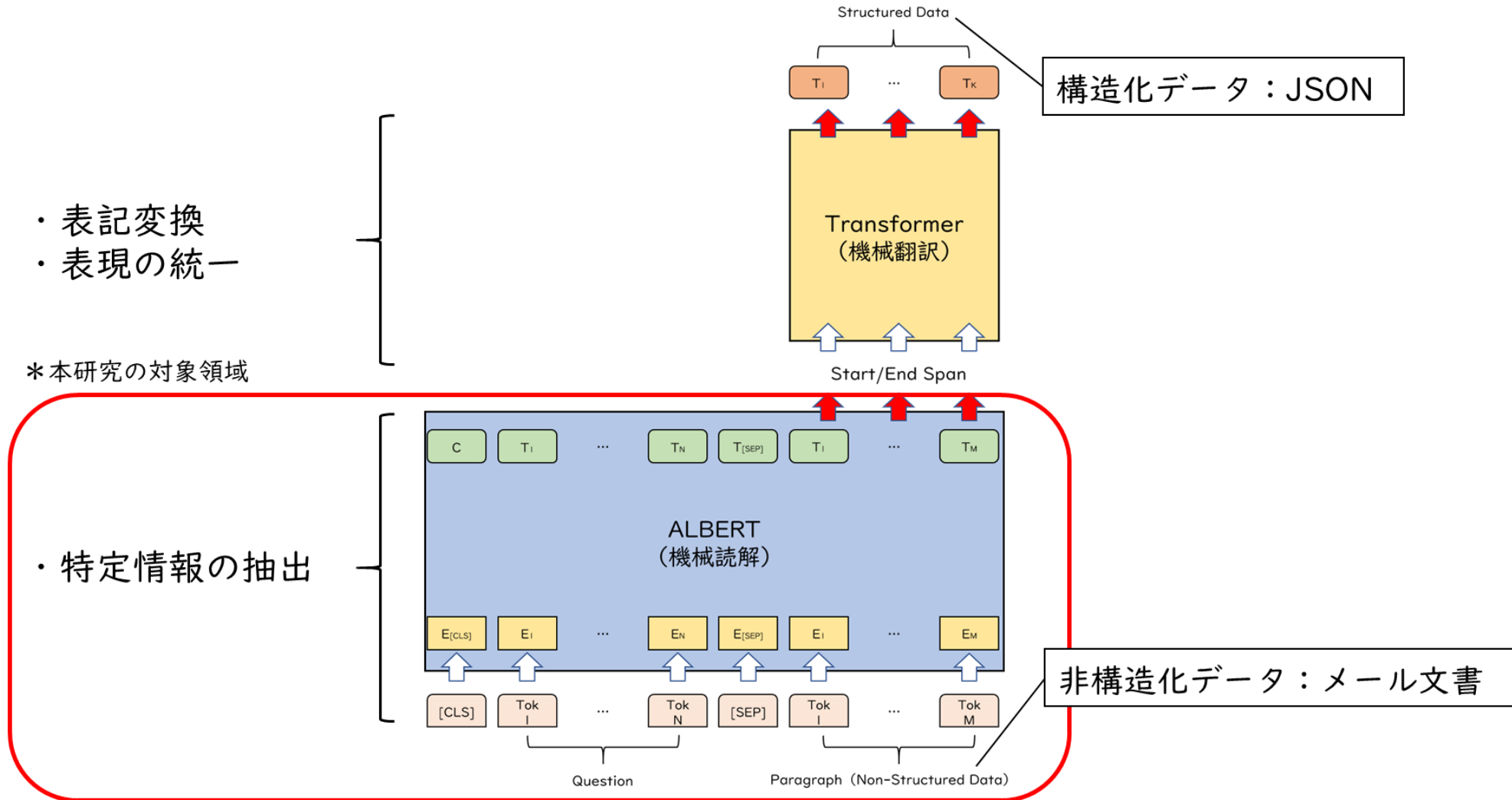


図 1 構造化イメージ

機械読解

- ・ 機械読解とは、ある文章とその文章にかかわる質問が与えられた時に、機械が文章の内容を読み解き、質問の回答にあたる部分を抽出するタスク
- ・ 質問応答のデータセットとしては、スタンフォード大学がSQuAD2.0*を公開しており、機械読解のベンチマークとして一般的に使用されている

Article: Endangered Species Act

Paragraph: “... Other legislation followed, including the Migratory Bird Conservation Act of 1929, a 1937 treaty prohibiting the hunting of right and gray whales, and the **Bald Eagle Protection Act of 1940**. These **later laws** had a low cost to society—the species were relatively rare—and little **opposition** was raised.”

Question 1: “Which laws faced significant **opposition**?”

Plausible Answer: **later laws**

Question 2: “What was the name of the 1937 treaty?”

Plausible Answer: **Bald Eagle Protection Act**

図 2 SQuAD2.0

* “SQuAD2.0 The Stanford Question Answering Dataset “. <https://rajpurkar.github.io/SQuAD-explorer/>

ALBERT

- ALBERT*はBERTを計量化したモデルであり、BERTと同等の精度を保ちつつ学習の高速化に成功している

	Model	Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3	4.7x
	large	334M	92.2/85.5	85.0/82.2	86.6	93.0	73.9	85.2	1.0
ALBERT	base	12M	89.3/82.3	80.0/77.1	81.6	90.3	64.0	80.1	5.6x
	large	18M	90.6/83.9	82.3/79.4	83.5	91.7	68.5	82.4	1.7x
	xlarge	60M	92.5/86.1	86.1/83.1	86.4	92.4	74.8	85.5	0.6x
	xxlarge	235M	94.1/88.3	88.1/85.1	88.0	95.2	82.3	88.7	0.3x

Table 2: Dev set results for models pretrained over BOOKCORPUS and Wikipedia for 125k steps. Here and everywhere else, the Avg column is computed by averaging the scores of the downstream tasks to its left (the two numbers of F1 and EM for each SQuAD are first averaged).

*Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut
 ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. p.4-5, 7-8. 2019

ALBERT

(1) Cross-layer parameter sharing

- ・レイヤー間でパラメーターを共有することで、各レイヤーの計算結果のばらつきが少なくなり、学習が安定する

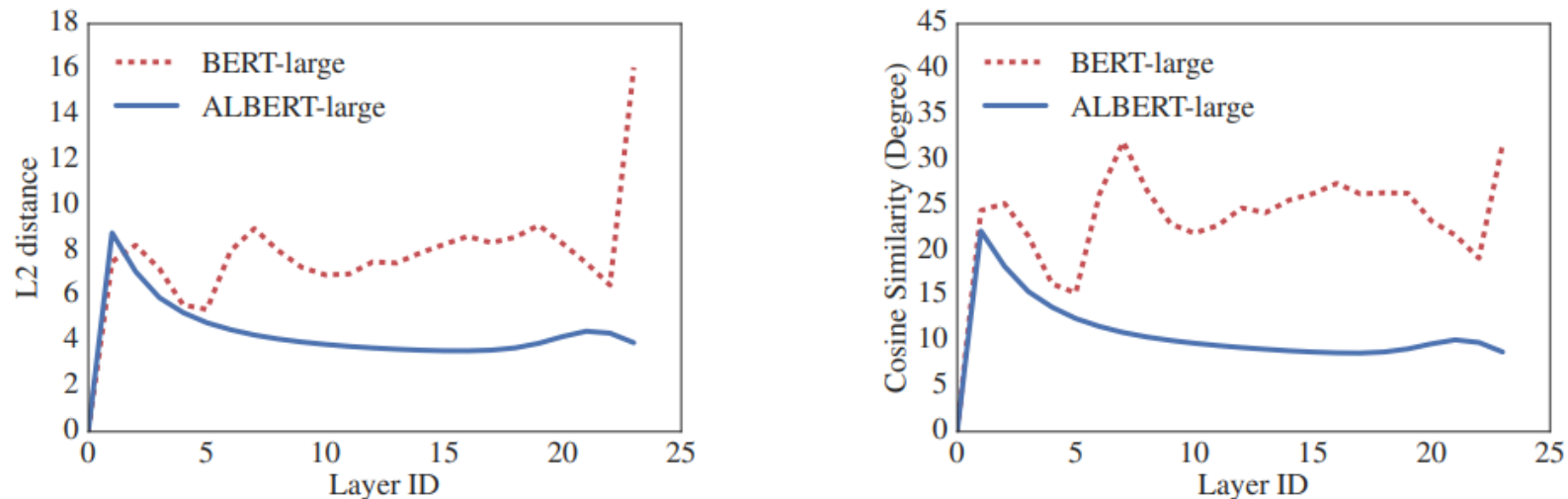


Figure 1: The L2 distances and cosine similarity (in terms of degree) of the input and output embedding of each layer for BERT-large and ALBERT-large.

*Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut
 ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. p.4-5, 7-8. 2019

ALBERT

(2) Sentence Order Prediction

- ・NSPによる学習ではSOPタスクを52.0%の正答率でしか解けない一方で、SOPによる学習ではNSPタスクも78.9%の正答率で解けるようになっている

SP tasks	Intrinsic Tasks			Downstream Tasks					Avg
	MLM	NSP	SOP	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	
None	54.9	52.4	53.3	88.6/81.5	78.1/75.3	81.5	89.9	61.7	79.0
NSP	54.5	90.5	52.0	88.4/81.5	77.2/74.6	81.6	91.1	62.3	79.2
SOP	54.0	78.9	86.5	89.3/82.3	80.0/77.1	82.0	90.3	64.0	80.1

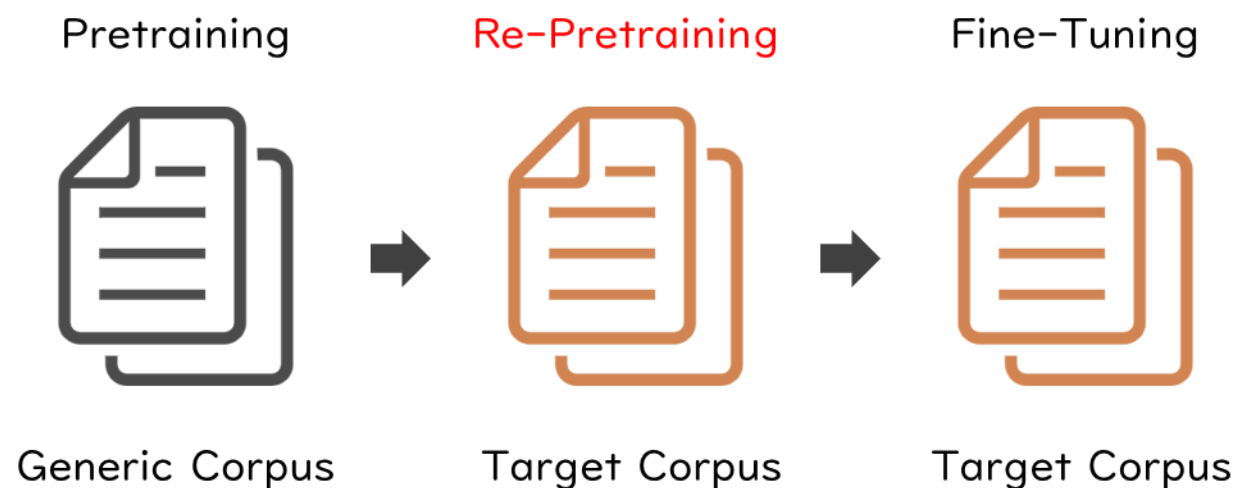
Table 5: The effect of sentence-prediction loss, NSP vs. SOP, on intrinsic and downstream tasks.

*Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut
 ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. p.4-5, 7-8. 2019

再事前学習

・ 転移学習

- 大規模コーパスによって学習した汎用的な言語能力をファインチューニングにより、特定のタスクに応用が可能
- しかしながら、強くドメインに依存するタスクである場合は、汎用的な言語能力だけでは対応しきれない
- ターゲットコーパスで再び事前学習を行うことで、ドメインに適応した事前学習モデルを構築することができる



実験

本研究では、次の三つの条件で実験を行い、情報抽出精度に関する比較した
また、学習にはALBERTを用いた

1. 大規模コーパス事前学習モデルによる転移学習
2. 小規模コーパスによる再事前学習
3. MeCabを用いた単語分割

実験

- 本研究では、これまでに作成したメール文とJSON形式のデータからSQuAD形式のデータを作成し、情報抽出に関して評価実験を行った

表1 メール文

【案件名】 AI Webアプリ開発案件
 【内容】 チャットボットを用いたWebアプリの設計・開発に携わっていただきます
 【勤務地】 茅場町（東京メトロ東西線・日比谷線）
 【期間】 10月～12月 ※延長の可能性あり
 【必須スキル】 Python, C#を使用した開発経験 自然言語処理に関する知識
 【推奨スキル】 javascript, django Webアプリ開発の経験

表2 JSON

```

{"job_name": ["AI Webアプリ開発案件"],
"contents": ["チャットボットを用いたWebアプリの設計・開発に携わっていただきます"],
"sites_station": ["茅場町"],
"durings": ["10月～12月"],
"price_min": [500000],
"price_max": [600000],
"age_min": [-1],
"age_max": [43],
"skill_required_os": [],
"skill_required_lang": ["Python", "C#"],
...,
"skill_recommended_web": ["JavaScript", "Django"],
"required_numbers": ["1名"],
"counts_for_interview": ["2回（弊社同席）"],
"etc": ["勤怠, コミュニケーションに問題のない方"]}
    
```

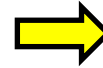


表3 SQuAD

```

{ 'paragraphs' : [{ 'context' : '【案件名】 AI Webアプリ・・・' ,
'qas' : [
    
```

```

    { 'answers' : [{ 'answer_start' : 7,
'text': 'AI Webアプリ開発案件'}],
'id': 'job_name_001',
'is_impossible': False,
'question': 'job_nameは何か?' },
    { 'answers': [{ 'answer_start' : 26,
'text' : 'チャットボットを用いたWebアプリの設計・開発に・・・'}],
'id': 'contents_001',
'is_impossible': False,
'question': 'contentsは何か?' },
    ...
    { 'answers': [{ 'answer_start': 193,
'text': '勤怠, コミュニケーションに問題のない方'}],
'id': 'etc_001',
'is_impossible': False,
'question': 'etcは何か?' }]}],
    
```



'title': 'job_001'}

実験

- ・ SQuAD形式のデータはJSONの項目数20に合わせ、各メール文に対する質問応答を作成した

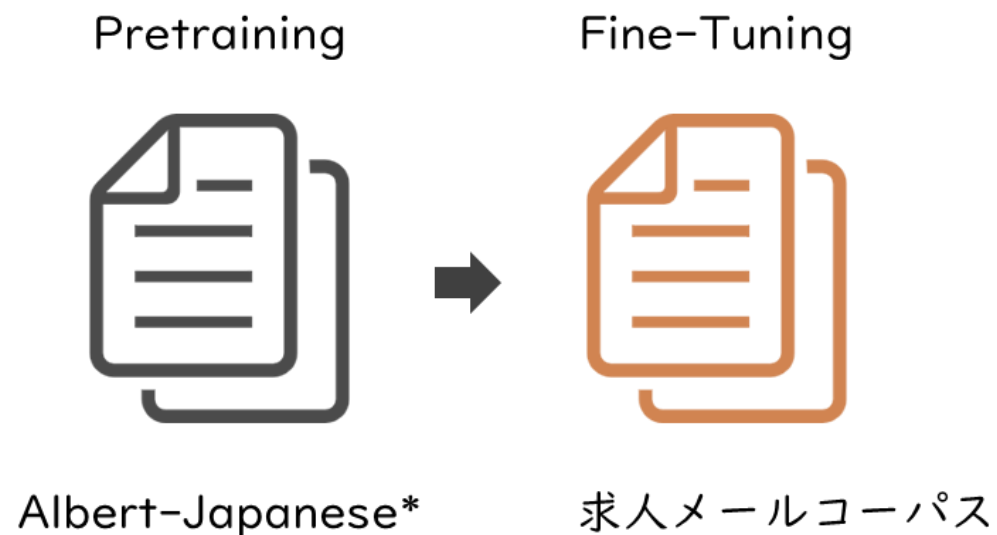
項目名	詳細
job_name	案件名
industry	業界
content	案件概要
sites	勤務地
durings	期間
price	単価
age	年齢
skill_required	必須スキル
skill_recommended	推奨スキル
environment	開発環境

項目名	詳細
required_numbers	要求人数
counts_for_interview	面談回数
working_time	始業時間, 終業時間
average_work_hours	平均労働時間
settlement_to_overhours	稼働時間
organization	組織名
payment_site	支払いサイト
commercial_flow	商流
can_accept_foreigner	外国人可否
etc	その他

実験

1. 大規模コーパス事前学習モデルによる転移学習

- Wikipediaの日本語全記事をもとに事前学習したalbert-japaneseを用いて転移学習を行った

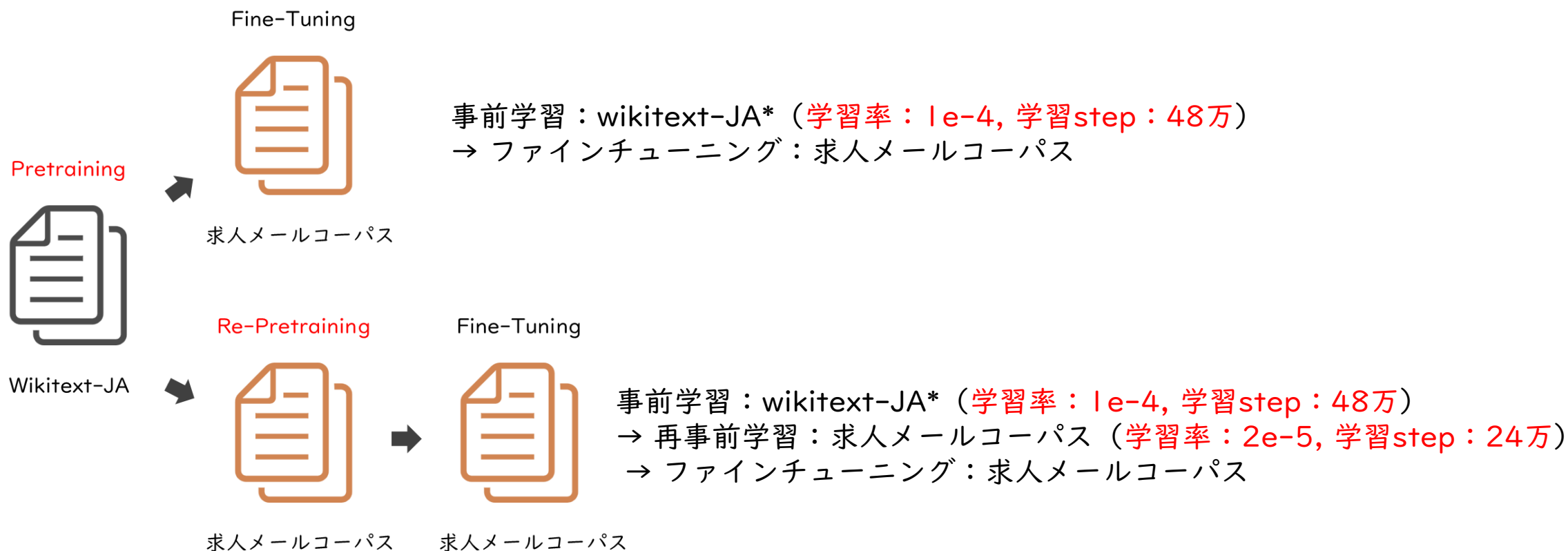


* “ALBERT with SentencePiece for Japanese text.” . <https://github.com/alinear-corp/albert-japanese>

実験

2. 小規模コーパスによる再事前学習

- Wikipediaの中でも秀逸な記事・良質な記事（1,647記事）を用いて事前学習を行い、その後、ターゲットドメインのコーパスを用いて再事前学習を行った

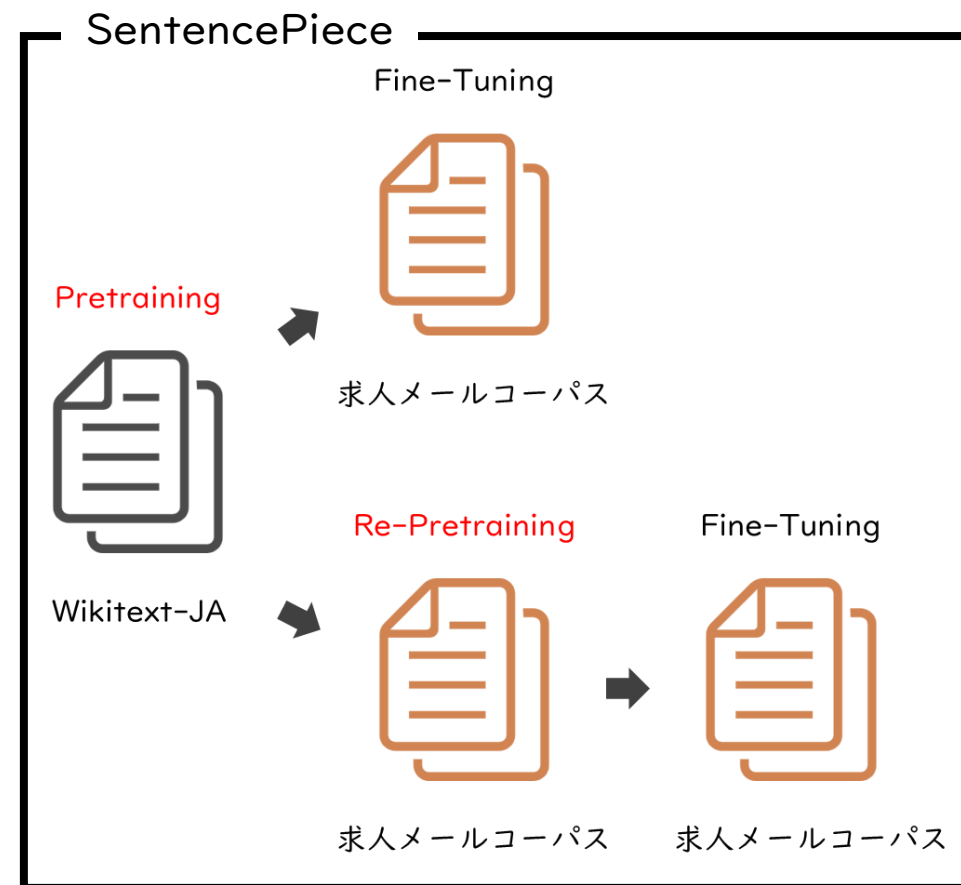
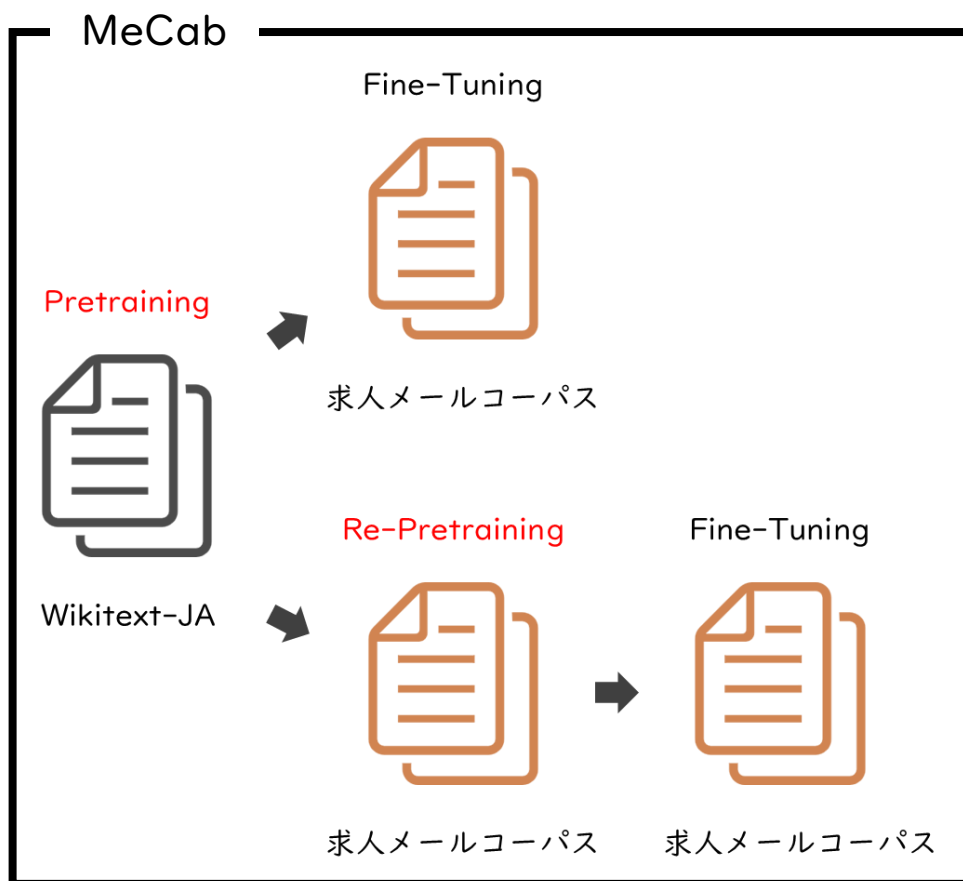


* "Wikitext-JA ". <http://www.lsta.media.kyoto-u.ac.jp/data/wikitext-ja/>

実験

3. MeCabを用いた単語分割

- 単語分割の方法をMeCabとSentencePieceで比較

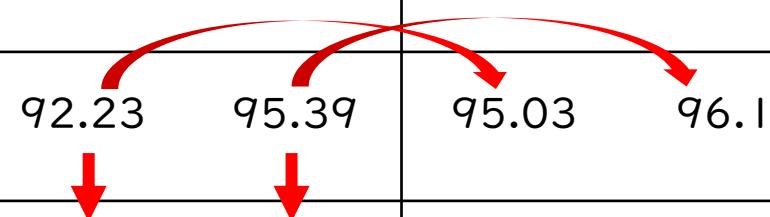


評価方法と結果

- 評価方法：Exact MatchとF値を用いて各質問に対応する箇所を抽出できているかを評価

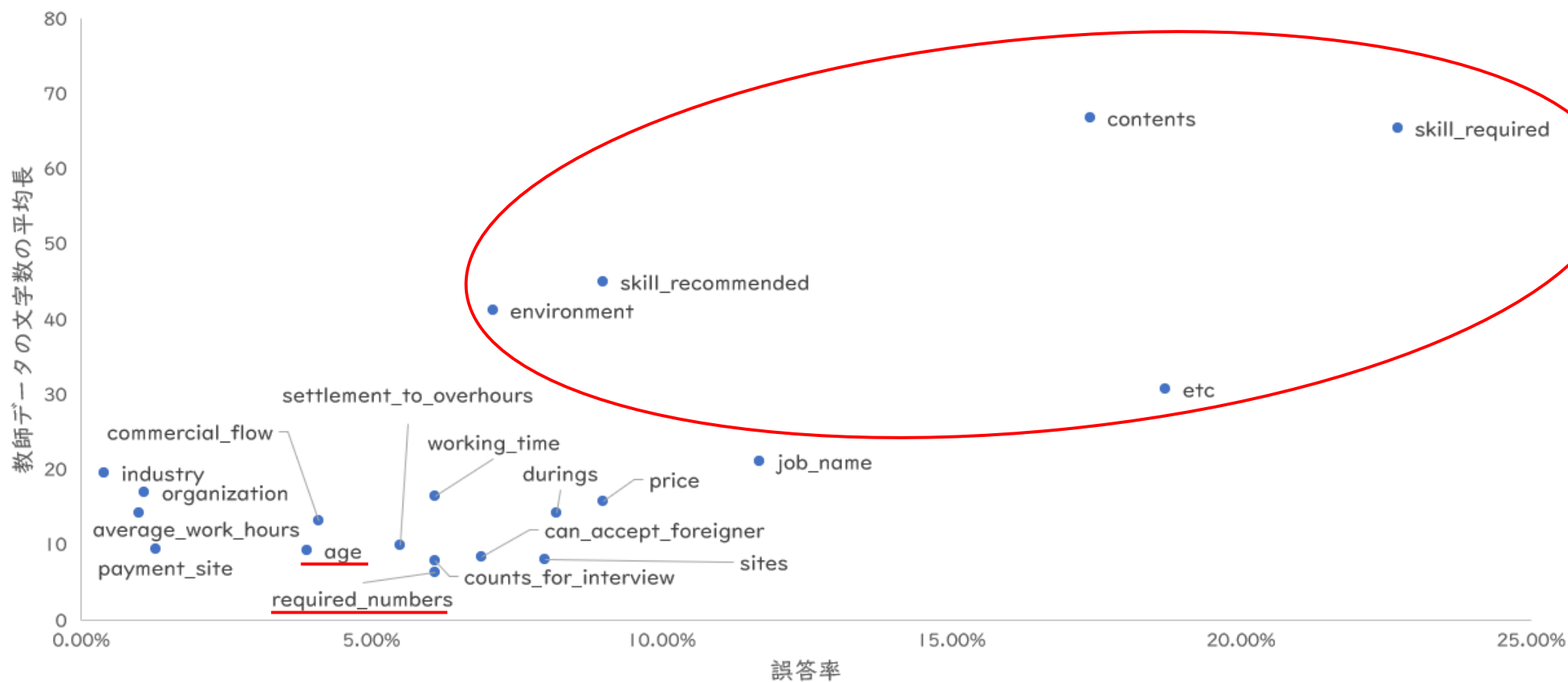
- 結果：

Pretrain model	Tokenizer			
	SentencePiece		MeCab	
	EM値	F値	EM値	F値
albert-japanese	96.95	97.79	-	-
Wikitext-JA (事前学習モデル)	92.23	95.39	95.03	96.13
Wikitext-JA (再事前学習モデル)	95.47	97.00	95.49	96.45



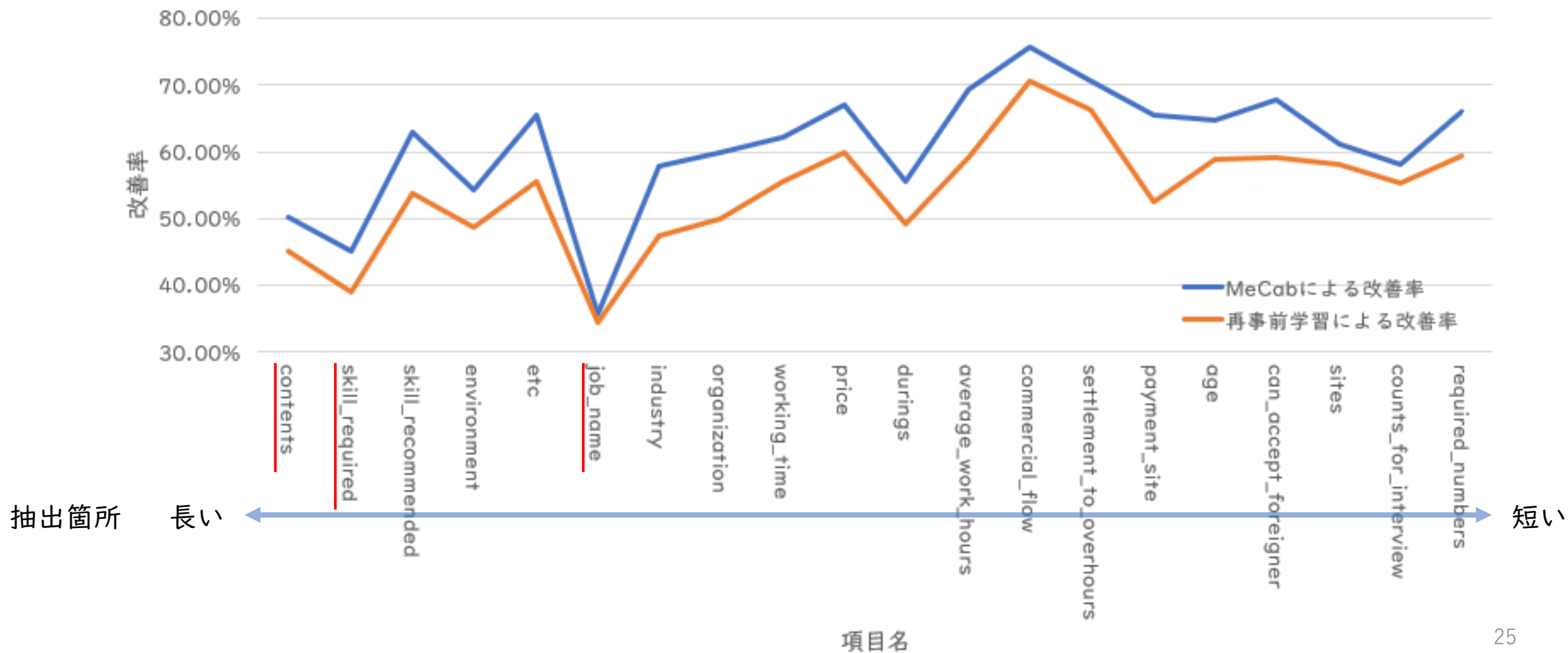
考察

- ・ Wikitext-JA（事前学習モデル）のEM値を算出した場合における、質問応答項目別の誤答率と正解コーパスにおける文字数の平均長



考察

- ・ Wikitext-JA（事前学習モデル）のEM値を算出した場合における誤答について、MeCabを用いて学習した場合と、再事前学習を行った場合における改善率





まとめ

- ・本研究では、ALBERTを用いて非構造化データの構造化における情報抽出の手法について検討した
- ・大規模コーパスによる事前学習には多少劣るものの、小規模コーパスでもトークナイザーにMeCab用いることや再事前学習を行うことで、より効率的に情報抽出をすることができた
- ・今後の研究としては、Transformer部分との結合を行い、構造化タスクとしての精度比較を行う予定
- ・また、膨大な非構造化データに対して、教師なし学習によるテキストセグメンテーションを行うことで、有益な情報とそうでない情報区別する手法について検討している



ご清聴ありがとうございました