マスク言語モデルを利用したデータ拡張に基づく日本語文内ゼロ照応解析

今野 颯人 [1]　松林 優一郎 [1,2]　清野 舜 [2,1]　大内 啓樹 [2,1]　高橋 諒 [1,2]　乾 健太郎 [1,2]
[1] 東北大学　[2] 理化学研究所
{ryuto,ryo.t,inui}@ecei.tohoku.ac.jp
y.m@tohoku.ac.jp,　{shun.kiyono,hiroki.ouchi}@riken.jp

1　はじめに

ゼロ照応解析（zero anaphora resolution; ZAR）は文中の述語の省略された項（ゼロ代名詞と呼ばれる）を特定するタスクである．日本語ではこの項の省略が頻出するため，意味解析を行う上での重要なタスクとして位置づけられている．

日本語における ZAR の例を図1に示す．ここでは，「逃走した」という述語の主語（項）が構文上省略されているため，この主語の特定が ZAR に分類される．ZAR は係り受け関係などの統語的手がかりが少ないため，解析が困難であることが知られている．例えば，ZAR の精度は，解析対象を述語と同一文内にある項に限っても，F_1 値が 58% 程度 [9] と低い水準に留まっている．本稿では，その要因の一つである，ZAR の教師データとして最も規模の大きなコーパス [4, 6] を用いても，その訓練事例数が十分ではないという問題点に注目する．我々は，ZAR のためのデータ不足問題を解決する方法として，構造アノテーション付きテキストデータのための新しいデータ拡張の手法を提案し，与えられた訓練データを最大限に活用することで ZAR の精度向上を目指す．

近年，文脈を考慮したデータ拡張の手法（contextual data augmentation; CDA）[3, 7, 15] は自然言語処理分野において活発に研究されている．CDA は訓練データ中の単語を別の単語で置換することによってデータ拡張を行う．その核となるアイデアは，文脈を考慮しながら置換先の単語を選択することである．具体的には，学習済み言語モデル（language model; LM）を用いて文脈を考慮した単語出現確率分布を計算し，そこから単語を選択する．この LM の利用法を **LM-for-replacement** と呼ぶことにする．我々の基本アイデアは，この CDA の手法を ZAR の訓練データ拡張に用いることである．しかし，CDA は以下の二つの理由で我々の目的に直接的には適用できない．

1. 近年の分類モデルでは，学習済み LM の最終隠れ層を入力素性として使うことが一般的となっている（この利用法を **LM-as-feature** と呼ぶ）が，CDA の LM-as-feature とこの方法の 2 種類の LM の利用法を統合する方法は自明ではない．
2. CDA は任意の単語を置換の対象として取り扱う．しかし，変更する単語によっては置換後の文と元の文の意味的/統語的な整合性が失われ，教師データとの対応が破壊される危険性がある．このため，文の統語的・意味的な構造を大きく変更するような単語の置き換えを抑制する必要がある．

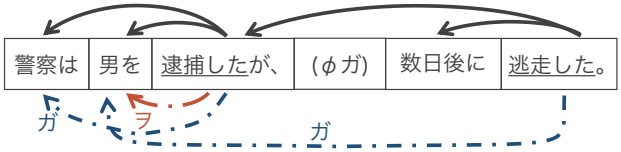

図1: 日本語ゼロ照応解析の例：実線が直接係り受けを，点線が項を表す．

そこで，我々はこれら二つの問題を解決するために既存の CDA の枠組みに（1）**マスクに基づくデータ拡張**と（2）**言語情報を利用したマスク戦略**という二つの拡張を加える（詳細は第4節を参照のこと）．実験では，BERT[2] を用いた強力なベースラインに対し，提案手法が解析性能の向上をもたらし，日本語 ZAR の世界最高性能を達成したことを示す．

2　文内ゼロ照応解析の定式化

日本語の ZAR は，同じく意味解析タスクの一つである述語項構造（Predicate-Argument Structure; PAS）解析の一部として定式化され，構文的に直接依存する項の解析（DEP）と同時に解析されることが多く [5, 9, 11]，本稿でもこの方法に習う．日本語の PAS 解析は，文 $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_I)$ と文内の述語 $\boldsymbol{p} = (p_1, \ldots, p_J)$ を入力として，それぞれの述語に対してガ格，ヲ格，ニ格となる項を特定するタスクである．文 $\boldsymbol{X}$ は各単語を表す one-hot ベクトル $\boldsymbol{x}_i \in \{0,1\}^{|\mathcal{V}|}$ で構成されているとする．$\mathcal{V}$ は語彙集合を表す．$p_j \in \mathbb{N}$ は述語の位置を表す 1 以上 I 以下の自然数である．

3 ベースモデル

本稿で使用するベースモデル（MP-MLM）は，Matsubayashi & Inui[9] の提案モデルである MP（multi-predicate）モデルを拡張したものである．唯一の違いは，MP-MLM では BERT[2] のような学習済みマスク言語モデル（masked language model; MLM）の最終隠れ層の系列を MP モデルの入力として用いる点である．MP-MLM は文 $\boldsymbol{X}$ と解析対象の述語位置 $p_j \in \boldsymbol{p}$ を入力とし，確率分布 $P(y_{i,j}|\boldsymbol{X}, \boldsymbol{p}, i, j)$ を計算する．ここで，$y_{i,j} \in \{$ ガ格，ヲ格，ニ格，NONE$\}$ は i 番目の単語 $\boldsymbol{x}_i$ と j 番目の述語のペアにおける項のラベルである．

まず，MP-MLM は入力文 $\boldsymbol{X}$ を学習済み MLM を用いてエンコードする．

$$\boldsymbol{E} = \overleftrightarrow{\mathrm{MLM}}(\boldsymbol{X}) \tag{1}$$

ここで $\overleftrightarrow{\mathrm{MLM}}$ は学習済み MLM の最終隠れ層 $\boldsymbol{E} = (\boldsymbol{e}_1, \ldots, \boldsymbol{e}_I)$ を計算する関数であり，$\boldsymbol{e}_i \in \mathbb{R}^D$ は文中の各単語に対応する最終隠れ層，D は隠れ層の次元数である．次に $\boldsymbol{E}$ は多層双方向 RNN に入力され，確率分布 $P(y_{i,j}|\boldsymbol{X}, \boldsymbol{p}, i, j)$ が計算される．

4 提案手法

研究背景とモチベーション CDA を用いた ZAR のデータ拡張を実現するため，本稿では二つの問題に取り組む．まず最初の問題として，学習済み LM には二つの異なる使い方，LM-for-replacement（図2-（a））と LM-as-feature（図2-（b））が存在するが，これらを統合することは自明ではない．具体的には，LM の異なる使い方として以下のような違いがある．CDA は入力文の任意の単語について，学習済み LM を用いて語彙確率分布を出力し，異なる単語に置換する [3, 7, 15]（LM-for-replacement）．一方で，最近の分類タスクの多くは LM の埋め込み表現 $\boldsymbol{E}$ そのものを入力素性として用いている [2, 12]（LM-as-feature）．

次の問題として，ZAR ではデータ拡張によって新たに生成された文に，元の文の意味構造を表す正解ラベルを使用する．そのため，新たに生成された文が元の意味的/統語的な構造を維持していることが重要である．CDA では任意の単語を置換の対象として取り扱う．しかし，特定の単語は意味構造に重要であり，これらを置換することによって得られた文は元の意味構造を維持できないことが予想される．よって任意の単語を置換することは好ましくない．

これら二つの問題に対して，我々は CDA に（i）**マスクに基づくデータ拡張**と（ii）**言語情報を利用したマスク戦略**といった二つの拡張を加え，LM の異なる利用法を統合し，かつより安全な方法でデータ拡張する手法を提案する．全体像を図2-（c）に示す．

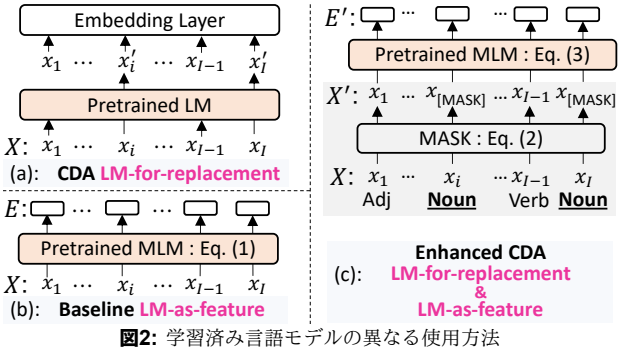

図2: 学習済み言語モデルの異なる使用方法

本研究による拡張 まず，二つの異なる LM の使い方を統合するため，**マスクに基づくデータ拡張**を提案する．この拡張では基本的に MLM の最終隠れ層を入力素性として使用する（LM-as-feature）．核となるアイデアは，文 $\boldsymbol{X}$ の単語を MLM のマスクトークン [MASK] で置換し，異なる最終隠れ層 $\boldsymbol{E}'$ を得ることである．狙いとして，MLM は [MASK] に文脈を考慮して単語を埋めるよう学習されていることから，ここで得られた最終隠れ層 $\boldsymbol{E}'$ は LM-for-replacement により得られたものとみなせる．この方法により，図2に示している二つの異なる LM の使い方を統合することができる．また，MLM によって [MASK] 箇所の予測を行うことで，[MASK] に対応する最終隠れ層が文脈上適切かつ，より抽象的な意味表現となることが期待できる．一方で元の CDA では，単語が別の単語で置換されたのち MLM によってエンコードされるため，同じ効果は望めない（図2-（a））．加えて，MLM が文脈を考慮して予測を行う性質上，マスクされていない単語に対応する最終隠れ層も元の表現から変化することにも注意されたい．

次に，入力をマスクしても元の構文/意味構造を維持するために，**言語情報を利用したマスク戦略**を提案する．ここでは品詞タグを使用してマスク箇所を制御する．直感的には，品詞の種類によってはマスクされるべきではないと思われるものがある．例えば，句読点や括弧などの記号は文の構文構造を決定づけるため，これらがマスクされた場合，マスクされた文から作られた中間表現と正解ラベルとの構造的な整合性が損なわれる可能性があり，モデルの訓練時にノイズとなると考えられる．

本稿の提案手法で拡張された CDA は，まず入力 $\boldsymbol{X}$ 内の単語を [MASK] を表す one-hot ベクトル $\boldsymbol{x}_{[\mathrm{MASK}]} \in \mathbb{R}^{|\mathcal{V}|}$ で置換する．その後，マスクされた入力 $\boldsymbol{X}'$ は MLM に与えられ，ベクトル系列 $\boldsymbol{E}' = (\boldsymbol{e}'_1, \ldots, \boldsymbol{e}'_I)$ を得る．

$$\boldsymbol{X}' = \mathrm{MASK}(\boldsymbol{X}; \mathcal{V}', \alpha), \tag{2}$$

$$\boldsymbol{E}' = \overleftrightarrow{\mathrm{MLM}}(\boldsymbol{X}'). \tag{3}$$

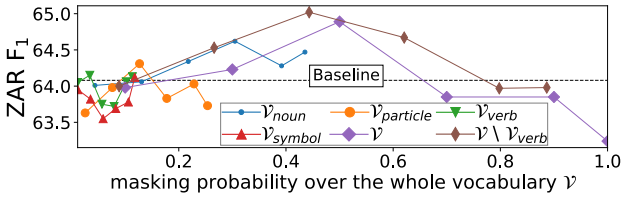

図3: マスク確率に伴う ZAR の F_1 値の推移

ここで，MASK は入力の $\mathcal{V}'$ に属する各単語を確率 α で $\boldsymbol{x}_{\texttt{[MASK]}}$ に置換する関数である．例えば，$\mathrm{MASK}(\boldsymbol{X}; \mathcal{V}' = \mathcal{V}_{\mathrm{noun}}, \alpha = 0.3)$ の場合について考える．この場合，単語 $\boldsymbol{x}_i$ が語彙集合 $\mathcal{V}_{\mathrm{noun}}$ に属するとき，割合 0.3 で $\boldsymbol{x}_{\texttt{[MASK]}}$ に置換される．ここで，$\mathcal{V}_{\mathrm{noun}}$ は訓練データ中に出現する名詞の集合を表す．以降の計算は第3節の MP-MLM と同様である．

5 実験

実験では，第4節で述べた二つの CDA の拡張の効果を検証する．まず，マスクする品詞タグの種類を変化させ，開発セットにおいて最適な設定を探索する（第5.2節）．その後，評価セットにおいて既存のデータ拡張の手法と比較する（第5.3節）．

5.1 実験設定

実験では NAIST Text Corpus（NTC）[4, 6] 1.5 を用いる．NTC 1.5 は ZAR や PAS 解析の既存研究 [9–11] で採用されているベンチマークデータセットである．Taira ら [14] の定義したデータ分割に従い訓練，開発と評価セットを作成した．各モデルは 10 個の異なるシードで訓練した後，F_1 の平均値を報告する．訓練済み MLM としては，日本語 Wikipedia で訓練された BERT[2] を用いる [13]．全ての実験で BERT のパラメータは固定した．マスクした文 $\boldsymbol{X}'$ と元の文 $\boldsymbol{X}$ を 1:1 の割合で混ぜて学習に用いた．解析の対象である述語位置 p_j はマスクの対象からは除外した[*1]．

5.2 CDA への拡張の効果の検証

本実験では，マスクの対象とする品詞タグの種類を変化させた際のモデルの性能の変化を調べる．具体的には，次に挙げる設定の組み合わせの全てでモデルを訓練する．(i) 品詞タグの種類: $\{\mathcal{V}_{\mathrm{noun}}, \mathcal{V}_{\mathrm{verb}}, \mathcal{V}_{\mathrm{particle}}, \mathcal{V}_{\mathrm{symbol}}, \mathcal{V}\}$.[*2] (ii) マスク確率 α: {0.1, 0.3, 0.5, 0.7, 0.9, 1.0}.[*3] 本実験では，計算コストを減らすため各エポックでのマスク位置は固定する．

各マスク確率と品詞タグの種類における ZAR F_1 を図3に示す．図3でのマスク確率は全単語中における割合を表すことに注意されたい．ここでは，比較的高いマスク確率（約 0.5）が良い性能となりやすい傾向が読み取れる．

[*1] 予備実験では，除外しない設定において精度の低下が確認できたため．
[*2] コーパスの 10% 以上を占める品詞タグを選択した．
[*3] 1.0 は対象とする品詞タグの単語全てをマスクすることを示す．

Method	ALL F_1	SD	ZAR F_1	DEP F_1
Baseline	87.43	±0.14	64.08	92.82
$\mathcal{V}$	87.64	±0.09	64.89	92.94
$\mathcal{V}_{\mathrm{noun}}$	87.53	±0.09	64.62	92.87
$\mathcal{V}_{\mathrm{verb}}$	87.35	±0.09	64.15	92.78
$\mathcal{V}_{\mathrm{particle}}$	87.43	±0.19	64.31	92.82
$\mathcal{V}_{\mathrm{symbol}}$	87.29	±0.16	64.12	92.74
$\mathcal{V}\backslash\mathcal{V}_{\mathrm{noun}}$	87.44	±0.16	64.34	92.85
$\mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}$	87.67	±0.11	**65.02**	92.96
$\mathcal{V}\backslash\mathcal{V}_{\mathrm{particle}}$	87.44	±0.15	64.23	92.83
$\mathcal{V}\backslash\mathcal{V}_{\mathrm{symbol}}$	87.59	±0.19	64.66	**92.98**
$\mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}} \cup \mathcal{V}_{\mathrm{symbol}}$	**87.72**	±0.16	64.81	92.97

表1: NTC 1.5 **開発セット**における F_1 値．手法名はマスク対象である品詞タグの集合を表す．**太字**の値は同じ列における最高精度を表す．

Method	ALL	SD	ZAR ALL	ZAR NOM	ZAR ACC	ZAR DAT	DEP ALL
M&I	83.94	±0.12	55.55	57.99	48.9	23	90.26
O&K	83.82	±0.10	53.50	56.37	45.36	8.70	90.15
Baseline†	86.85	±0.11	63.89	66.45	57.2	27	92.24
Replace†	86.84	±0.19	63.87	66.16	58.5	**29**	92.27
ZeroDrop†	86.94	±0.14	64.23	66.80	57.4	28	92.29
Masking†	86.98	±0.13	64.15	66.60	57.9	**29**	92.46
MaskDrop†	**87.14**	±0.11	**64.86**	**67.36**	**58.6**	**29**	**92.52**
M&I*	85.34	-	58.07	60.21	52.5	26	91.26
MaskDrop†*	**88.35**	-	**68.02**	**70.59**	**61.3**	**30**	**93.25**

表2: NTC 1.5 の評価セットにおける F_1 値．**太字**の値は各列における最高性能を示す．†: 我々の実験結果．∗: アンサンブルモデル．M&I: Matsubayashi ら [9] の結果．O&K: Omori らの結果 [10]．

図3より，各品詞タグの種類について最も高い ZAR F_1 を達成したモデルの結果を表1に示す．表より，$\mathcal{V}$ の設定が Baseline の ZAR F_1 を改善したとわかる．この結果より，**マスクに基づくデータ拡張**が有効であることが示唆される．

次に，表1より**言語情報を利用したマスク戦略**の効果は次の結果で検証される：(i) $\mathcal{V}$ とは異なり，$\mathcal{V}_{\mathrm{verb}}$ と $\mathcal{V}_{\mathrm{symbol}}$ は性能の向上は見られなかった：これらの ZAR F_1 は Baseline とほぼ同等である；(ii) 動詞と記号をマスクしない設定 ($\mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}$ と $\mathcal{V}\backslash\mathcal{V}_{\mathrm{symbol}}$) はそれぞれ最も高い ZAR F_1 と DEP F_1 を達成した．

5.3 他手法との比較

本節では，我々の提案手法と（i）Baseline から性能の向上が見込める他手法と（ii）既存研究との比較を行う．ここでは，我々の提案手法を Masking と呼ぶ．本実験では，表1において ZAR F_1 の最高性能を達成した $\mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}$ 設定を用いる．また，既存研究に従い [8] 各エ

										NOM		PRED	
$\boldsymbol{X}$	内閣	改造	を	通常	国会	召集	前	に	やる	**考え**	は	ない	。
$\boldsymbol{X}'$	内閣	[M]	[M]	[M]	[M]	[M]	[M]	に	やる	**考え**	[M]	ない	。
$\boldsymbol{X}''$	内閣	は	は	、	、	、	ため	に	やる	**考え**	は	ない	。

図4: Replace におけるデータ拡張の実例．$\boldsymbol{X}' = \mathrm{MASK}(\boldsymbol{X}; \mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}, \alpha = 0.5)$．[M] は [MASK] を表す．

ポックでマスクする位置をランダムに変更する．

比較対象とする手法は以下の二つである．

- ZeroDrop: MLM の出力 $\boldsymbol{E}$ を受け取り，各 $\boldsymbol{e}_i$ が $\mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}$ に属する場合は確率 α でゼロベクトル $\mathbf{0} \in \mathbb{R}^D$ に置換する．この手法は word-dropout に着想を得ている [1, 16]．
- Replace: まず最初にマスクされた系列 $\boldsymbol{X}' = \mathrm{MASK}(\boldsymbol{X}; \mathcal{V}\backslash\mathcal{V}_{\mathrm{verb}}, \alpha = 0.5)$ を受け取り，[MASK] を MLM が予測した単語で埋める．新しい系列を $\boldsymbol{X}''$ と表す．その後，系列 $\boldsymbol{X}''$ を MLM を用いて $\boldsymbol{E}$ へとエンコードする．このモデルは CDA の手法を直接 LM-as-feature に適用させたものとみなせる．なお，文中に複数の [MASK] が存在する場合は，全てを同時に埋める．*4

結果を表2に示す．まず，Baseline の性能が既存研究 [9, 10] をすでに大幅に上回っていることに注意されたい．我々の提案手法である Masking を加えることで，この強力な Baseline から更に性能が向上した（ZAR F_1 で 0.26 ポイント向上）．一方で，Replace の ZAR F_1 は Baseline とほぼ同等の結果となった．この結果は我々の仮説を裏付けるものである：単純に CDA を適用することは，LM-as-feature のモデルには効果が見込めない．また，Masking の代替手法である ZeroDrop も ZAR F_1 において Baseline から精度向上が見られた．この結果から，我々は Masking と ZeroDrop を組み合わせた．この手法を MaskDrop と呼ぶ．MaskDrop では，式3における入力 $\boldsymbol{E}'$ を受け取り，[MASK] に対応する位置に ZeroDrop を適用する．このモデルは ZAR F_1 において 64.86 と最高精度を達成した．

MaskDrop が ZAR F_1 の最高性能を達成したという事実は，これら二つの方法が相補的な関係にあることを示している．この精度改善の理由を，図4の実例で説明する．図4は元の文 $\boldsymbol{X}$，マスクされた文 $\boldsymbol{X}'$，$\boldsymbol{X}'$ のマスク部分が BERT の予測単語で埋められ変更された文 $\boldsymbol{X}''$ を示す．$\boldsymbol{X}''$ において，マスクされた単語は意味のない反復的な助詞や句読点で埋められているため，訓練中にノイズとして機能する可能性がある．ZeroDrop はこのようなノイズを除去することで，Masking の性能を向上させると考えられる．

*4 一度に一つずつ [MASK] を埋める手法も考えられるが，予備実験において全ての [MASK] を同時に埋める手法がが高い性能を示した．

6 おわりに

本稿では，日本語の ZAR における訓練データが十分でない問題にデータ拡張のアプローチによって取り組んだ．我々は，言語情報を利用したマスク戦略を元に，学習済み LM の二つの異なる使用法を組み合わせた新しい手法を提案した．実験によって，動詞以外の語彙全体をランダムにマスクすることが性能を改善することを示した．word-dropout に基づく手法を組み合わせることで性能をさらに改善し，世界最高性能を達成した．

謝辞. 本研究は JST CREST（課題番号: JPMJCR1513）の支援および JSPS 科研費 JP19K12112 の助成を受けたものである．

参考文献

[1] Kevin Clark et al. “Semi-Supervised Sequence Modeling with Cross-View Training”. In: *EMNLP*. 2018, pp. 1914–1925.
[2] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *NAACL*. 2019, pp. 4171–4186.
[3] Fei Gao et al. “Soft Contextual Data Augmentation for Neural Machine Translation”. In: *ACL*. 2019, pp. 5539–5544.
[4] Ryu Iida et al. “Annotating Predicate-Argument Relations and Anaphoric Relations: Findings from the Building of the NAIST Text Corpus”. In: *Natural Language Processing* 17.2 (2010), pp. 25–50.
[5] Ryu Iida et al. “Intra-sentential Zero Anaphora Resolution using Subject Sharing Recognition”. In: *EMNLP*. 2015, pp. 2179–2189.
[6] Ryu Iida et al. “NAIST Text Corpus: Annotating Predicate-Argument and Coreference Relations in Japanese”. In: *Handbook of Linguistic Annotation*. Springer, 2017, pp. 1177–1196.
[7] Sosuke Kobayashi. “Contextual Augmentation: Data Augmentation by Words with Paradigmatic Relations”. In: *NAACL*. 2018, pp. 452–457.
[8] Yinhan Liu et al. “RoBERTa: A Robustly Optimized BERT Pretraining Approach”. In: *arXiv preprint arXiv:1907.11692* (2019).
[9] Yuichiroh Matsubayashi and Kentaro Inui. “Distance-Free Modeling of Multi-Predicate Interactions in End-to-End Japanese Predicate-Argument Structure Analysis”. In: *COLING*. 2018, pp. 94–106.
[10] Hikaru Omori and Mamoru Komachi. “Multi-Task Learning for Japanese Predicate Argument Structure Analysis”. In: *NAACL*. 2019, pp. 3404–3414.
[11] Hiroki Ouchi and Yuji Shindo Hiroyuki and Matsumoto. “Neural Modeling of Multi-Predicate Interactions for Japanese Predicate Argument Structure Analysis”. In: *ACL*. 2017, pp. 1591–1600.
[12] Matthew Peters et al. “Deep Contextualized Word Representations”. In: *NAACL*. 2018, pp. 2227–2237.
[13] Tomohide Shibata, Daisuke Kawahara, and Sadao Kurohashi. “Improving Japanese Syntax Parsing with BERT”. In: *Natural Language Processing* (2019), pp. 205–208.
[14] Hirotoshi Taira, Sanae Fujita, and Masaaki Nagata. “A Japanese Predicate Argument Structure Analysis using Decision Lists”. In: *EMNLP*. 2008, pp. 523–532.
[15] Xing Wu et al. “Conditional BERT Contextual Augmentation”. In: *ICCS*. 2019, pp. 84–95.
[16] Ziang Xie et al. “Data Noising as Smoothing in Neural Network Language Models”. In: *ICLR*. 2017.