

# 静的および動的な埋め込みに基づくエンティティの曖昧性解消

大葉大輔<sup>1,2</sup> 山田育矢<sup>2,3</sup> 吉永直樹<sup>4</sup> 豊田正史<sup>4</sup><sup>1</sup> 東京大学 <sup>2</sup> Studio Ousia <sup>3</sup> RIKEN <sup>4</sup> 東京大学 生産技術研究所

oba@tkl.iis.u-tokyo.ac.jp ikuya@ousia.jp {ynaga, toyoda}@iis.u-tokyo.ac.jp

## 概要

エンティティの曖昧性解消タスクでは、各エンティティを埋め込み表現としてパラメタ化した上で曖昧性解消モデルの学習を行うことが有効である (Yamada et al. 2021) が、パラメタ化対象外のエンティティの曖昧性解消は行えない。本稿では、エンティティの説明文やエンティティを含む文からエンティティの埋め込み表現を動的に推定し、学習済み曖昧性解消モデルを拡張する手法を提案する。実験では、評価データに出現する候補エンティティ全てをパラメタ化した上で訓練された既存モデルらと同等またはそれ以上の性能が得られることを示した。

## 1 はじめに

入力テキストに含まれるエンティティのメンションを知識ベース (KB) のエントリに分類する Entity Disambiguation (ED) は、文書分類や質問応答等の言語処理タスクや、テキストからの KB 構築等の情報抽出タスクにおける重要な前処理の一つである [1]。

近年、ED モデルのパラメタとして分類候補のエンティティ集合に埋め込み表現を与えた上で、Wikipedia のハイパーリンクを正解データとする大量の学習データを用いた訓練を行うことで ED を高精度に解けるようになった [2, 3, 4, 5, 6, 7]。しかし、埋め込み表現は訓練時にパラメタ化の対象となる一部のエンティティに対してのみ学習されるため、その他エンティティ (未知エンティティ) のメンションは曖昧性を解消できない。実際、多くの既存システムは、評価時の正解候補エンティティすべてをパラメタ化した上で訓練および評価を行っている。

対して、エンティティの説明文とエンティティを言及する文のエンコード結果を近づける学習により zero-shot で ED を解く方法 [8, 9] は未知エンティティの曖昧性解消を可能とする。しかし、埋め込み表現に基づいた上述の方法より性能が低い上 [10]、各メンションについて、その候補エンティティの数

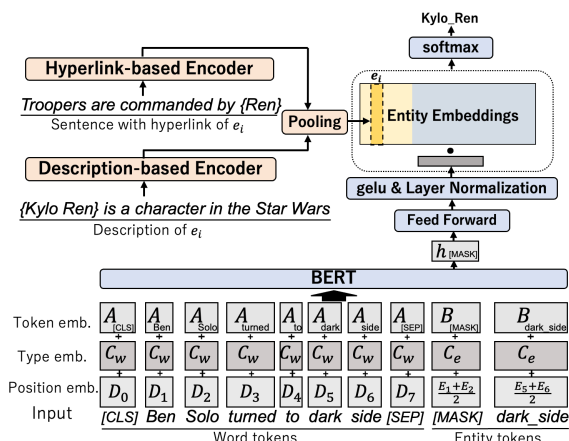


図 1 提案手法の概要。エンティティの説明文 (Description) やエンティティを言及する文 (Sentence with hyperlink) からその埋め込み表現を推定し、ED モデルのエンティティ埋め込み表現行列 (Entity Embeddings) を補充する。

だけエンコーダの順伝搬計算を必要とする。

本稿では、訓練時にパラメタ化されたエンティティの静的な埋め込み表現と、テキストから推定した動的な埋め込み表現を相補的に用いることを提案 (§ 3) し、静的なエンティティ表現に基づく ED モデル [7] (§ 2) を拡張する (図 1)。具体的には、まず高頻度エンティティに限定した形で ED モデルを構築し、Wikipedia のハイパーリンクを正解とするデータを用いて訓練を行う (§ 3.1)。次にエンティティの埋め込み表現を動的に推定する 2 種類のエンコーダを訓練する (§ 3.2)。そして低頻度エンティティの埋め込み表現を推定し、ED モデルのパラメタ (エンティティの埋め込み表現行列) を拡張する (§ 3.2)。これにより学習時にパラメタ化されない未知エンティティを含む評価データでも、精度の高い静的なエンティティ表現に基づく ED を適用出来る。

代表的な評価データ (AIDA-CoNLL, MSNBC, AQUAINT, ACE2004, WNED-CWEB, WNED-WIKI) を用いた実験では、提案手法により、正解候補エンティティが既知な状態で訓練を行う既存手法と同様またはそれ以上の性能が得られることを示す (§ 4)。

## 2 事前知識

本節では、我々が基盤とする Yamada ら [7] の Global Model (以降  $\mathbf{Y}_G$ ) を説明する。 $\mathbf{Y}_G$  は  $\text{BERT}_{\text{LARGE}}$  [11] を基盤とする現在の最高精度のモデルである。テキストの単語とエンティティのトークン列を入力に取り (§ 2.1), 各エンティティトークンに対応する BERT の出力を基に曖昧性解消を行う (§ 2.2)。以降、単語とエンティティの語彙セットを  $V_w, V_e$ , 埋め込み表現の次元を  $H$  とする。

### 2.1 入力表現

$\mathbf{Y}_G$  では、入力テキストの単語トークン列とその内のエンティティに対応するトークン列を結合し、単語トークン列の先頭と末尾に特殊トークンの [CLS] と [SEP] を挿入する。この時、各トークンは以下3種の埋め込み表現の和により表現される:

**Token embedding:** 各単語およびエンティティに対応。表現行列は  $\mathbf{A} \in \mathbb{R}^{|V_w| \times H}$  および  $\mathbf{B} \in \mathbb{R}^{|V_e| \times H}$ 。

**Type embedding:** トークンの種類 (単語かエンティティか) に対応。  $\mathbf{C}_w \in \mathbb{R}^H$  および  $\mathbf{C}_e \in \mathbb{R}^H$ 。

**Position embedding:** 入力系列中の単語トークンの位置に対応。位置  $i$  にある単語およびエンティティの位置埋め込みは、  $\mathbf{D}_i \in \mathbb{R}^H$  および  $\mathbf{E}_i \in \mathbb{R}^H$  とする。複数トークンで構成されるエンティティについては対応する位置埋め込みを平均する。

### 2.2 曖昧性解消

$\mathbf{Y}_G$  は Masked Language Model (MLM) [11] の要領でエンティティの曖昧性解消を行う。より具体的には、メンションに該当する入力エンティティトークンを [MASK] に置き換え、対応する BERT の出力  $\mathbf{h}_e \in \mathbb{R}^H$  からエンティティの推定を行う:

$$\mathbf{m}_e = \text{layernorm}(\text{gelu}(\mathbf{W}_f \mathbf{h}_e + \mathbf{b}_f)) \quad (1)$$

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{B} \mathbf{m}_e + \mathbf{b}_o) \quad (2)$$

$\mathbf{W}_f \in \mathbb{R}^{H \times H}$ ,  $\mathbf{b}_o$ ,  $\mathbf{b}_f \in \mathbb{R}^H$  はパラメタであり,  $\text{gelu}$  は活性化関数 [12],  $\text{layernorm}$  は層正則化 [13] を表す。

**訓練** エンティティトークンを一定確率で [MASK] に置換し、対応する推定結果  $\hat{\mathbf{y}}$  の対数尤度を最大化させる最適化を行う。Wikipedia のハイパーリンクを正解ラベルと見なした訓練データを使用する。

**推論**  $\mathbf{Y}_G$  は推論を逐次的に行う。まず全エンティティトークンをマスクし、各 [MASK] に対応するエンティティを推定する (式 2)。次に  $\hat{\mathbf{y}}$  の最大確

率が最も高い [MASK] を、その推定結果のエンティティトークンに置き換えて新しい入力を得る。この処理を全 [MASK] が曖昧性解消されるまで繰り返す。

### 2.3 問題点

$\mathbf{Y}_G$  は多くの評価データ (§ 4.1) で最高精度を達成している [7] が、 $V_e$  外のエンティティは学習時に埋め込み表現としてパラメタ化されないため曖昧性解消が行えない。全てのエンティティを  $V_e$  に含める場合、訓練時の計算量が大幅に増加するだけでなく特に低頻度エンティティに関するデータの過疎性が問題になる。また、こうした制約から  $\mathbf{Y}_G$  を含む先行研究の多く [2, 3, 4, 5, 6, 7] は訓練の段階で評価データに出現する全メンションのリンク先候補エンティティの集合を  $V_e$  に定め、エンティティの埋め込み行列  $\mathbf{B} \in \mathbb{R}^{|V_e| \times H}$  を学習し評価に使用している。しかし、訓練の段階で評価データに現れる全メンションを知っている設定は非現実的である。

## 3 提案

前節の問題点を踏まえ、訓練時のパラメタ化対象外のエンティティの埋め込み表現を動的に推定することを提案する。まず前節で導入した  $\mathbf{Y}_G$  を基盤に、Wikipedia で頻出する一定数のエンティティ  $V'_e$  のみ静的な埋め込み表現を与え、その  $V'_e$  に閉じた ED モデルの訓練を行う (§ 3.1)。次にその他エンティティ  $V''_e \notin V'_e$  の埋め込み表現を別途推定し (§ 3.2), 訓練済みモデルの静的な埋め込み表現行列を拡張する (§ 3.2)。これにより、頻出エンティティに関しては高い性能への寄与が報告されている静的な埋め込み表現を保証しつつ、他のエンティティが出現する評価データについても同じ ED モデルで曖昧性解消が行える。

### 3.1 高頻度エンティティに限定した訓練

**実装** 後に拡張可能な形式で  $\mathbf{Y}_G$  の再定義 ( $\mathbf{Y}'_G$ ) を行う。エンティティ埋め込み行列  $\mathbf{B}$  のみを補完するため、項  $\mathbf{b}_o$  (式 2) を取り除く。また高頻度エンティティ  $V'_e$  のみで表現行列  $\mathbf{B}' \in \mathbb{R}^{|V'_e| \times H}$  を構成する:

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{B}' \mathbf{m}_e) \quad (3)$$

**訓練**  $\mathbf{Y}_G$  と同じく Wikipedia のハイパーリンクを正解データとして、エンティティトークンを 30% の確率で [MASK] に置換し、[MASK] に対応する推定結果  $\hat{\mathbf{y}}$  (式 3) の対数尤度を最大化させる最適化を行う。

### 3.2 エンティティ表現の推定エンコーダ

異なる言語資源に基づく2つのエンコーダを用いてエンティティの埋め込み表現を動的に推定する。エンコーダは訓練済み  $Y'_G$  (§ 3.1) を基盤とする:

**Description-based Encoder** エンティティ  $e_i$  の説明文を § 2.1 と同じ方法で  $Y'_G$  の入力表現とし, [CLS] に対応する BERT の出力表現  $\mathbf{h}_{CLS}$  からエンティティ  $e_i$  の埋め込み表現を推定する:

$$\hat{\mathbf{e}}_{DESC_i} = \mathbf{W}_{DESC} \mathbf{h}_{CLS} + \mathbf{b}_{DESC} \quad (4)$$

$\mathbf{W}_{DESC} \in \mathbb{R}^{H \times H}$  および  $\mathbf{b}_{DESC} \in \mathbb{R}^H$  は学習パラメタ。

**Hyperlink-based Encoder** エンティティのハイパーリンクを含む文集合を用いて埋め込み表現を推定する。まず,  $e_i$  のハイパーリンクを含む文を [SEP] を挟んで連結したものを § 2.1 における単語トークン列とする。次に, エンティティトークン列のうち  $e_i$  に該当するものを [MASK] に置換し, 各 [MASK] に対応する BERT の出力  $[\mathbf{h}_{MASK_1} \dots \mathbf{h}_{MASK_N}]$  の平均から埋め込み表現を推定する:

$$\bar{\mathbf{h}}_{MASK} = \frac{1}{N} \sum_j \mathbf{h}_{MASK_j} \quad (5)$$

$$\hat{\mathbf{e}}_{HL_i} = \mathbf{W}_{HL} \bar{\mathbf{h}}_{MASK} + \mathbf{b}_{HL} \quad (6)$$

$\mathbf{W}_{HL} \in \mathbb{R}^{H \times H}$  および  $\mathbf{b}_{HL} \in \mathbb{R}^H$  は学習パラメタ。

**訓練** 訓練済み  $Y'_G$  (§ 3.1) におけるエンティティ埋め込み表現行列  $\mathbf{B}'$  を教師信号とし, エンティティ  $e_i$  に該当するインデックスの埋め込み表現と, エンコーダが推定した埋め込み表現  $\hat{\mathbf{e}}_i$  の平均二乗誤差を最小化する。埋め込み表現行列  $\mathbf{B}'$  のみ固定し, 高頻度エンティティ  $V'_e$  に関する Wikipedia エントリから抽出した説明文 (概要) およびそれらのハイパーリンクを含む文を学習データに用いて訓練する。

**推定表現のアンサンブル** 最終的に, 上記の異なるエンコーダから推定された埋め込み表現を平均化することで一つのエンティティ埋め込み表現とする。これにより, 2つの異なる言語資源に含まれる情報を利用できるようにする。

**エンティティ表現の推定と補完** 高頻度エンティティ集合に含まれないエンティティ集合  $V''_e \notin V'_e$  の埋め込み表現を上記エンコーダで推定し, 訓練済み  $Y'_G$  (§ 3.1) の埋め込み表現行列  $\mathbf{B}'$  (式 3) を補完する ( $\mathbf{B}''' \in \mathbb{R}^{|V'_e+V''_e| \times H}$ ).  $Y'_G$  は追加のエンティティ集合に対しても同様の推論が行える:

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{B}''' \mathbf{m}_e) \quad (7)$$

表 1 AIDA-CoNLL における In-KB Mico-F1. 最も性能が良い2つの数値に太字, 下線の順で印をつけた。

Method	KB+YAGO	PPRforNED
Ganea と Hofmann [3]	92.2	-
Le と Titov [4]	93.1	-
Broscheit [5]	87.9	-
Février ら [6]	92.5	96.7
Yamada ら [7]	<b>95.0</b>	<b>97.1</b>
<b>Ours</b>	92.3	95.4
<b>w/ fine-tuning</b>	<b>94.3</b>	<b>96.9</b>

## 4 実験

Yamada ら [7] の  $Y_G$  (§ 2) に提案手法を実装し, 評価データ中の全候補エンティティの埋め込み表現をパラメタを持った上で訓練された既存システム [3, 4, 5, 6, 7] と同等の性能が得られるか確認する。

### 4.1 実験設定

**実装** まず, BERT<sub>BASE</sub> [11] を基に, Wikipedia 上の頻出エンティティ  $|V'_e| = 1M$  に限定した  $Y_G$  の再定義 ( $Y'_G$ ) および訓練 (§ 3.1) を行う。評価データで使用される約 270K 件の正解候補エンティティうち,  $V'_e$  のカバレッジは 55.9%, 既存システムは 100% である。次に, 訓練済み  $Y'_G$  と Wikipedia を用いて提案エンコーダを訓練した後, 未知エンティティ  $V''_e \notin V'_e$  の埋め込みを推定し  $Y'_G$  を拡張する (§ 3.2)。

**評価データ** Ganea & Hofmann (2017) [3] の設定に倣い, AIDA-CoNLL [14], MSNBC, AQUAINT, ACE2004, WNED-CWEB, WNED-WIKI [15] を用いる。また, メンション表層に対する各エンティティへの被リンク事前確率 [3] に基づいて, 正解候補エンティティ集合を各メンションに割り当てる (KB+YAGO) [14]。式 7 における推論の際には, 候補エンティティ集合に対応するインデックスの埋め込み表現行列のみを用いて内積計算および softmax を実行する。AIDA-CoNLL では異なる候補エンティティ集合 PPRforNER [16] を使用した評価も行う。

**評価指標** micro-F1 値を報告する。

**Fine-Tuning** AIDA-CoNLL では, 提案手法により拡張した  $Y'_G$  を付属訓練セットを用いて Fine-tuning した評価も行う。Yamada ら [7] と同様にエンティティの埋め込み表現行列を固定, またエンティティトークンを 90% の確率でマスクして, 出力  $\hat{\mathbf{y}}$  (式 7) の対数尤度を最大化させる。その他ハイパーパラメタの探索空間は Devlin ら [11] と同じものを用いた。

表2 AIDA-CoNLL 以外のデータにおける Micro-F1 値. 最も性能が良い2つの数値に太字, 下線の順で印をつけた.

Method	MSNBC	AQUAINT	ACE2004	CWEB	WIKI	Avg.
Ganea と Hofmann (2017) [3]	93.7	88.5	88.5	77.9	77.5	88.2
Le と Titov (2018) [4]	93.9	88.3	89.9	77.5	78.0	85.5
Yamada ら (2021) [7]	<b>96.3</b>	<b>93.5</b>	<b>91.9</b>	<b>78.9</b>	<b>89.1</b>	<b>89.9</b>
<b>Ours</b>	<u>94.9</u>	<u>92.6</u>	<b>91.9</b>	<u>78.6</u>	<b>89.5</b>	<u>89.5</u>

表3 エンティティの動的な埋め込み表現を部分的に用いた時の AIDA-CoNLL における Micro-F1 値の変化.

Method	KB+YAGO	PPRforNED
<b>Ours</b>	92.3 $\Delta$	95.4 $\Delta$
w/ description-based emb.	92.2 -0.1	95.3 -0.1
w/ hyperlink-based emb.	92.0 -0.2	95.3 -0.1
<b>Ours w/ fine-tuning</b>	94.3 $\Delta$	96.9 $\Delta$
w/ description-based emb.	94.0 -0.3	96.5 -0.4
w/ hyperlink-based emb.	93.5 -0.8	96.5 -0.4

表4 未知エンティティメンションに関する Precision. また, 未知エンティティの hyperlink-based emb. を推定する際に使用出来たハイパーリンク文の数 (# sentences) によってメンションを分類し各々で precision を算出した. Baseline は被リンク事前確率 [3] 最大のエンティティを選択する.

Method	overall	# sentences		
		1-2	3-10	$\geq 11$
<b>Ours</b>	78.2	68.7	78.6	82.8
w/ description-based emb.	77.1	69.3	76.8	82.4
w/ hyperlink-based emb.	78.9	68.4	79.6	83.5
<b>Baseline: prior probability</b>	68.3	63.9	69.3	68.3

## 4.2 実験結果

AIDA-CoNLL およびその他評価データにおける実験結果を表1, 表2に示す. 提案モデルは WIKI および ACE2004 において最高性能であり, 他の評価データでは Yamada ら [7] を除く全システムに優れた. 総合して, 提案モデルの基盤である Yamada ら [7] のモデルが最も優れていたが, 提案モデルは MSNBC を除く全評価データで 1pt. 以内に迫った.

比較システム [3, 4, 5, 6, 7] と異なり, 提案モデルは評価データの約 60% の候補エンティティについてしか訓練を行なえていない. 加えて, Yamada ら [7] は, 提案モデルが基盤とする BERT<sub>base</sub> よりも大きな BERT<sub>large</sub> を基盤としている (§ 2). 提案手法がこれらシステムと同等またはそれ以上の性能を得たことは, エンティティの埋め込み表現の推定および補完が一定の水準で達成されていることを示す.

## 4.3 考察

**推定表現のアンサンブルの影響** 動的な埋め込み表現の平均化 (§ 3.2) の影響を観るため, いずれかのエンコーダで推定された埋め込み表現のみを用いて  $Y_c$  [7] の拡張を行い, AIDA-CoNLL で評価し

た (表 3). 僅かではあるが, いずれの場合も一貫して性能が低下しており, 埋め込み表現の平均化により異なる性質を補い合っていることが推測される.

**未知エンティティの曖昧性解消性能** 訓練対象外のエンティティをどの程度曖昧性解消できているのか確認するために, 全評価データ (§ 4.1) の中で, 未知エンティティ  $v_e$  をラベルに持つメンションについて precision を計算した (表 4). 動的なエンティティ埋め込み表現の種類に関わらず約 8 割の未知エンティティが正しく曖昧性解消されていた. 一方, その中でも Hyperlink-based な埋め込み表現が最も効果的であるという傾向は, 既知エンティティを含めた結果の傾向 (表 3) に反していることから, 同埋め込み表現は既知エンティティ  $V_e$  の正しい曖昧性解消を惑わせる影響もあることを示唆している.

**エンコードに使用する文の数の影響** Hyperlink-based emb. の質と, そのエンコードに使用できるエンティティを言及する文の数の関係を分析するため, その文の数に基づいて未知エンティティおよびそれらをラベルに持つメンションを大まかに分類し, それぞれで precision を計算した (表 4). 表より, エンティティを言及する文が 3 文以上利用できる場合はそこからエンコードされた埋め込みは全ての埋め込みの中で最も質のよい表現であった. 反対に, 2 文以下の場合はエンティティの頻度に依存しない説明文の利用が最も効果的であった. 今後, エンティティの頻度に応じて両埋め込み表現に重みをつけてアンサンブルするといった改善が考えられる.

## 5 おわりに

本稿では, 未知エンティティの埋め込み表現を動的に推定し, 静的な埋め込み表現に基づく曖昧性解消モデルを拡張する方法を提案した (§ 3). 実験 (§ 4) では, 提案手法を用いて拡張したモデルが, 評価データで使用される全エンティティの埋め込みをパラメタ化して訓練しているモデルらと同等またはそれ以上の性能を得られることを示した. また, 提案手法による未知エンティティの曖昧性解消への有効性と, 動的な埋め込みの品質改善の余地を示した.



## 参考文献

- [1] 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.
- [2] Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation. In **Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning**, pp. 250–259, 2016.
- [3] Octavian-Eugen Ganea and Thomas Hofmann. Deep Joint Entity Disambiguation with Local Neural Attention. In **Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing**, pp. 2619–2629, 2017.
- [4] Phong Le and Ivan Titov. Improving Entity Linking by Modeling Latent Relations between Mentions. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 1595–1604, 2018.
- [5] Samuel Broscheit. Investigating Entity Knowledge in BERT with Simple Neural End-To-End Entity Linking. In **Proceedings of the 23rd Conference on Computational Natural Language Learning**, pp. 677–685, 2019.
- [6] Thibault Févry, Nicholas FitzGerald, Livio Baldini Soares, and Tom Kwiatkowski. Empirical Evaluation of Pretraining Strategies for Supervised Entity Linking. In **Automated Knowledge Base Construction**, 2020.
- [7] Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. Global Entity Disambiguation with Pretrained Contextualized Embeddings of Words and Entities. **arXiv preprint arXiv:1909.00426**, 2021.
- [8] Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. Zero-Shot Entity Linking by Reading Entity Descriptions. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 3449–3460, 7 2019.
- [9] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. Scalable zero-shot entity linking with dense entity retrieval. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 6397–6407, 2020.
- [10] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive Entity Retrieval. In **International Conference on Learning Representations**, 2021.
- [11] 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)**, pp. 4171–4186, 2019.
- [12] Dan Hendrycks and Kevin Gimpel. Gaussian Error Linear Units (GELUs). **arXiv preprint arXiv:1606.08415v3**, 2016.
- [13] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer Normalization. **arXiv preprint arXiv:1607.06450v1**, 2016.
- [14] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust Disambiguation of Named Entities in Text. In **Proceedings of the Conference on Empirical Methods in Natural Language Processing**, pp. 782–792, 2011.
- [15] Zhaochen Guo and Denilson Barbosa. Robust Named Entity Disambiguation with Random Walks. **Semantic Web**, Vol. 9, No. 4, pp. 459–479, 2018.
- [16] Maria Pershina, Yifan He, and Ralph Grishman. Personalized Page Rank for Named Entity Disambiguation. In **Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, p. 238–243, 2015.