既知語との表層類似性に基づく未知語の埋め込み表現の計算

福田 展和 * 吉永 直樹 † 喜連川 優 †,⋆
* 東京大学大学院 情報理工学系研究科 † 東京大学 生産技術研究所 ⋆ 国立情報学研究所
{fukuda, ynaga, kitsure}@tkl.iis.u-tokyo.ac.jp

1 はじめに

現在，自然言語処理で標準的に用いられる深層学習モデルでは利用できる語彙が制限されるため，語彙に含まれない未知語 (Out of vocabulary: OOV) が問題となりやすい．特にモデルを学習する訓練データが小規模である場合や，訓練データとテストデータのドメインが異なる場合では，未知語が大量に出現しタスクの性能が低下する．

この問題に対し，大規模テキストから言語モデルなどの補助的なタスクを介して広範な語彙の単語埋め込みを事前に学習し，モデルの埋め込み層に固定して利用するアプローチが用いられる．しかし実応用では，モデルの訓練データより未来のテキストを処理したり，訓練データと異なるドメインのテキストを処理することも多く，新語や低頻度な合成語，表記揺らぎ，綴り誤りなどの多様な未知語に対処することは困難である．

そこで本稿では，そのような未知語，特に固有名詞や表記ゆれ，綴り誤りに対処するために，未知語の埋め込み表現を表層の類似する既知語の埋め込みを利用して計算する手法を提案する．提案手法は，図 1 のように対象となる未知語 (brexit) の表層に含まれる既知語 (exit) や，表層の類似する既知語 (grexit) の埋め込み表現を利用する．これらの既知語と未知語との間の表層の類似性を学習して，既知語の埋め込み表現を集約することで未知語の埋め込み表現を計算する．未知語の埋め込み表現を計算する既存手法 [1, 2] と異なり，サブワードの埋め込み表現を学習・利用せず，既知語の埋め込み表現から未知語の埋め込み表現を計算することで，サブワードから構成的に計算することが困難な固有名詞や綴り誤りなどの未知語に対して，より高精度な埋め込み表現を計算できる．

実験では，提案手法と既存手法 [1, 2] により計算される未知語の埋め込み表現を，低頻度語の類似度判定データセット [3] と綴り誤りコーパス [4] を用いた内的評価で比較し，固有名詞や綴り誤りについてより良い埋め込み表現を計算できていることを確認した．また，未知語が多く含まれるツイッタードメインの品詞タグ付けと固有表現抽出による外的評価を通して，文脈を用いた既存手法 [5, 6] とも比較を行い，特に未知語に対する分類において精度が改善することを確認した．

2 関連研究

単語埋め込みを直接学習することが困難な未知語や低頻度語の埋め込み表現を計算するために，主に表層を用いる手法と文脈を用いる手法が提案されている．

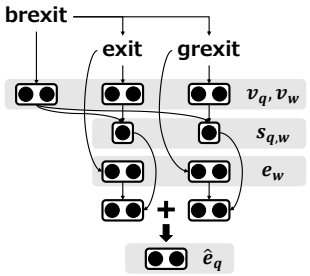

図 1: 提案手法の概要図．

表層の情報を利用する手法 [7, 1, 2] は，基本的にサブワードの埋め込み表現を学習して未知語の埋め込み表現の計算に利用する．これらの手法は単語の埋め込み表現がその単語を構成するサブワードから構成的に計算できるという仮定に基づいており，元の単語の埋め込み表現を模倣するように文字やサブワードの埋め込み表現を学習する．例えば，Zhao ら [1] は既知語の埋め込み表現をサブワードの埋め込み表現の和として再構成するようにサブワードの埋め込み表現を学習し，得られたサブワードの埋め込み表現を未知語の埋め込み表現の計算に用いる手法を提案している．ここで，サブワードの埋め込み表現は，そのサブワードを含む全ての単語の意味を復元するように学習される．そのため，単語の埋め込み表現と比較してサブワードの埋め込み表現は多義性が高くなり単独のベクトルで多様な意味を捉えることは難しい．さらに，埋め込み表現を計算する上で重要なサブワードに綴り誤りや表記揺れがある場合には，埋め込み表現の品質が大きく悪化する可能性がある．

これらの問題に対処するために，提案手法はサブワード埋め込みを用いず，既知語の埋め込み表現から直接未知語の埋め込み表現を計算する．本稿では，上述のサブワード埋め込みを用いて未知語の埋め込み表現を再構成する手法をベースラインとして比較した．

文脈の情報を用いる手法 [5, 6] は，低頻度な単語が出現する限られた文脈の情報から高精度な埋め込み表現を学習する．これらの手法を用いることで文脈を数件程度しか利用できないような低頻度語であっても比較的高性能な埋め込み表現を計算できる．しかし，実際には単語の意味を詳細に類推することが可能な文脈が利用できるとは限らない，これらの手法は表層の情報を利用する手法と組み合わせることができる [5, 6] ため，本稿では両モデルを組み合わせる設定について外的評価を行った．

3 提案手法

本節では未知語の埋め込み表現を計算する提案手法について述べる．本手法は，既知語の埋め込み表現の線形和によって未知語の埋め込み表現を計算する．各既知語の埋め込みを足し合わせる際の重みは既知語との表層の類似性に基づいて計算する．まず，未知語に対して表層の類似する既知語を抽出する手法として，(i) 既知語による分割 (seg) と (ii) 類似文字列検索 (approx) について述べる．

既知語による分割 未知語に含まれる既知語を抽出して利用する．まず，単語の表層を複数の既知語もしくは文字に分割する．次に，分割数が最小となる分割列に含まれる既知語を文字列が長い順に n^{seg} 個抽出する．分割数が最小となる分割列が複数存在する場合には，各分割列に含まれる既知語から同様に抽出する．単語として埋め込み表現が学習された既知語はサブワード埋め込みより意味が一義的であると期待できる．

類似文字列検索 未知語と表層の特徴量が類似する単語を既知語から抽出して利用する．単語中の文字 3-gram の集合を表層の特徴量として，特徴量の Jaccard 係数を単語間の表層の類似度とする．未知語に対して，表層類似度の高い順に既知語を n^{approx} 個抽出する．この類似文字列検索には simstring [8] を用いた．

次に，上述の 2 つの方法で得られた既知語の埋め込みを利用して未知語の埋め込み表現を計算する方法について述べる．未知語 q に対して，既知語による分割と類似文字列検索によって得られた既知語をそれぞれ $w_i^{\text{seg}}, w_i^{\text{approx}}$ とする．この既知語と未知語の表層ベクトル $\boldsymbol{v}_w$ を文字 CNN を用いて計算する．次に，$\boldsymbol{v}_w$ から未知語 q と既知語 w の間の表層の類似度 $s_{q,w}$ を計算する．その後，$s_{q,w}$ を重みとして各既知語の埋め込み $\boldsymbol{e}_w$ を足し合わせて $\boldsymbol{e}_q^{\text{seg}}, \boldsymbol{e}_q^{\text{approx}}$ を計算する．ここで $\boldsymbol{W}_k$ ($k \in \{\text{seg}, \text{approx}\}$) は学習により推定するパラメータである．

$$s_{q,w_i^k} = \text{softmax}\left(\boldsymbol{v}_q^{\mathsf{T}} \cdot \boldsymbol{W}_k \cdot \boldsymbol{v}_{w_i^k}\right) \tag{1}$$

$$\boldsymbol{e}_q^k = \sum_i^{n^k} s_{q,w_i^k} \boldsymbol{e}_{w_i^k} \tag{2}$$

最後に，$\boldsymbol{e}_q^{\text{seg}}, \boldsymbol{e}_q^{\text{approx}}$ の重み付き線形和を計算して未知語 q の埋め込み表現 $\hat{\boldsymbol{e}}_q$ とする．ここで $\boldsymbol{\theta}_k$ は学習により推定するパラメータである．

$$\alpha_k = \text{softmax}\left(\boldsymbol{\theta}_k \cdot \boldsymbol{s}_{q,w_i^k}\right) \tag{3}$$

$$\hat{\boldsymbol{e}}_q = \alpha_{\text{seg}} \boldsymbol{e}_q^{\text{seg}} + \alpha_{\text{approx}} \boldsymbol{e}_q^{\text{approx}} \tag{4}$$

上述のように計算した埋め込みと既知語の埋め込みのコサイン類似度を損失関数としてモデルを訓練する．

4 実験

本節では未知語の埋め込み表現を評価するため行った内的評価と外的評価について順に説明する．なお，内的評価と外的評価において既知語の単語埋め込みには **GloVe**[1] [9] の埋め込み表現を用いた．提案手法において，未知語と表層の類似する既知語を抽出する際に $n^{\text{seg}} = 10, n^{\text{approx}} = 10$ とした．また，表層の類似度を計算するための文字埋め込みの次元は 100 とし，文字 CNN のフィルタサイズは 2,4,6,8 とした．

	CARD ALL	CARD OOV	TOEFL
GloVe [9]	27.3	-	-
BoS [1]	37.3	18.0	11.6
KVQ-FH [2]	45.5	28.8	10.9
提案手法	**48.3**	**37.1**	**36.2**

表 1: 未知語の埋め込み計算手法の内的評価．

4.1 内的評価

内的評価として低頻度語の類似度判定タスクと綴り誤りの埋め込み評価タスクを通して未知語の埋め込み表現の性能を評価した．以下で実験設定，比較手法，実験結果について順に述べる．

実験設定 低頻度語類似度判定タスクには CARD-660 データセット [3] (CARD) を用いた．CARD はコンピュータサイエンスやソーシャルメディア，生体医学などのドメインから収集された 1306 個の単語からなる 660 組の単語対から構成されており，各単語対に人手で類似度が付与されている．この類似度と単語対の埋め込み表現のコサイン類似度との相関をスピアマンの順位相関係数で評価した．なお，既知語に関しては学習済みの埋め込み表現を用い，未知語については各手法で埋め込み表現を計算して評価を行った．

綴り誤りの埋め込みの評価には英語学習者の作成したエッセイのコーパス [4] (TOEFL) を利用した．具体的には，TOEFL データセットから綴りの正しい単語 (既知語) と綴りの誤った単語 (未知語) の組を 1514 組抽出し，これを用いて評価を行った．各手法を用いて綴り誤りの単語の埋め込みを計算し，正しい綴りの単語の埋め込みとのコサイン類似度を評価した．

なお，内的評価において各手法が未知語の埋め込みを計算できない場合には，既存研究 [10] に倣ってその単語を含む単語対の類似度を 0 として計算した．

比較手法 以下に示すベースラインとの比較を行った．

GloVe [9] 既知語に学習済みの埋め込み表現[1]を利用し，未知語を含む単語対については類似度を 0 とする．

BoS [1] **GloVe** の語彙に含まれる n-gram に対してサブワード埋め込みを学習しておき，未知語の埋め込み表現は未知語に含まれるサブワードの埋め込み表現の平均として計算する．

KVQ-FH ($F = 1\text{M}, H = 0.5\text{M}$) [2] **BoS** と同様にサブワード単位の埋め込みを学習して用いるが，サブワード間で埋め込みを共有する点と埋め込みの集約時に Self-attention を計算する点が異なる．

実験結果 実験結果を表 1 に示す．表中の CARD における ALL は全ての単語対での評価であり，OOV は

[1] https://nlp.stanford.edu/projects/glove

	表層	文脈	ARK		T-POS		Rare-NER		Multi-NER	
			ALL	OOV	ALL	OOV	ALL	OOV	ALL	OOV
Single UNK			83.6	55.5	81.7	56.7	37.7	6.7	70.4	29.1
BoS	✓		84.3	61.1	81.2	59.9	**39.2**	8.2	70.6	34.8
KVQ-FH	✓		84.4	61.5	81.2	60.9	37.6	6.1	70.5	35.5
提案手法	✓		**86.1**	**76.2**	**82.3**	**71.7**	37.5	**12.5**	**71.2**	**41.4**
HiCE (文脈のみ)		✓	82.0	60.4	81.4	62.9	34.6	3.6	69.3	34.1
BoS	✓	✓	84.2	61.6	81.1	58.4	**38.7**	6.9	**70.9**	33.4
KVQ-FH	✓	✓	84.2	61.6	81.4	60.1	37.9	7.0	70.8	34.6
提案手法	✓	✓	**85.9**	**74.8**	**82.3**	**71.2**	38.2	**9.7**	70.8	**39.9**

表 2: 未知語の埋め込み計算手法の外的評価.

未知語を含む単語対に限定した評価である．どちらのデータセットにおいても既存手法と比較して提案手法の性能が向上した．特に，提案手法によって綴り誤りにより頑健な単語埋め込みが計算できることは注目に値する．

4.2 外的評価

次に，外的評価として未知語の埋め込み表現の品質の下流タスクへの影響を品詞タグ付けと固有表現抽出を通して評価した．以下で実験設定，比較手法，実験結果について順に述べる．

実験設定 品詞タグ付けにはツイートに品詞が付与された ARK [11]，及び T-POS [12] データセットを用いた．品詞タグ付けのモデルには [7] と同様のものを利用し，単語単位の分類精度を評価した．

固有表現抽出にはツイートに固有表現が付与された Rare-NER [13]，及び Multi-NER [14] データセットを用いた．固有表現抽出のモデルには LSTM-CRF [15] を利用し，エンティティ単位の F1 値を評価した．

外的評価では，内的評価の比較手法である表層の情報を利用するモデルに加えて，文脈の情報を用いるモデルと組み合わせたモデルの評価も行った．文脈を利用するモデルでは，未知語の表層と未知語が出現する文脈を入力して，未知語の埋め込み表現を推定する Few-shot のタスク設定を考える．文脈情報のエンコードには [5] と同様のモデルを用いた．外部テキストコーパスに出現する単語のうち，埋め込みの存在する既知語であるものを利用して表層と文脈のモデルを訓練した．外部テキストコーパスには Wikitext-103 [16] を用いた．テスト時には Wikitext とテストデータを含むタスクのテキストデータを文脈として，未知語が出現する文脈を抽出して未知語の埋め込み表現を計算した．下流タスクの訓練時には，計算した単語埋め込みを固定してタスクのモデルを学習し，評価を行った．表層のみ，文脈のみを用いるモデルも同じ訓練データで同様の学習を行った．

比較手法 内的評価で述べた手法に加えて以下に示すベースラインとの比較を行った．

Single UNK 未知語に単一の未知語ベクトル割り当て，タスクと同時に訓練する．

HiCE (文脈のみ) [5] Transformer のエンコーダブロックを用いて複数の文脈をエンコードする．[5] と異なり，文脈のモデルのみを利用した．

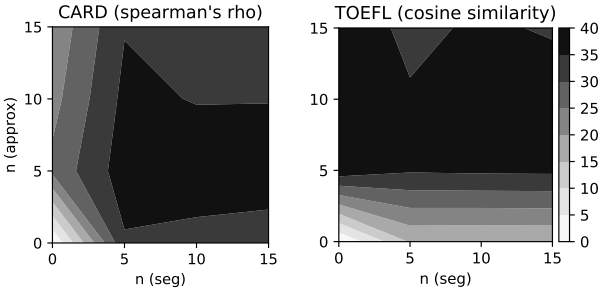

図 2: 感度分析の結果.

実験結果 実験結果を表 2 に示す．表中の ALL は全単語対での評価である，OOV は品詞タグ付けにおいては，未知語に限定して品詞を評価した性能を表し，固有表現抽出においては，単語の中に未知語を含むエンティティに限定して評価した性能を表す．単独の未知語埋め込みを利用する **Single UNK** と比較して，表層や文脈を利用するモデルの OOV の性能が Rare-NER 以外の 3 つのデータセットで向上した．これは下流タスクにおいても未知語埋め込みを計算することで性能が向上することを示している．また，表層を利用するモデルにおいて，提案手法の OOV の性能が既存手法から向上した．これは文脈を組み合わせるモデルにおいても同様の傾向が見られた．**Single UNK** と比較して文脈のみを用いるモデルの性能は向上しているが，表層モデルを組み合わせたときに文脈モデルによる効果は見られなかった．これは，文脈の情報の学習に用いた Wikitext のドメインが下流タスクのツイッターのドメインと異なることや，ツイート中に出現する表記揺らぎ等の未知語について得られる文脈の数が少ないことなどが原因と考えられる．特に後者について，外的評価に用いた 4 つのデータセットを通して，利用できる未知語の文脈数の中央値は 1 であった．

5 考察

本節では提案手法の計算する未知語埋め込みについて詳細に分析を行う．

まず，提案手法のハイパーパラメータの感度分析について述べる．提案手法の既知語による分割と類似文字

単語	手法	コサイン類似度	埋め込み表現の近傍の既知語				
espically	GloVe	100	espically	exspecially	especally	espeically	espescially
	BoS	48	budgetarily	similarily	charily	particualrly	palatably
	提案手法	77	especally	espescially	esecially	escpecially	epsecially
LANs	GloVe	100	LANs	WANs	WLANs	VPNs	SANs
	BoS	78	LANs	WANs	WLANs	MANs	VLANs
	提案手法	54	LAN	WAN	WLAN	ETHERNET	LANs

表 3: 提案手法の出力例.

	ALL	固有名詞	その他
BoS	25.2	31.4	23.9
KVQ-FH	33.5	37.9	31.5
提案手法	38.6	58.6	33.5

表 4: 単語のタイプ別の CARD の結果.

列検索において，それぞれ $n^{\text{seg}}, n^{\text{approx}} \in \{0, 5, 10, 15\}$ を変化させて，CARD の未知語を含む単語対 (表 1 の OOV) と TOEFL で評価した結果を図 2 に示す．図 2 の CARD の結果から既知語による分割が CARD に有効であることが示された．これは CARD に含まれる合成語に対して既知語による分割が効果的であるためと考えられる．また，図 2 の TOEFL の結果から類似文字列検索を利用することによって綴り誤りに対して頑健になることが示された．既知語による分割で抽出する単語数が多いほどより短い単語を考慮でき，類似文字列検索で抽出する単語数が多いほどより表層の離れた単語を考慮できる．しかし，参照する単語数が多いとモデルの学習・推論コストを増大させるため，これらはトレードオフの関係にあると考えられる．

次に，内的評価における低頻度語を分類して各手法の性能を比較した．各手法の性能を CARD のデータにおいて未知語のタイプ別に分析した結果を図 4 に示す．まず，CARD のデータに含まれる単語対について，片方のみの単語が **GloVe** の未知語である 205 組を抽出し，未知語について固有名詞とその他とに単語対を分類した．次に，各手法によって未知語の埋め込み表現を計算し，既知語の埋め込みとのコサイン類似度を評価した．結果としてサブワード埋め込みを利用する手法と比較して，提案手法によって固有名詞の性能が向上した．このことから，形態素と意味の関係が薄い固有名詞などの未知語には既知語の埋め込み表現を直接利用する手法が有効であることが示された．

最後に，計算された埋め込みの出力例の分析について述べる．提案手法によって計算された埋め込み表現の近傍の単語を **BoS** と比較した．**GloVe** の既知語の表層を基に，**BoS** と提案手法によって計算した埋め込み表現の元の埋め込み表現とのコサイン類似度と，計算した埋め込み表現の近傍の既知語を表 3 に示す．1 つ目の例において単語 `espically` に対して，提案手法が **BoS** より元の単語に近い埋め込みを計算できた．これは **BoS** の計算する埋め込み表現が接尾辞の `ly` に影響を受けてコサイン類似度が低下したと考えられる．2 つ目の例において単語 `LANs` に対して，提案手法が **BoS** より元の単語から離れた埋め込みを出力している．これは接尾辞の `s` が単語の意味を大きく変えないと提案手法が学習したためと考えられる．このような単数形と複数形の混同は品詞タグ付け等の構文的情報を利用するタスクにおいて悪影響を及ぼすと考えられる．これらを踏まえて，合成語や固有名詞に統一的に適用できる手法を開発することが今後の課題である．

6 おわりに

本稿では表層の類似度を推定するモデルを学習して未知語の埋め込み表現を計算する手法を提案した．さらに，内的評価と外的評価を行い，既存手法と提案手法の性能を実験的に比較し，サブワード埋め込みを利用する手法と同程度の性能を達成した．今後の課題としてサブワードによる分かち書きを利用するモデルへの適用が挙げられる．

謝辞 この研究の一部は，2019 年度国立情報学研究所 CRIS 委託研究，および JST，CREST，JPMJCR19A4 の支援を受けています．

参考文献

[1] J. Zhao, S. Mudgal, Y. Liang. Generalizing word embeddings using bag of subwords. In *EMNLP*, 2018.

[2] S. Sasaki, J. Suzuki, K. Inui. Subword-based compact reconstruction of word embeddings. In *NAACL-HLT*, 2019.

[3] M. T. Pilehvar *et al.* Card-660: Cambridge rare word dataset - a reliable benchmark for infrequent word representation models. In *EMNLP*, 2018.

[4] M. Flor, M. Fried, A. Rozovskaya. A benchmark corpus of english misspellings and a minimally-supervised model for spelling correction. In *BEA@ACL*, 2019.

[5] Z. Hu *et al.* Few-shot representation learning for out-of-vocabulary words. In *ACL*, 2019.

[6] M. Peng *et al.* Learning task-specific representation for novel words in sequence labeling. In *IJCAI*, 2019.

[7] Y. Pinter, R. Guthrie, J. Eisenstein. Mimicking word embeddings using subword rnns. In *EMNLP*, 2017.

[8] N. Okazaki, J. Tsujii. Simple and efficient algorithm for approximate dictionary matching. In *COLING*, 2010.

[9] J. Pennington, R. Socher, C. D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[10] Z. Yang *et al.* Embedding imputation with grounded language information. In *ACL*, 2019.

[11] K. Gimpel *et al.* Part-of-speech tagging for twitter: Annotation, features, and experiments. In *ACL*, 2011.

[12] A. Ritter *et al.* Named entity recognition in tweets: An experimental study. In *EMNLP*, 2011.

[13] L. Derczynski *et al.* Results of the wnut2017 shared task on novel and emerging entity recognition. In *NUT@EMNLP*, 2017.

[14] Q. Zhang *et al.* Adaptive co-attention network for named entity recognition in tweets. In *AAAI*, 2018.

[15] G. Lample *et al.* Neural architectures for named entity recognition. In *HLT-NAACL*, 2016.

[16] S. Merity *et al.* Pointer sentinel mixture models. In *ICLR*, 2017.