Vol.2018-NL-236 No.1
2018/7/9

分散表現のファインチューニングによる語義曖昧性解消の領域適応

柳沼 大輝[,a)]　古宮 嘉那子[,b)]　新納 浩幸[,c)]

概要：概要：本論文では語義曖昧性解消の領域適応を扱う。質の高い分散表現を用いることで、語義曖昧性解消の精度が向上することが知られている。しかし質の高い分散表現を使用した語義曖昧性解消であっても領域が変化することによる悪影響を受ける。これが領域シフトの問題である。そこでターゲット領域のコーパスを利用して、分散表現をターゲット領域にファインチューニングすることで、語義曖昧性解消の領域シフトの問題に対処する。実験では分散表現として nwjc2vec 及び日本語 Wikipedia から構築した分散表現 (wiki2vec) を用いる。領域としては「Yahoo! 知恵袋 (Y)」「新聞 (N)」「白書 (W)」の 3 つを設定し、Y → N、Y → W、N → Y、N → W、W → Y、W → N の 6 通りの領域適応を行い、ターゲット領域別の平均正解率によって手法を評価する。語義曖昧性解消は各領域に共通して出現する 36 単語を対象とした。実験の結果、比較的小さなターゲットデータに対してはファインチューニングの効果が確認できた。

キーワード：語義曖昧性解消、領域適応、分散表現、ファインチューニング

WSD of domain adaptation by distributed representation of fine tuning

Yaginuma Daiki[,a)]　Komiya Kanako[,b)]　Shinnou Hiroyuki[,c)]

1. はじめに

複数の語義を持つ単語を多義語といい、文中に存在する多義語の語義を識別する処理を語義曖昧性解消という。語義曖昧性解消は意味解析の基盤要素技術であり、その重要性は高い。語義曖昧性解消は一般に教師あり学習により解決できるが、その場合、領域シフトの問題が生じることが知られている。領域シフトの問題とは、教師あり学習において、モデルの学習元となる訓練データの領域（ソース領域）と、学習できたモデルを適用するテストデータの領域（ターゲット領域）が異なる場合に、モデルの精度が悪くなる問題である。領域シフトの問題に対する手法が領域適応である。

ここでは語義曖昧性解消の手法として、分散表現を利用する。質の高い分散表現を利用することで語義曖昧性解消の精度が向上することが知られているが [16]、質の高い分散表現であっても領域シフトの問題が生じることも指摘されている [13]。そこで本論文ではターゲット領域のコーパスを利用して、分散表現をファインチューニングし、そのファインチューニングした分散表現を利用して語義曖昧性解消を行うことで、語義曖昧性解消の領域適応を行う。ただし分散表現をファインチューニングする際には、ターゲット領域のコーパスが存在するので、ファインチューニングを行わずに、そのコーパス自身から分散表現を作成することも考えられる。大規模なコーパスが利用できる場合には、ファインチューニングする効果はないが、小規模な

[1] 情報処理学会
IPSJ, Chiyoda, Tokyo 101–0062, Japan
[2] 茨城大学大学院理工学研究科情報工学専攻
Major in Computer and Information Sciences,Graduate School of Science and Engineering, Ibaraki University
[†1] 現在，茨城大学工学部情報工学科
Presently with Department of Computer and Information Sciences,College of Engineering, Ibaraki University
[a)] 18nm740n@vc.ibaraki.ac.jp
[b)] kanako.komiya.nlp@vc.ibaraki.ac.jp
[c)] hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

コーパスしか利用できない場合には、ファインチューニングの意味があることも示す。

2. 関連研究

領域適応の手法はターゲット領域のラベル付きデータを利用するかしないかで教師ありと教師なしに分類される。教師ありの場合は Daume の手法 [1] が簡易で性能も高いことから標準手法となっている。また教師なしの場合は事例ベースの手法と素性ベースの手法とに分けられる [7]。

事例ベースの手法は共変量シフト下の学習に帰結される。Shinnou は共変量シフト下の学習により語義曖昧性解消の領域適応を行った [8][15]。また素性ベースの手法は基本的にデータの特徴ベクトルをターゲット領域に適した形に変換する手法であり、非常に多くの研究がある。語義曖昧性解消に関してはトピックモデルを利用したものがある [14]。またディープラーニングを用いた手法も素性ベースの手法と見ることもできる。語義曖昧性解消に関しては AutoEncoder を利用したものがある [3]。近年では素性ベースの手法として CORAL と呼ばれる手法 [10] が注目され、それをネットワークによる学習に拡張した Deep CORAL [11] もある。また敵対性ネットワークを領域適応に利用する研究も行われている [2][12] 。

また、分散表現を語義曖昧性解消に利用した研究は Sugawara の研究がある [9]。これは単に通常の素性ベクトルに対象単語の前後 2 単語の分散表現を連結するという簡易なものであるが、質の高い分散表現を利用することでかなり精度が向上するという報告がある。

Komiya は分散表現の構築元になるコーパスの領域に注目し、分散表現を語義曖昧性解消の領域適応に利用した。ベースとなる分散表現の作成元のコーパスとターゲット領域のコーパスとの組み合わせて分散表現を構築した場合の有効性を論じている [4]。

本研究は上記研究の改良版と見なせる。ベースとなる分散表現をターゲット領域のコーパスを用いてファインチューニングすることでターゲット領域に合った分散表現を構築する。

3. 提案手法

3.1 分散表現を利用した語義曖昧性解消

語義曖昧性解消で利用する素性の種類は以下の 6 種類である。

(1) 対象単語と前後 2 単語の表記
(2) 対象単語と前後 2 単語の品詞
(3) 対象単語と前後 2 単語の品詞の細分化
(4) 係り受け
(5) 前後 2 単語の 5 桁の分類コード
(6) 前後 2 単語の 4 桁の分類コード

ここで分類コードとは分類語彙表の語彙番号である。レコードを構成する項目は「レコード ID 番号／見出し番号／レコード種別／類／部門／中項目／分類項目／分類番号／段落番号／小段落番号／語番号／見出し／見出し本体／読み／逆読み」という項目で構成されている。分類語彙表では、単語が分類番号によって単分類されており、この分類番号は、単語の「類・部門・中項目・分類項目」を表したものである。

(1),(2),(3) からは各々 5 つの素性ができる。(5),(6) からは各々 4 つの素性ができる。以上より合計 24 個の素性ができる。各素性は one-hot-vector で表現され、それらが連結されて素性ベクトル v が構築される。

分散表現は対象単語の前後 2 単語の分散表現 (b1; b2; f1; f2) を利用する。先の素性ベクトル v にこの 4 つのベクトル b1; b2; f1; f2 を連結することで、最終的な素性ベクトルを構築する。

上記の分散表現として、オリジナルの分散表現に代えて、ファインチューニングした分散表現を用いるのが提案手法である。

3.2 分散表現のファインチューニング

本論文では分散表現を構築する手法として word2vec[5][6] を用いる。

ファインチューニングの方法としては既存の分散表現を word2vec の初期値として与え、新たなコーパスを使って word2vec を走らせるという簡易な手法を採る。

4. 実験

4.1 実験設定

領域として「YAHOO!知恵袋 (Y)」、「新聞 (N)」、「白書 (W)」の 3 つを用いる。つまり Y → N, Y → W, N → Y, N → W, W → Y および W → N の 6 通りの領域適応が行われるが、ターゲット領域毎の平均正解率によって手法を評価する。

各領域におけるコーパスのサイズ（単語数）を表 1 に示す。

表 1 使用データの語彙数

領域	データ (単語数)
YAHOO!知恵袋	11692780
新聞	30544759
白書	5130578

またファインチューニングの際に word2vec で利用する各パラメータは表 2 のように設定した。

表 2　word2vec の主要なパラメータ

意味	変数	パラメータ
次元数	-unit	200
ウィンドウ幅	-window	5
バッチサイズ	-batchsize	1000
ネガティブサンプリング数	-negative-size	5
反復回数	-epoch	10
選択手法	-model	cbow

また語義曖昧性解消の対象となる単語は各領域である程度の出現頻度を持つ以下の 36 単語である。

「聞く」「技術」「経済」「現在」「合う」「言う」「子供」「時間」「自分」「社会」「情報」「進む」「進める」「今」「入れる」「高い」「出す」「地方」「手」「出る」「取る」「乗る」「場合」「入る」「計る」「一つ」「開く」「外」「前」「見る」「持つ」「大きい」「良い」「上げる」「考える」「関係」

各単語に各領域毎の事例数を表 3 以下に示す。ソース領域になる場合は、この事例が訓練事例となり、ターゲット領域となる場合はテスト事例となる。

4.2　実験結果

ここでの実験データは Komiya らの先行研究 [4] と同一である。先行研究と実験結果の比較のために、実験結果はターゲット領域に注目してまとめることにする。つまり領域適応は 6 種類があるが、以下の 3 つに分け、その中の 2 つの領域適応の正解率の平均値により正解率を算出する。

新聞 : Yahoo!→新聞 と BCCWJ →新聞　の平均
Yahoo! : 新聞→ Yahoo! と BCCWJ → Yahoo!　の平均
BCCWJ : Yahoo!→ BCCWJ と 新聞→ BCCWJ　の平均

また先行研究 [4] ではターゲット領域のコーパスから word2vec で分散表現を構築し、それを語義曖昧性解消に利用するという手法も試している。この手法を先行研究に倣い Add Target と名付ける。

実験結果を表 4 と表 5 に示す。なお表中の Add wiki や Add nwjc2vec はそれらの分散表現を利用した手法を表し、Add f- wiki と f-nwjc2vec は wiki2vec と nwjc2vec をファインチューニングし利用した手法を表す。ファインチューニングの効果があったものは太字にしている。

wik2vec ではファインチューニングの効果はなかったが、nwjc2vec ではファインチューニングの効果が確認できた。

分散表現のファインチューニングを行う場合、ターゲット領域のコーパスが必要である。そのコーパスが十分大きければ、ファインチューニングを行わず、そのコーパス自体から分散表現を構築してそれを語義曖昧性解消に利用することも考えられる。先の実験の Add Target はその結果を示しており、提案手法よりも良い結果を出している。ただし現実的にはターゲット領域のコーパスが大規模に収集できるとは限らない。そこで表 6 に示す小規模コーパスを利用して、先の実験を行ってみた。実験結果を表 7 と表 8 に示す。なお、そもそも分散表現の追加を行わない Baseline 及び、既に作成された分散表現を用いる Add wiki、Add nwjc2vec は比較のため掲載する。

表 3　対象単語の事例数

対象単語	Yahoo	新聞	白書
聞く	5229	275	140
技術	250	125	7490
経済	158	454	5267
現実	1177	209	3750
合う	1726	120	95
言う	17387	1046	1733
子供	3268	342	582
時間	2265	218	1758
自分	6657	297	257
社会	344	267	3698
情報	1170	215	5747
進む	299	127	919
進める	274	157	3160
今	10114	297	234
入れる	3549	149	148
高い	1978	173	4019
出す	2208	259	173
地方	270	171	2448
手	1206	151	60
出る	6637	490	185
取る	2318	241	927
乗る	1775	92	65
場合	6630	234	3311
入る	3314	301	671
計る	255	109	8691
一つ	810	92	651
聞く	654	222	182
外	2614	263	4347
前	4216	305	594
見る	7101	859	2580
持つ	2730	331	923
大きい	1206	166	2514
良い	5403	280	174
上げる	1200	222	1002
考える	3954	310	2179
関係	1974	390	6215

表 6　追加実験データの単語数

データ	単語数
YAHOO!知恵袋	1000050
新聞	1000002
白書	1000045

表 4　実験結果（マクロ平均）

ターゲット領域	Baseline	Add Target	Add wiki	Add nwjc2vec	Add f-wiki	Add f-nwjc
新聞	0.7884	0.7892	0.7919	0.7700	0.7811	**0.7736**
Yahoo!知恵袋	0.7401	0.7458	0.7445	0.7296	0.7368	**0.7326**
白書	0.8086	0.8215	0.8215	0.8036	0.8152	**0.8086**

表 5　実験結果（マイクロ平均）

ターゲット領域	Baseline	Add Target	Add wiki	Add nwjc2vec	Add f-wiki	Add f-nwjc2vec
新聞	0.7815	0.7843	0.7852	0.7673	0.7758	**0.7712**
Yahoo!知恵袋	0.7659	0.7730	0.7710	0.7586	0.7692	**0.7629**
白書	0.8470	0.8417	0.8316	0.8316	**0.8422**	**0.8363**

5. 考察

初めに、nwjc2vec をもとにファインチューニングを行い分散表現を追加した手法の結果より考察を行う。前章の実験結果より、nwjc2vec の手法より全体的に良い結果が得られているものの、ベースラインや、Add Target より低い結果となっている。本手法の結果より Add Target の結果のほうが良い結果が得られていることから、ファインチューニングを行い作成した分散表現が、ファインチューニングを行わず直接ターゲットデータから作成した分散表現より精度が下がっていることがわかる。また、f-nwjc2vec より本手法の結果のほうが良い結果を得られていることから、出現する単語の語義に大きく差がある分散表現を使用するよりは、ファインチューニングにより、ターゲットデータに対して適応させた分散表現を使用する方が精度が上昇することがわかる。

個々結果で見た際、Add Target と本手法のうち、新聞の差がマクロ平均、マイクロ平均ともに最も大きく、白書の差がマクロ平均、マイクロ平均ともに最も小さかった。また、実験に使用したターゲットデータのサイズは白書が最も少なく、新聞が最も大きなデータであったことから、ターゲットデータが大きなデータである場合、ファインチューニングを行わず分散表現を作成したほうが、分散表現の精度は上がるのではないかと考えられる。また、Add Target の結果が nwjc2vec に比べ有意に高いことから、大きなデータでファインチューニングを行えば、出現する単語の語義に差があったとしても、その差は縮まるものと考えていた。しかし、同様に nwjc2vec と本手法を比較を行うと、データサイズの大きな新聞の結果が差は小さく、データサイズの小さな白書の差がマクロ平均、マイクロ平均ともに最も差が大きかった。これの明確な原因は不明だが、元々ベースラインの白書の結果と nwjc2vec の白書の結果の差は小さいことから、他のデータに比べて、出現する単語の語義が近く、改善が容易だったのではないかと考えられる。逆に、ベースラインの新聞の結果と nwjc2vec の新聞の結果の差は大きく、出現する単語の語義が遠かったため、改善が難しかったのではないかと考えられる。

次に、wiki2vec の手法に関する結果の考察を行う。全体的に wiki2vec や、Add Target より低い結果となっている。ベースラインも、白書が部分的に高いものの、やはり全体的に低い結果となっている。f-nwjc2vec では nwjc2vec の結果が元々低かったため、精度を上昇させることができた。こちらの手法では、wiki2vec の結果が元々高いため、精度を上昇させることが難しかったものと考える。また、このことから nwjc2vec とは逆に、出現する単語の語義の差が小さい場合、ファインチューニングを行っても、その精度向上には繋がらないことが分かった。

最後に、ターゲットデータのサイズを減らした実験を行い考察を行う。

基本的に、小さいターゲットデータで分散表現を作る場合、ターゲットデータのみで直接作成するよりも、大規模データの分散表現を用いてファインチューニングを行った方が結果が良かった。これは当初の予想通り、小さいデータでは十分な学習が行われなかったためであると考えられる。しかし、nwjc2vec を元に新聞のデータで学習を行った結果のみ直接分散表現を作成する手法のほうが精度が良かった。元々 nwjc2vec の手法は baseline よりも精度が下がっており、逆に Add Target の精度は上がっていることからも、今回の新聞のように出現する単語の語彙に大きく差がある場合、かえって精度を下げてしまうのではないかと考えられる。

以上より、本手法は、大規模データに出現する単語の語義がターゲットデータに出現するものと大きく違う場合には、ファインチューニングを行うことで、分散表現の改良を行うことができ、精度を上げることができた。しかし、単にターゲットデータをもとに分散表現を作成し、追加する手法程の精度を上げることができなかった。これにはいくつかの理由が考えられるが、その一つはターゲットデータのサイズが十分に大きく、精度の良いモデルが作成できていたのではないかと考えた。そこで行った追加実験の結果により、ターゲットデータのサイズが不十分な量であるならばファインチューニングを行うことで、精度を向上させることができた。しかし、出現する単語の語彙に大きな差がある場合は例外的にファインチューニングを行わず、

表 7　小規模コーパスによる実験結果（マクロ平均）

ターゲットデータ	Baseline	Add Target	Add wiki	Add nwjc2vec	Add f-wiki	Add f-nwjc2vec
新聞	0.7884	0.7782	0.7919	0.7700	0.7826	0.7681
Yahoo!知恵袋	0.7401	0.7255	0.7445	0.7296	0.7298	0.7274
白書	0.8086	0.8000	0.8215	0.8036	0.8087	0.8042

表 8　小規模コーパスによる実験結果（マイクロ平均）

ターゲットデータ	Baseline	Add Target	Add wiki	Add nwjc2vec	Add f-wiki	Add f-nwjc2vec
新聞	0.7815	0.7751	0.7852	0.7673	0.7799	0.7639
Yahoo!知恵袋	0.7659	0.7560	0.7710	0.7586	0.7600	0.7572
白書	0.8470	0.8383	0.8316	0.8316	0.8450	0.8383

そのまま分散表現を作成する方が精度が高くなることがわかった。このことからも十分に大きなコーパスを利用できる場合はファインチューニングを行わず分散表現を作成したほうがよいと考えられるが、その十分に大きなコーパスが利用できない場合はファインチューニングを行うことで精度の良い分散表現を作成することができることがわかった。今後の課題として、先行研究でも使用されている手法である、複数の分散表現を組み合わせる実験や、対象データに合わせたパラメータの変更など、従来の手法としての強化や、小さなコーパスしか手に入らない領域での実験などを課題とし今後も検証していきたい。

6. おわりに

本論文では語義曖昧性解消の領域適応において、文章から作成した分散表現を追加し、性能を向上させる実験において、作成する分散表現をファインチューニングによって作成し精度の向上を図った。

結果として Add Target や wiki2vec 等の結果を超えることができなかったが、nwjc2vec の結果を超えることができていた。そのことから元々精度の低い分散表現であるならば、ファインチューニングによって精度を向上させることができることがわかった。また、ファインチューニングという手法の性質を考慮し、ターゲットデータのサイズを減らした実験を行ったところ、その精度を向上させることができた。このことから、ターゲットデータのサイズが不十分な量であるなら、ファインチューニングによる分散表現の作成が有効であることがわかった。先行研究の結果を超えることはできなかったが、分散表現の改良そのものには成功していると考えられる。先行研究でも使用されている手法である、複数の分散表現を組み合わせる実験や、対象データに合わせたパラメータの変更など、他の分散表現の改良手法等について今後も検証を行っていきたい。

参考文献

[1] Daume III, Hal: Frustratingly Easy Domain Adaptation, ACL-2007, pp. 256-263 (2007).
[2] Ganin, Y. and Lempitsky, V. S.: Unsupervised Domain Adaptation by Backpropagation, ICML, pp. 1180-1189 (2015).
[3] Kazuhei Kouno, Hiroyuki Shinnou, M. S. and Komiya, K.: Unsupervised Domain Adaptation for Word Sense Disambiguation using Stacked Denoising Autoencoder, PACLIC-29, pp. 224-231 (2015).
[4] Komiya, K., Suzuki, S., Sasaki, M. and Shinnou, H.: Domain Adaptation for Word Sense Disambiguation Using Word Embeddings, CICLING-2017, p. No57 (2017).
[5] Mikolov, T., Chen, K., Corrado, G. and Dean, J.: Efcient Estimation of Word Representations in Vector Space, ICLR Workshop paper (2013).
[6] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. and Dean, J.: Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems, pp. 3111-3119 (2013).
[7] Pan, S. J. and Yang, Q.: A survey on transfer learning, Knowledge and Data Engineering, IEEE Transactions on, Vol. 22, No. 10, pp. 1345-1359 (2010).
[8] Shinnou, H., Sasaki, M. and Komiya, K.: Learning under Covariate Shift for Domain Adaptation for Word Sense Disambiguation, PACLIC-29, pp. 215-223 (2015).
[9] Sugawara, H., Takamura, H., Sasano, R. and Okumura, M.: Context Representation with Word Embeddings for WSD, PACLING-2015, pp. 149-155 (2015).
[10] Sun, B., Feng, J. and Saenko, K.: Return of Frustratingly Easy Domain Adaptation, AAAI (2016).
[11] Sun, B. and Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation, Computer Vision-ECCV 2016 Workshops, pp. 443-450 (2016).
[12] Tzeng, E., Hoffman, J., Saenko, K. and Darrell, T.: Adversarial discriminative domain adaptation, arXiv preprint arXiv:1702.05464 (2017).
[13] 新納浩幸，古宮嘉那子，佐々木稔：nwjc2vec の ne-tuning，国語研言語資源活用ワークショップ，pp. PB-4 (2017).
[14] 新納浩幸，佐々木稔：k 近傍法とトピックモデルを利用した語義曖昧性解消の領域適応，自然言語処理，Vol. 20, No. 5, pp. 707-726 (2013).
[15] 新納浩幸，佐々木稔：共変量シフトの問題としての語義曖昧性解消の領域適応，自然言語処理，Vol. 21, No. 1, pp. 61-79 (2014).
[16] 新納浩幸，浅原正幸，古宮嘉那子，佐々木稔：nwjc2vec: 国語研日本語ウェブコーパスから構築した単語の分散表現データ，自然言語処理，Vol. 24, No. 5, pp. 705-720 (2017).