CrossWeigh の日本語 NER データセットへの適用とラベルノイズの調査

西村柾人 [1] 新納浩幸 [2]
[1] 茨城大学工学部情報工学科 [2] 茨城大学大学院理工学研究科情報科学領域
19t4054a@vc.ibaraki.ac.jp
hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

概要

通常, 教師あり学習は訓練データに誤りがないという前提で学習が行われるが, 実際には誤りを含む場合も多い. 特に NER のデータセットはラベルの定義に曖昧なものがあるため, タグ付けには誤りが生じやすい. このような背景から Wang らは誤ったラベルの付いたデータセットから NER のモデルを学習する CrossWeigh を提案した. 本論文では CrossWeigh を日本語 NER データセットに適用し, CrossWeigh の効果を確認する. 同時に, CrossWeigh によって生成される重み付きデータを調査することでデータセット内にある誤りの発見を試みた.

1 はじめに

固有表現認識（NER）とはテキストから固有表現（NE; Named Entity）となる単語を特定し, その単語に対する NE のラベルを推定するタスクである.

近年では, 学習データを大量に用意し, モデルを学習させる手法 [1] が多い. しかし, この学習データ内のラベルにノイズが含まれている場合, 深層学習を利用した手法では, これらのノイズに対して過学習してしまい, 性能劣化につながる問題が指摘されている [2].

Wang らもこのラベルノイズの影響について指摘している [3].Wang らによると NER ベンチマークの 1 つである CoNLL03 データセットにおいて, 約 5.38%のラベル付けミスが確認されている. さらに, これらのラベルノイズを修正したデータを用いることによって, 性能向上が確認されている. Wang らは同論文内で, ラベルミスを処理するための汎用的なフレームワークとして CrossWeigh を提案している. このフレームワークを利用することで, データ内のそれぞれの文章に対して再重み付けが行われ, 生成された重み付けデータを利用することによってラベルノイズの影響を軽減することができる. このような, より正確なデータを使って学習することでモデルの性能向上を図るアプローチは Data-Centric AI(DCAI) と呼ばれ, 注目を集めている [4][5].

本研究では, CrossWeigh をストックマーク株式会社が提供している, Wikipedia の日本語 NER データセット [6] に適用し, 重み付けデータの生成を行う. さらに, この重み付きデータを用いて, NER モデルの Flair[7] を学習することによって, 日本語での CrossWeigh の効果を確認する.

また, 先行研究 [8] では, データセット内のラベルノイズの調査を, CrossWeigh によって生成された重み付きデータの中で最も低い重みが付けられたデータから, 無作為に抽出した 100 件のデータのみ調査を行った. 本研究では, 最も低い重みが付けられたデータ全てに対して調査を行った. 加えて, 発見したラベルノイズと NER モデルの予測スコアとの関連度に応じてタイプ分類を行い, それら 2 つの相関性についても調査を行った.

2 関連研究

CrossWeigh のように, ラベルの誤りを自動的に検出する試みは以前から研究されている [9] が, この研究では品詞の間違いに対して修正を行うため, 今回のように NER に対して適用することができない. 他にも [10][11][12] ではラベルノイズが含まれる学習データを利用した NER の研究がされているが, これらでは NE が O ラベルとなる誤りに対してのみを対象としている.

また, CrossWeigh はモデルを改良するのではなく, 使用する学習データに対して処理を行い, モデルの性能向上を図る手法であるため DCAI の研究 [4] とも関連がある. ただし, 今回の実験では

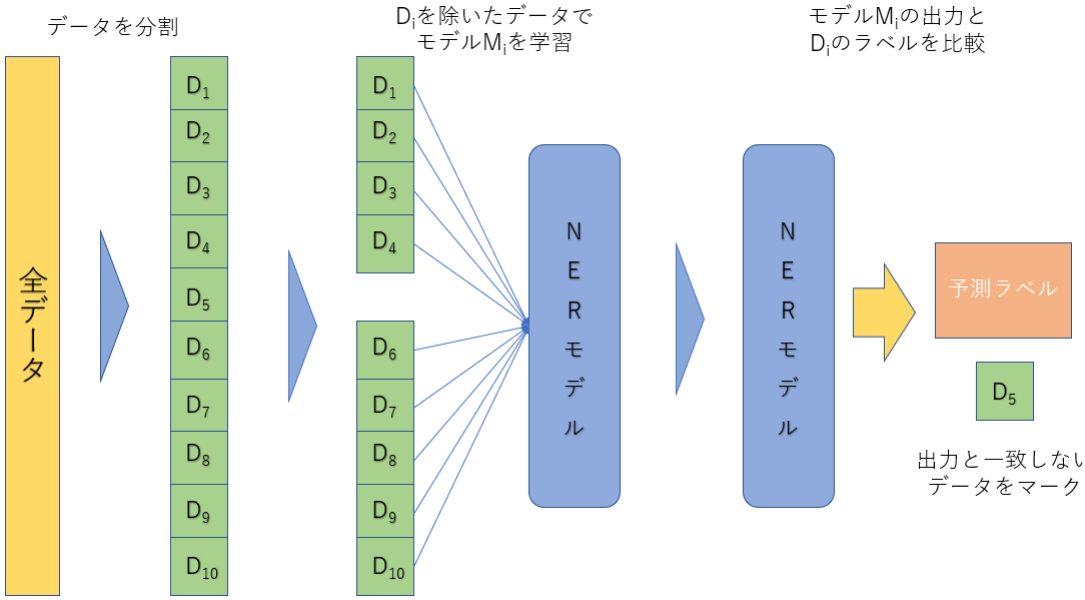

図 1　ラベルノイズの推定

DeepLearning.AI が行う Data-Centric AI Competition[1] に取り組むのではなく, CrossWeigh が日本語 NER データセットにおいても有効であるかについて検証を行う.

日本語データセットにおけるラベルの誤り問題に取り組んだ研究として [13] が挙げられる. この研究では, アノテーション漏れを推定しそのエンティティを学習データに追加することによって, 系列ラベリングを用いたエンティティ抽出の再現率の向上を行っている. また, [14] では Teacher-Student 学習を利用することでラベル誤りを含むデータにおける NER の性能向上について研究を行い, CrossWeigh と同様にラベルノイズの影響を緩和し性能向上が確認されている.

3　CrossWeigh

CrossWeigh は 2 つのモジュールから構成されている. 1 つ目はラベルノイズの推定, 2 つ目は推定したラベルノイズが含まれているデータに再重み付けを行うモジュールである.

1 つ目のラベルノイズの推定は k-分割交差検証をもとにした考えで実装されている. 実行手順を**図 1** に示す. まず学習データを k 個に分割することで $D_1, D_2, \ldots, D_k$ を作成する. そして, NER モデル $M_i (1 \leq i \leq k)$ を学習データの中から D_i を除いたデータによって学習させる. 作成した各モデル M_i を使って D_i の文のラベルに対して予測させ, モデルの出力と異なるラベルを持つ文は, ラベルノイズである可能性が高いものとしてマークする. この**図 1** の処理を, それぞれ異なるランダムな分割で行い, t 回実行する. ここまでを実行することで, データセットのラベルノイズの推定が行われる. 本実験では $k = 10, t = 3$ と設定して行われた.

2 つ目の再重み付けは, ラベルノイズの推定でマークされた文 x_i に対して重み w_i を調整することで実装されている. 最初に, 全ての文の重みを $w_i = 1$ として設定する. 次に, ラベルノイズとしてマークされた文の重みを**式 (1)** で計算する.

$$w_i = \epsilon^{c_i} \tag{1}$$

式 (1) の ϵ はパラメータであり, ラベルノイズ推定モジュールの精度に応じて設定する. 本実験では $\epsilon = 0.7$ とした. c_i は t 回実行されたラベルノイズの推定で, 何回モデルの出力と異なるラベルが推定されたかによって決まる. つまり c_i の取りうる値は $1 \leq c_i \leq t$ のいずれかの整数値である.

CrossWeigh はこれら 2 つのモジュールを使って, ラベルノイズの影響を緩和する重み付きデータを作成する.

4　実験

4.1　実験設定

本研究では, データセットとして, ストックマーク株式会社が提供している, Wikipedia の日本語 NER

1)　https://https-deeplearning-ai.github.io/data-centric-comp/

データセット [6] を用いた. このデータセットに対して, CrossWeigh および固有表現認識のタスク適用するために, トークンごとに分割し, ラベル付けを行った. 固有表現認識のラベルのフォーマットとしては IOB2 フォーマットを利用した. また, 整形したデータを学習データ, 検証用データ, テストデータの比率が 8:1:1 になるように分割した.

NER モデルとしては, Flair を用いた. ここで用いる Flair とは Akbik らの Contextual String Embeddings for Sequence Labeling[7] で提案された NER モデルを, 日本語データを用いて事前学習を行ったものである. さらに, CrossWeigh によって付与された重みを考慮して学習できるようにモデルを調整した.

4.2 評価方法

評価指標として F1-score(F 値) を用いた. CrossWeigh を適用していないデータ (Original data) と CrossWeigh を適用したデータ (Weighed data) それぞれに対して学習を行い, モデルがトークン列に対して適切なラベルを付与できたか否かをもとに算出を行った.

4.3 結果

Flair の結果を表 1 と付録の**図** 3 に示す. F1-score に関しては micro, macro のどちらにおいても CrossWeigh を適用したものが良い性能を示している. それぞれのラベルに対する結果では, 最大で 0.1 ポイント程のスコア向上がみられた. これらのことから日本語においても CrossWeigh の有効性がわかる. しかし, ラベルによってはポイントが減少しているものもあるため, これらに対しては今後の課題として分析が必要であると考えられる.

表 1 Flair の NER モデルにおけるラベルごとの結果

	Base line	Original data	Weighed data
その他の組織名	0.81	0.7674	0.8625
イベント名	0.84	0.8367	0.8324
人名	0.95	0.9341	0.9181
地名	0.87	0.9041	0.8932
政治的組織名	0.8	0.7863	0.8889
施設名	0.81	0.7800	0.8545
法人名	0.88	0.8755	0.8756
製品名	0.73	0.8117	0.7810
F1(micro)	0.86	0.8640	0.8758
F1(macro)		0.8370	0.8633

5 データセット内のラベルノイズの検出

ここでは CrossWeigh によって作成された重み付けデータを利用して, データセット内のラベルノイズを発見する手法を試みる.

5.1 ラベルノイズ

今回使用した Wikipedia の日本語 NER データセットにおけるラベルノイズの例として**図** 2 が挙げられる. (a) では文中の「旭川」と「小樽」に対して地名のラベルが付与されるべきであるが, ラベルが付与されていないため O タグとして処理が行われてしまう. また, (b) では「水夏希は」に人名のラベルが付与されてしまっているが, スパンが正しくないために「は」も I-人名タグとして処理が行われてしまう. これらのラベルノイズは CrossWeigh の再重み付けの際に, より小さい重みが付与されていると考えられる.

トークン	旭川	の	「	北海日日新聞	」	を
ラベル	O	O	O	法人名	O	O

(a) 異なるラベルが付与されているデータ

トークン	水夏希は	、	日本	の	女優	。
ラベル	人名	O	地名	O	O	O

(b) スパンがずれているデータ

図 2 ラベルノイズ例

5.2 手法

CrossWeigh を適用することで重み付きデータが作成される. この重み付きデータは各文ごとにラベルミスに応じた重みがつけられているため, この重みが最小のものがラベルノイズが含まれている可能性が高いデータであると考え, それらの重みが最小のデータを全て調査した. また同時に, ラベルノイズとなるトークンは, モデルがそのトークンのラベルを予測する際に, ラベルごとの分類スコアも小さいものになると考えたため, ラベルノイズと予測結果に関係があるかどうかも調査する. ラベルノイズであるか否かの判定は, Wikipedia の日本語 NER データセットを構築する際に使われた関根の拡張固有表現階層 [15] を参考にした.

5.3 結果

CrossWeigh によって生成された重み付きデータを調査したところ, データセットの全 5343 個のデータ

の内, 1484 個のデータに重みが最小のものが付与されていることがわかった. これらの 1484 個のデータを, トークンごとのラベルの分類スコアに着目して調査した結果, 大きく分けてのような 4 つのタイプに分類できた.

1. 分類スコアが最も低いトークンがラベルノイズであるもの
2. モデルの予測結果が間違っており, 元のデータがあっているもの
3. 分類スコアが最も低いトークン以外のものがラベルノイズであるもの
4. 予測結果と元のデータどちらもあっているもの

以上 4 つのタイプは以下の表 2 にまとめたデータ数であることがわかった.

表 2 タイプ分類の結果

Type	データ数
1	16
2	416
3	57
4	995

このタイプ分類からラベルノイズとなるものは Type1, 3 となる. つまり, 最小の重みが付与された 1484 個のデータの内, 73 個のデータがラベル付けに誤りがあるデータであることがわかった.

6 考察

6.1 CrossWeigh の日本語 NE データへの適用

Flair の結果である表 1 から, Original data で学習した Flair よりも, Weighed data で学習した Flair の方が NE の識別精度が向上していることがわかる. このことから CrossWeigh の手法は日本語 NE データにおいても有効であることがわかった.

しかし, 付録の**図** 3 に示したラベルごとの結果を見ると, 全てのラベルに対して識別精度が向上しているのではなく, 大きく精度が向上しているものから減少しているものもあるため, これらを分析, 改善することによってより良い重み付きデータを生成できると考えられる.

6.2 ラベルノイズの特定

5.3 節より, 使用した Wikipedia の日本語 NER データセット内のラベルノイズの探索範囲を狭めることに成功した. CrossWeigh の重み付きデータを利用することで効率的にデータセット内のラベルノイズを発見することができることがわかった.

ラベルノイズと予測の際の分類スコアの関係としては, あまり大きな相関は見られなかった. これには, 予測に使うモデルがラベルノイズを含むデータセットを学習に使用してしまっているため, ラベルノイズとなるトークンに対して実際に正しいラベルを予測させるのが難しいためと考えられる.

今回の実験で発見したラベルノイズが, 元のデータではどのラベルが付けられていて, 実際にはどのラベルが付くべきであったかを付録の表 3 にまとめた. ここから, 調査した Wikipedia の日本語 NER データセット内のラベルノイズでは, 元のラベルが O ラベルであるものが最も多いことがわかった. ここからラベルノイズの発生原因としては, 固有表現となる単語を見落としてしまうというものが最も多いことがわかった. CrossWeigh の有効性から考えられるように, ラベルノイズがモデルに与える影響は大きいため, どのようにしてデータセットのラベルノイズの発生を抑えることが今後の課題であることがわかった.

7 おわりに

本研究では日本語 NER データセットにおける CrossWeigh の有効性について検証を行った. CrossWeigh によって生成された重み付きデータを利用することで NER モデルの性能向上を確認することができた. 今後の課題は精度が下がったラベルに対しての分析である.

また, データセット内のラベルノイズの調査に関しては, CrossWeigh が生成する重み付きデータを利用することで, 実際にデータセット内のラベルノイズを発見することに成功した. ラベルノイズとなるトークンはモデルの予測結果がよい数値ではないと考え, 2 つの関係について調べてみたが, 相関性は見られなかった. ラベルノイズの内訳についても調査したが, 最も多いラベルノイズの発生原因はラベルの付け忘れであることがわかり, データセットを構築する際に, このようなミスの発生を抑制することが今後の課題である.

謝辞

本研究は 2022 年度国立情報学研究所公募型共同研究（22FC04）の助成を受けています.

参考文献

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. **arXiv preprint arXiv:1810.04805**, 2018.

[2] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization, 2016.

[3] Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. CrossWeigh: Training named entity tagger from imperfect annotations. In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)**, pp. 5154–5163, Hong Kong, China, November 2019. Association for Computational Linguistics.

[4] Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Douwe Kiela, David Jurado, David Kanter, Rafael Mosquera, Juan Ciro, Lora Aroyo, Bilge Acun, Sabri Eyuboglu, Amirata Ghorbani, Emmett Goodman, Tariq Kane, Christine R. Kirkpatrick, Tzu-Sheng Kuo, Jonas Mueller, Tristan Thrush, Joaquin Vanschoren, Margaret Warren, Adina Williams, Serena Yeung, Newsha Ardalani, Praveen Paritosh, Ce Zhang, James Zou, Carole-Jean Wu, Cody Coleman, Andrew Ng, Peter Mattson, and Vijay Janapa Reddi. Dataperf: Benchmarks for data-centric ai development, 2022.

[5] 古田拓毅. データ中心の視点から捉える深層強化学習. 人工知能, Vol. 37, No. 4, pp. 507–515, 2022.

[6] 近江崇宏. Wikipedia を用いた日本語の固有表現抽出のデータセットの構築. 言語処理学会第 27 回年次大会発表論文集 (NLP2021), 2021.

[7] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In **COLING 2018, 27th International Conference on Computational Linguistics**, pp. 1638–1649, 2018.

[8] 西村柾人, 新納浩幸. Crossweigh の日本語 ner データセットへの適用. Technical Report 21, 茨城大学工学部情報工学科, 茨城大学大学院理工学研究科情報科学領域, sep 2022.

[9] 中川哲司, 松本雄二. サポートベクターマシンを用いた誤り検出. 言語処理学会年次大会発表論文集, Vol. 8, pp. 563–566, 2002.

[10] Kun Liu, Yao Fu, Chuanqi Tan, Mosha Chen, Ningyu Zhang, Songfang Huang, and Sheng Gao. Noisy-labeled ner with confidence estimation, 2021.

[11] Zhanming Jie, Pengjun Xie, Wei Lu, Ruixue Ding, and Linlin Li. Better modeling of incomplete annotations for named entity recognition. 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. 729–734, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[12] Stephen Mayhew, Snigdha Chaturvedi, Chen-Tse Tsai, and Dan Roth. Named entity recognition with partially annotated training data, 2019.

[13] 伊藤雅弘, 山崎智弘. アノテーション漏れ推定を用いたエンティティ抽出. 言語処理学会第 27 回年次大会発表論文集 (NLP2021), 2021.

[14] 田川裕輝, 中野騰久, 尾崎良太, 谷口友紀, 大熊智子, 鈴木裕紀, 木戸尚治, 富山憲幸. Teacher-student 学習を利用したラベル誤りを含むデータにおける固有表現認識の性能向上. 言語処理学会第 28 回年次大会発表論文集 (NLP2022), 2022.

[15] Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. Extended named entity hierarchy. In **Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)**, Las Palmas, Canary Islands - Spain, May 2002. European Language Resources Association (ELRA).

A　付録

A.1　ラベルごとの F 値の比較

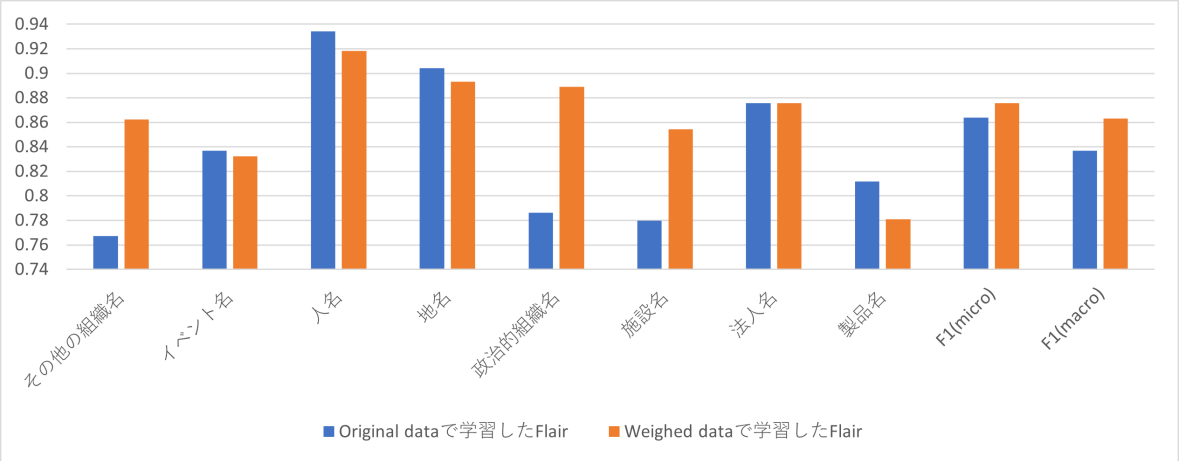

図 3　ラベルごとの F 値

A.2　発見したラベルノイズの内訳

表 3　ラベルごとの修正内訳

修正前のラベル / 修正後のラベル	その他の組織名	イベント名	人名	地名	政治的組織名	施設名	法人名	製品名	O
その他の組織名		0	1	0	2	0	3	0	3
イベント名	0		0	0	0	0	0	1	3
人名	0	0		1	0	0	6	1	6
地名	0	0	0		0	2	0	0	11
政治的組織名	0	0	2	1		1	6	0	0
施設名	0	0	0	2	0		0	1	1
法人名	1	0	0	1	0	0		0	1
製品名	0	1	0	1	0	0	1		12
O	0	1	2	0	0	0	1	2	