固有表現抽出における機械学習モデルの効率的な教示方法に関する研究

筑波大学
図書館情報メディア研究科
2021年3月
小林　滉河

目 次

図 目 次

第1章　固有表現抽出と教示方法

1.1　固有表現抽出

固有表現抽出 (Named Entity Recognition; NER) とは，文章から人名 (PER) や地名 (LOC)，組織名 (ORG) といった固有表現を自動的に取り出すことを目的とした自然言語タスクである．取り出された固有表現は検索や形態素解析といった様々なタスクに利用され，応用先の精度を大きく左右させる要因となり，非常に重要な技術である [1, 2].

固有表現抽出の具体的な例として次のような文について考える。

田中　は　筑波　大学　の　学生　だ　.

この例文に対して，期待される出力は次の通りである．

[$_{\text{PER}}$ 田中]　は　[$_{\text{ORG}}$ 筑波　大学]　の　学生　だ　.

例の通り，固有表現抽出は系列中から重複の無いようにラベル付き括弧を付与する系列セグメントタスクであるが，これは系列ラベリング問題として扱われることが多い [3]．系列ラベリング問題とは，ある入力列 $x = (x_1, \ldots, x_l)$ に対して，ラベル列 $y = (y_1, \ldots, y_l)$ を予測するタスクのことを指す．系列ラベリング問題として扱われるタスクは固有表現抽出の他に，文章中の単語に名詞や動詞といった品詞を予測する品詞タグ付けが知られている．固有表現抽出を系列ラベリング問題として解くために，BIO フォーマットというラベルフォーマットがよく利用される．BIO フォーマットは，単語が各固有表現のどこの位置に属するかを表す接頭字 B (Begin), I (Inside) とその固有表現のクラス名を表す単語から構成される．また固有表現ではない単語に対しては O (Other) というタグが付与される．先程の例文に対して BIO フォーマットを適用し，系列ラベリング問題における入力列を文章，ラベル列を固有表現タグ列とすると次のように表現出来る．

入力列 x	田中	は	筑波	大学	の	学生	だ	.
ラベル列 y	B-PER	O	B-ORG	I-ORG	O	O	O	O

固有表現抽出は一般的な分野のみならず近年，物質材料やバイオ，薬学といったより専門性の高いドメインに対する適用を目指す研究が増加している [4, 5]．しかし，専門分野における教師データの作成には，専門家によるアノテーションが必要不可欠である．そのため低コストで高い抽出性能を持つ固有表現抽出モデルを作成する手法が求められている．

1.2　固有表現抽出モデル

固有表現抽出モデルは，近年では機械学習のモデルを利用して系列ラベリング問題として解かれることが多い．本節ではいくつかの主要なモデルを紹介する．

1.2.1 条件付き確率場

条件付き確率場 (Conditional Random Field; CRF) [6] は系列ラベリング問題を扱う分類器である．自然言語処理の分野では条件付き確率場でも特に隣り合うラベル同士の依存関係を考慮して予測を行う Linear-Chain CRF がよく利用される．条件付き確率場では入力列 x とラベル列 y に対するスコア関数は次のように定義される．

$$\mathrm{score}(x, y) = P(y|x) = \frac{\exp\left(s(x, y)\right)}{\sum_{y' \in \mathcal{Y}_x} \exp\left(s(x, y')\right)} \tag{1.1}$$

ここで，$\mathcal{Y}_x$ は入力列 x に対して，取りうる全てのラベル列 y の集合を表す．また s は入力列 x とラベル列 y を入力とし，実数を出力する関数であり，ラベル y_i からその次のラベル y_{i+1} への遷移スコア $T_{y_i, y_{i+1}}$ と入力 x_i がラベル y_i を生成するスコア E_{x_i, y_i} の和によって表すことが出来る．

$$s(x, y) = \sum_{i=0}^{n} T_{y_i, y_{i+1}} + \sum_{i=1}^{n} E_{x_i, y_i} \tag{1.2}$$

このときの教師データ (x, y) に対する対数損失は次のように定義される．

$$\begin{aligned} L(y) &= -\log \mathrm{score}(x, y) \\ &= \log \sum_{y' \in \mathcal{Y}_x} \exp\left(s(x, y')\right) - s(x, y) \end{aligned} \tag{1.3}$$

つまり，対数損失は取りうるラベル列の確率の和と正解のラベル列の確率の差に関係している．また図 1.1 からも分かるように，式 1.1 の右辺の分母は単純に計算を行うと，ラベルの種類の系列長乗の計算オーダとなり，求めることは困難である．しかし，動的計画法の一種である前向き・後向きアルゴリズムを用いることで，効率的な計算が可能であることが知られている．

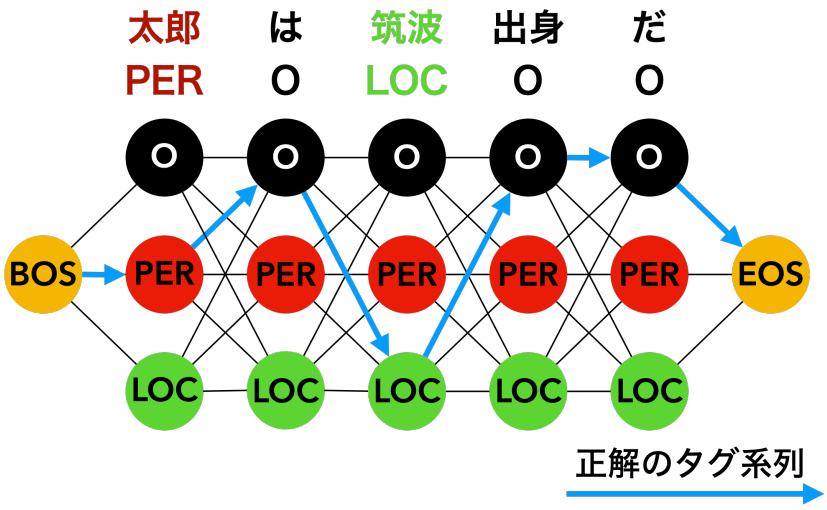

図 1.1: 固有表現抽出における CRF の例

また入力列 x に対して，最適なラベル列 $\hat{y}$ をスコア関数を最大化する $\mathcal{Y}_x$ の要素としたとき，最適なラベル列は次のように求めることが出来る．

$$\hat{y} = \underset{y' \in \mathcal{Y}_x}{\arg\max}\{\mathrm{score}(x, y')\} \tag{1.4}$$

最適なラベル列の計算も愚直に計算を行った場合，系列の長さに対して指数オーダの計算量が必要となる．しかし，分配関数を求めるときと同様に動的計画法のビタビアルゴリズム (Viterbi algorithm) を用いることで効率的な計算が可能になる．

1.2.2 LSTM-CRF

次にニューラルネットワークを用いた固有表現抽出モデルとして，よく知られている Lample ら [7] の LSTM-CRF モデルについて説明する．Lample らは事前学習済みの分散表現と文字単位の分散表現を利用することで，特徴量エンジニアリングを必要とせず高い抽出性能を持つ固有表現モデルを提案した．まず，LSTM-CRF の特徴抽出に使われている LSTM について説明する．

長短期記憶ユニット (Long Short-Term Memory: LSTM) [8] は，勾配消失問題を解決するために考案されたゲート付き再帰的ニューラルネットワークの一つである．数学的には次のように定義される [9].

$$\begin{aligned}
\boldsymbol{h}_t &= \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t) \\
\boldsymbol{c}_t &= \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i} \odot \tilde{\boldsymbol{c}}_t \\
\tilde{\boldsymbol{c}}_t &= \tanh(\boldsymbol{x}_t \boldsymbol{W}^{xc} + \boldsymbol{h}_{t-1} \boldsymbol{W}^{hc} + \boldsymbol{b}^c) \\
\boldsymbol{o}_t &= \sigma(\boldsymbol{x}_t \boldsymbol{W}^{xo} + \boldsymbol{h}_{t-1} \boldsymbol{W}^{ho} + \boldsymbol{b}^o) \\
\boldsymbol{i}_t &= \sigma(\boldsymbol{x}_t \boldsymbol{W}^{xi} + \boldsymbol{h}_{t-1} \boldsymbol{W}^{hi} + \boldsymbol{b}^i) \\
\boldsymbol{f}_t &= \sigma(\boldsymbol{x}_t \boldsymbol{W}^{xf} + \boldsymbol{h}_{t-1} \boldsymbol{W}^{hf} + \boldsymbol{b}^f)
\end{aligned} \tag{1.5}$$

ここで $\odot$ はアダマール積であり，同じサイズの行列について要素同士をかけ合わせる操作である．$A, B, C \in \mathbb{R}^{n \times m}$ のとき，

$$\begin{aligned}
C &= A \odot B \\
c_{ij} &= a_{ij} b_{ij}
\end{aligned} \tag{1.6}$$

のように演算を行う．また tanh と σ は次のような活性化関数である．

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{1.7}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{1.8}$$

図 1.2 に LSTM セルのアーキテクチャを示す．

LSTM を条件付き確率場における生成スコア E_{x_i, y_i} の算出に用いたモデルが LSTM-CRF である．LSTM では文字レベルと単語レベルの事前学習済み単語埋め込みの二つと LSTM 層を利用して，CRF に入力する特徴量を作成する．単語 x_i に対応する事前学習済みの単語埋め込みを $\boldsymbol{e}_i \in \mathbb{R}^{d_w}$ とし，この単語 x_i の文字列長が n_i のときの文字レベル単語埋め込み

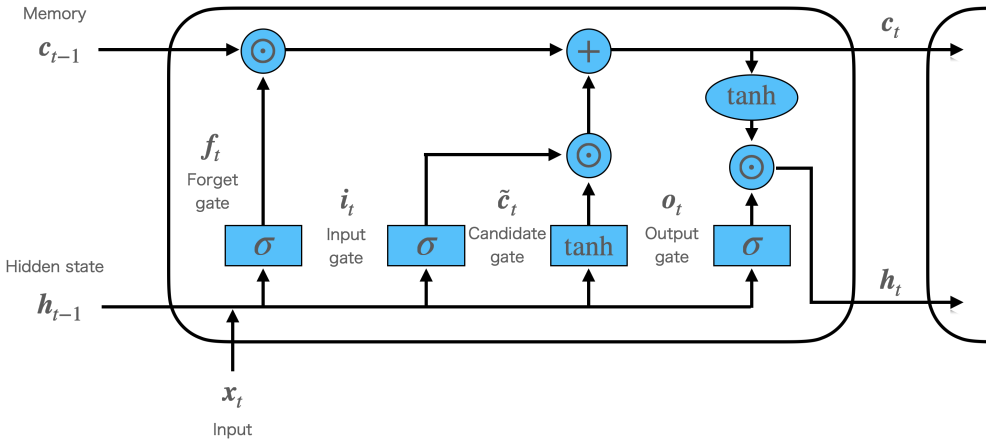

図 1.2: LSTM セルのアーキテクチャ

列を $\boldsymbol{c}_i = (\boldsymbol{c}_1^{(i)}, \ldots, \boldsymbol{c}_{ni}^{(i)})$, $\boldsymbol{c}_j^{(i)} \in \mathbb{R}^{d_c}$ とする．このとき LSTM-CRF のアーキテクチャは数学的に次のように示すことが出来る．

$$\begin{aligned} \hat{y} &= \mathrm{CRF}(\mathrm{MLP}(\mathrm{LSTM}(\boldsymbol{z}_{1:l})) \\ \boldsymbol{z}_i &= [\boldsymbol{e}_i; \boldsymbol{e}_{\boldsymbol{c}_i}] \\ \boldsymbol{e}_{\boldsymbol{c}_i} &= \mathrm{LSTM}(\boldsymbol{c}_i) \end{aligned} \tag{1.9}$$

ここで，[;] はベクトルの結合，MLP は多層パーセプトロンを表す．また LSTM 層に双方向 LSTM を用いた Bi-LSTM を適用した際の LSTM-CRF のアーキテクチャは図 1.3 の通りである．

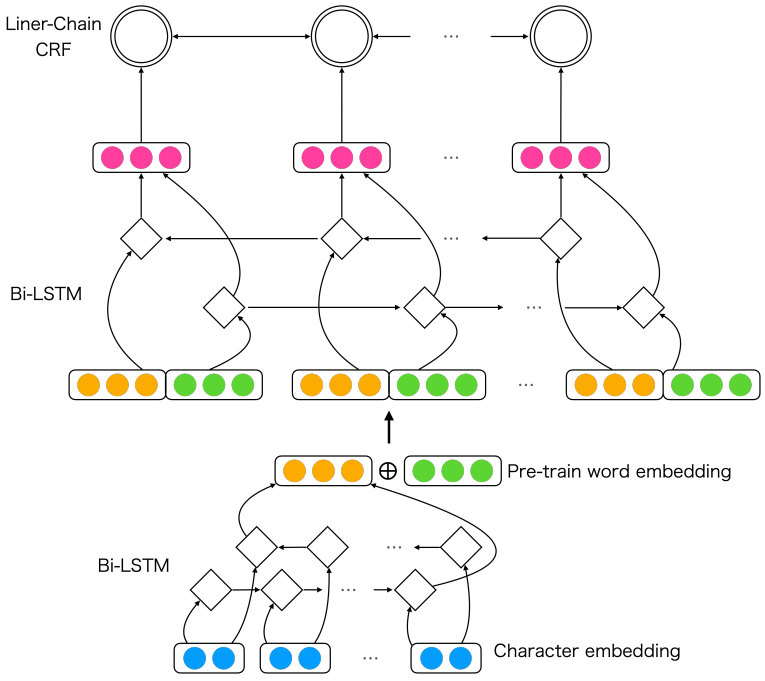

図 1.3: BiLSTM-CRF

1.3 学習に利用可能な資源

固有表現抽出における教師データは通常アノテーションコーパスといわれるデータセットを利用することが多い．これは文章中の単語に対応する固有表現タグを人手によって付与したデータセットのことを指す（図 1.4）．

佐藤	太郎	は	4月	から	東京	で	暮らす
B-PER	I-PER	O	B-DATE	O	B-LOC	O	O

図 1.4: アノテーションコーパスの例

付与したタグのことをアノテーションといい，アノテーションを行う人をアノテータと呼ぶ．アノテーションコーパスの作成には，文章中に含まれる全ての単語に対して固有表現タグを付与する必要がある．そのためアノテータは，固有表現かどうか確信が持てない単語に対しても何らかのタグを付与しなければならない．これによって，本来信用すべきアノテーションコーパスに確信が持てないタグを付与したことによって生じた間違いが含まれるといった問題がある．そこで図 1.5 のように，アノテータが自信の無い単語や固有表現か分からない単語に対して不確かなラベルを付与せず，確信の持てる一部の単語に対してのみアノテーションを行い作成したコーパスが存在する．

佐藤	太郎	は	4月	から	東京	で	暮らす
B-PER	I-PER	-	-	-	B-LOC	-	-

図 1.5: 部分的アノテーションコーパスの例

これを通常のアノテーションコーパスと区別できるように，本論文中では部分的アノテーションコーパスと呼ぶことにする．

対して，アノテーションが全く行われていない文章も存在する (図 1.6)．このようなラベルが付与されていない文章の集合を生コーパスと呼ぶ．生コーパス単体での学習は教師なし学習といい，教師あり学習に比べ得られる情報は少ない．しかしアノテーションコーパスや部分的アノテーションコーパスに比べて安価に手に入れやすい．

佐藤	太郎	は	4月	から	東京	で	暮らす
-	-	-	-	-	-	-	-

図 1.6: 生コーパスの例

これら三つのテキストから作成される学習資源に加え，化学や医学といった専門分野においてはそのドメインに関する用語を集めた辞書が利用可能なことも多い．辞書を用いた学習については第 3 章にて詳しく述べる．

1.4 教示方法

固有表現抽出器に関わらず，多くの機械学習モデルを訓練する際に必要となるのは教師データである．ここで教師データの作成とそのデータを用いてモデルを学習する一連の流れのことを教示と呼ぶことにする．現時点で主要な教師データの作成方法は以下の三種類である．

1. 専門家がデータセットにラベリングを行う．

2. 外部知識を用いて，教師データを自動的に作成する．

3. クラウドソーシングによって，教師データを収集する．

これらの教師データの作成方法には課題が存在しているが，一般的な機械学習モデルでは，その解決方法となる手法がそれぞれ提案されている．1 つ目の作成方法では，専門家を大量の教師データ作成のために雇用する必要があり，高いコストが生じるといった課題が存在する．これには能動学習を用いて，必要な教師データを減少させる手法の提案がなされている．2 つ目の外部知識を用いた教師データの作成方法では，遠距離教師あり学習という手法が提案されている．最後のクラウドソーシングを用いて作成した教師データには間違ったラベルが生じやすいという課題には，生成モデルを用いたラベルの集約手法が提案されている．しかし，固有表現抽出は自然言語における系列ラベリングタスクであるため，これらの手法を直接適用することは出来ない．

本論文では固有表現抽出におけるこれらの教師データの作成にて生じる課題を解決するため，いくつかの手法の提案と調査を行う．一番目の課題を解決するために，第 2 章では少量の教師データでも高い抽出性能を実現するために，固有表現抽出における点予測モデルと能動学習の適用手法について提案する．二番目の課題を扱う第 3 章では固有表現抽出における辞書と生コーパスを用いる遠距離教師あり学習について述べた後，辞書マッチングの誤りを考慮したモデルによる適合率向上の可能性について検証を行う．最後の課題に対応する第 4 章にて，不特定多数からのアノテーションを前提とするクラウドソーシングにて，ラベルの集約性能と集約モデルの複雑性について調査を行う．第 5 章にて全ての提案手法と調査についてと固有表現抽出における教示方法の今後の展開についてまとめる．

第2章　能動学習

2.1　固有表現抽出における能動学習の問題

少数の教師データで効果的な学習を行うアプローチの一つとして能動学習という手法が知られている．この手法は教師データの作成時に専門的な知識がアノテーターに要求されるような高いコストがかかるような場合やそもそも教師データの作成にあまりコストをかけることが出来ないときによく利用される．能動学習ではプールと呼ばれる大量のラベル無しデータの中から，機械学習モデルがラベルが付与されると多くの情報が得られるデータの選択を行う．選択したデータについてオラクルと呼ばれるラベルの答えを知っている存在に対して問い合わせを行う．固有表現抽出タスクにおいて，オラクルはアノテーターとなる．そうして情報の多いデータだけを集めることで，情報が少ないデータに対する無駄なアノテーションを減らし，少ないデータで高い性能を持つモデルを作成することを目的としている．

固有表現抽出に対して能動学習を適用する場合，大きく分けて二つの問題が存在する．1つ目はアノテーションについてである．固有表現抽出手法の多くは部分的アノテーションコーパスを教師データとして利用できない．そのため，アノテーターは文章全体に対してアノテーションを行う必要がある．文章には「a」や「the」といったストップワードが含まれる上に，よく頻出する固有表現に対して度々アノテーションを行わなければいけない．これらの比較的に得られる情報量が少ない単語にアノテーションを付与することによる，アノテータへの負担は大きい．2つ目の問題は学習速度についてである．能動学習では多くの回数の学習と推論を行う．そのため部分的アノテーションコーパスを利用可能であっても，大量の時間を必要とするモデルに対して能動学習を適用することが難しいといった問題が生じる．

本章では，点予測 [10] を用いた固有表現抽出手法と能動学習への適用を提案する．点予測は現在主流の手法である深層学習を用いた固有表現抽出に比べ，アノテーションコーパスが大量にある状況において性能は劣る．しかし，点予測は部分的アノテーションコーパスが利用可能であり，文章中の全単語に対してアノテーションを行う必要がない．また利用する分類器によっては高速な学習と推論が可能になる．そのため提案手法では，多クラスロジスティック回帰を用いた点予測による能動学習について検討を行う．また既存手法に対して能動学習を適用させたモデルと提案手法を比較し，少ないアノテーションコストで高い抽出性能を持つことを示す．

2.2　先行研究

2.2.1　能動学習

能動学習において，オラクルに問い合わせるデータが得られる環境のことをシナリオという．例えば逐次的に教師なしデータが入ってくる状況や元から大量の教師なしデータが用意されている状況が考えられる．シナリオには大きく分けて Membership Query Synthesis、

Stream-based Selective Sampling、Pool-based Sampling の三種類が存在する [11]。Membership Query Synthesis、Stream-based Selective Sampling の二つは次々にラベルなしデータが入力されるようなデータについてのシナリオである．Membership Query Synthesis では入力されたデータが含まれる全てからデータの生成及び選択を行う．そのため実世界には存在しないデータを生成し，問い合わせを行うことも可能である．しかし，オラクルがアノテータつまり，人間のとき答えが分からないようなデータを生成し問い合わせてしまうという問題が発生する．Stream-based Selective Sampling は入力されたデータに対して，ラベルを問い合わせるかどうかを決めるシナリオである．このシナリオは実際に入力されたデータのみを問い合わせの対象とするので実世界のタスクに対しても適用可能であり，自然言語処理の分野では，Ido ら [12] は HMM を用いた品詞タグ付けに，Yu ら [13] は情報検索におけるランキング学習に適用している．これまでに説明した二つの問い合わせ戦略に対して，Pool-based Sampling はまとまった教師無しデータがある状況を対象としている．本研究では Pool-based Sampling を用いて実験を行った．

またこれらのシナリオでは各データの情報量尺度を計算し，それを元に問い合わせを行う．この情報量尺度の計算方法をクエリ戦略といい，数多くの手法が提案されている．その中でも最もよく利用されるフレームワークが Uncertainty Sampling[14] である．Uncertainty Sampling は教師無しデータの中でモデルが最もラベル予測しにくいものをアノテータに問い合わせる手法である．ここでは実験に用いたラベル予測の難しさを表す尺度について説明する．

１つ目の尺度は Least Confident である．この指標は Uncertainty Sampling の中で最もシンプルな指標である．Least Confident ではデータが最も属する可能性が高いクラスについての予測確率が小さいデータをラベル予測しづらいデータとして扱う．例えば扱うタスクが二値分類のとき，Least Confident ではモデルがある一方のクラスに所属する確率が 0.5 に近いような値であるデータについて問い合わせを行う．系列ラベリングタスクにおいて入力列を $x \in \mathbb{R}^l$ ，モデルが予測する最適なラベル列を $\hat{y} \in \mathbb{R}^l$ としたとき，以下のように表すことが出来る．

$$\phi^{LC}(x) = 1 - P(\hat{y}|x) \tag{2.1}$$

またこの数式は $l = 1$ のとき多クラス分類モデルに対する指標としてみなすことが出来る．

２つ目は Marginal Sampling である．この指標では一番属する可能性が高いクラスと二番目に属する可能性が高いクラスの確率の差を利用する．系列ラベリングタスクでは，モデルが予測した最適なラベル列を $\hat{y}_1$，二番目に最適なラベル列を $\hat{y}_2$ としたとき次のように定義される．

$$\phi^{M}(x) = -\Big(P(\hat{y}_1|x) - P(\hat{y}_2|x)\Big) \tag{2.2}$$

この手法では二番目に最適なラベル列の曖昧性についても考慮しているため，Least Confident に比べ多くの情報を元に曖昧なデータを選択することが出来る．

Uncertainty Sampling の他にもラベルなしデータに対して複数個のモデルによる評価を行い，評価が分かれるようなデータを問い合わせる Query-By-Committee [15]．そのデータを学習することでモデルが大きく変わるようなものを対象に問い合わせを行う Expected Model Change といった手法が提案されている．

2.2.2 固有表現抽出に対する能動学習の適用

現在，固有表現抽出タスクに能動学習を適用させる研究は数多くなされている．Settles ら [16] は系列ラベリングタスクに対する能動学習の効果を確認するために，CRF に対して様々なクエリ戦略を適用し，固有表現抽出タスクに対する性能比較を行った．Yanyao ら [17] は対象となる固有表現が多いとき CRF に比べて LSTM が高速であることを示し、文字レベル CNN と単語レベルの CNN，LSTM を組み合わせた CNN-CNN-LSTM モデルを提案し，深層学習モデルであっても比較的高速に能動学習が行えることを示した． しかし，これらの手法はアノテータに対して文章単位でのアノテーションを必要となり，文章中の一部について単語を問い合わせることは出来ない．そのためアノテータはラベルを付与しても学習があまり進まないような単語にもアノテーションを行わなければならないといった問題が残されている．本章では，高速に学習可能かつ部分的アノテーションコーパスを教師データとして利用できる点予測に能動学習を適用させることで，アノテーションコストの削減を目指す．

2.3 提案手法

2.3.1 点予測による固有表現抽出

本手法では Neubig [18] が提案した品詞予測のための点予測を固有表現抽出に活用できるよう拡張を行う．まずある入力列 $x = (x_1, \ldots, x_l)$ とラベル列 $y = (y_1, \ldots, y_l)$ について考える．点予測では入力された単語全てに対して，独立にラベルの推定を行うため CRF や LSTM のように系列としての情報を利用しない．つまり予測に入力列の各要素 x_i から構成される特徴のみを入力とし，y_i を予測する．このとき y_i の候補は B-LOC, I-LOC, O のような様々なクラスが対象となり，これは多クラス分類問題としてみなすことが出来る．多クラス分類問題を解くための手法として多クラスロジスティック回帰やサポートベクターマシン，決定木といった様々なモデルが存在する．本研究では多くの学習と推論を行う能動学習に対してモデルを適用する必要があるため，高速に学習を行うことが出来る多クラスロジスティック回帰を点予測に利用した．

まずロジスティック回帰について説明する．ロジスティック回帰は二値分類のための線形モデルであり，入力 x に対して出力 $y \in \{+1, -1\}$ が属している確率を求めるためにシグモイド関数を用いて計算を行う．

$$P(y|x) = \frac{1}{1 + \exp\left(-y\boldsymbol{w}^T\boldsymbol{\phi}(x)\right)} \tag{2.3}$$

ここで $\boldsymbol{\phi}(x)$ は素性関数 ϕ から構成される素性 $\phi(x)$ からなる素性ベクトルである．ロジスティック関数は入力 x を素性関数 ϕ を通して観測を行い，y のラベルを予測する．この素性関数は人手によって設計される．素性関数の例として，入力 x が単語の場合その品詞情報や単語そのものといったものが挙げられる．$\boldsymbol{w}$ は素性ベクトルと同じ次元を持つ重みベクトルであり，学習の際はこの重みベクトルを学習データに適合するように学習を行う．ロジスティック回帰では，学習に用いる教師データ $(\boldsymbol{x}, \boldsymbol{y}) = (\langle x^{(1)}, \cdots, x^{(n)} \rangle, \langle y^{(1)}, \cdots, y^{(n)} \rangle)$ に対する対数損失は次のように計算される．

$$L(\boldsymbol{y}) = \log \prod_{i=1}^{n} P(y^{(i)}|x^{(i)}) = \sum_{i=1}^{n} P(y^{(i)}|x^{(i)}) \tag{2.4}$$

この対数損失は凸関数であるため，勾配降下法を用いることで重みベクトル $\boldsymbol{w}$ の局所最適解を求めることが出来る．

二値分類器であるロジスティック回帰を多クラスに拡張したものが，多クラスロジスティック回帰である．多クラスロジスティック回帰では K 個の要素からなるラベル集合 $\mathcal{Y} = \{1, 2, \cdots, K\}$ のラベル $y \in \mathcal{Y}$ の確率を求める機械学習モデルである．多クラスロジスティック回帰において，入力 x に対するラベル y の条件付き確率は次のように計算出来る．

$$P(y|x) = \frac{\exp\left(-y\boldsymbol{w}_y^T\boldsymbol{\phi}(x)\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(-y'\boldsymbol{w}_y'^T\boldsymbol{\phi}(x)\right)} \tag{2.5}$$

本手法では i 番目の単語 x_i のタグ y_i を予測するとき，単語 x_i の周辺単語である $x_{i-m+1}, \cdots, x_{i-1}$ と $x_{i+1}, \cdots, x_{i+m}$ を含めた情報を入力とする．ここでの m は窓枠のことを指す (図 2.1).

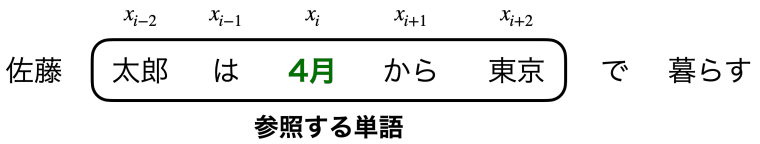

図 2.1: y_i を予測の際に参照する単語の例 ($m = 2$ において)

つまり，本手法において先程の多クラスロジスティック回帰の式は以下のように書き下すことができる．

$$P(y|x_{i-m+1}, \cdots x_i \cdots, x_{i+m}) = \frac{\exp\left(-y\boldsymbol{w}_y^T\boldsymbol{\phi}(x_{i-m+1}, \cdots x_i \cdots, x_{i+m})\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(-y'\boldsymbol{w}_y'^T\boldsymbol{\phi}(x_{i-m+1}, \cdots x_i \cdots, x_{i+m})\right)} \tag{2.6}$$

実験では入力 $x_{i-m+1}, \cdots x_i \cdots, x_{i+m}$ の素性として，Graham ら [18] の手法を参考として次の情報を利用する．

- 単語: 識別するラベル y_i に対応する単語 x_i について窓枠 m を設定し，単語 $x_{i-m+1}, \ldots, x_i, \ldots, x_{i+m}$ を素性として用いた．
- 単語種別: 単語に関する素性と同様に文字種に関する情報も $x_{i-m+1}, \ldots, x_i, \ldots, x_{i+m}$ を元に作成し素性とした．具体的には対象の単語は大文字から始まっているか、数字のみで構成されているか等を利用した．
- 品詞: 推定する単語の品詞と周辺単語の品詞も合わせて素性として利用した．

また森ら [10] は多クラス分類である形態素解析タスクを対象とした点予測において，教師データや辞書に出現しなかった単語を名詞とみなし，出現した単語に関しては品詞候補毎にモデルを作成し一対多方式 (one-versus-rest) を用いることで品詞の推定を行った．対して固有表現抽出タスクでは，形態素解析のタスクとは異なり，未知の単語に対し O タグや B-LOC のような特定のタグを自動的に付与すると，抽出性能が著しく減少する様子が予備実験を通じて分かった．そのため本研究では一つのモデルを全単語に対して使用した。

2.3.2 能動学習の適用

点予測は部分的アノテーションコーパスを教師データとして学習可能である．しかし，部分的アノテーションコーパスが利用可能な状況においても，情報量が少ない単語に対して多くのアノテーションが付与されたところで性能は向上しない．そこで情報が多い単語に対して，優先的にアノテーションを行うアルゴリズムが要求される．本手法では点予測の部分的アノテーションコーパスが利用可能かつ高速に学習と予測が出来るというメリットを最大限に活かすために能動学習を用いる．

本手法では Pool-based Sampling を対象して実験を行った．Pool-based Sampling は少数の教師ありデータの集合 L と大量の教師なしデータ集合 U が資源としてあることを前提として考案された能動学習の手法である．この大量の教師なしデータ集合は一般的にプールと呼ばれる．モデルは現在の推論に基づいて，最も学習に有効と思われるデータをプール中から選択する．選択したデータをアノテータに対し問い合わせ，ラベルを付与してもらう．ラベルが付与されたデータを教師データに加え，再度学習を行いモデルを更新する．これら一連の流れを繰り返すことで，モデルを改善していくことが Pool-based Sampling である．

Pool-based Sampling ではデータを一つ問い合わせ，ラベルが付与されるごとに学習を行う逐次型能動学習と複数個のデータを一度に問い合わせ，全てのラベルがアノテーションされたときに再度モデルの学習を行うバッチ型能動学習の二つが存在する．本論文ではより実環境に沿った実験を行うため，バッチ型能動学習を選択した．バッチ型能動学習は一般的にアルゴリズム 1 で表すことが出来る．

Algorithm 1 Pool-based Sampling におけるバッチ型能動学習

Require: ラベル付きデータ L, ラベルなしデータ U, クエリ戦略 $\phi(\cdot)$, バッチサイズ B

```
repeat
  θ ⇐ train(L)
  for b = 1 to B do
    x*_b ⇐ arg max_{x∈U} φ(x)
    L ⇐ L ∪ ⟨x*_b, label(x*_b)⟩
    U ⇐ U \ x*_b
  end for
until
```

ここでバッチ型能動学習について詳しく説明する．アルゴリズム 1 ではまずラベル付きデータ L に対し，関数 $train(\cdot)$ を用いてモデルのパラメータ θ を更新する．その後，クエリ戦略 $\phi(\cdot)$ に基づいてプール U の B つの教師なしデータを取り出し，アノテータに問い合わせを行う．問い合わせたデータはラベル付きデータとして L に加えられ，U から削除される．これを繰り返しが Pool-based Sampling におけるバッチ型能動学習である．またバッチ型能動学習は $B = 1$ のとき逐次型能動学習とみなすことができる．

次に具体的な点予測によるクエリ戦略について説明する．プールからデータを選択する際，CRF や LSTM，LSTM-CRF のような機械学習モデルは系列の情報を用いて学習するため，図 2.2 のように文中に含まれる全単語のラベルをアノテータに問い合わせなければならない．

しかし，点予測は部分的アノテーションコーパスを教師データに利用できるため，文章中に含まれる特定の単語だけを問い合わせることが出来る（図 2.3）．

このような単語単位での問い合わせを行うことで既存手法と比べ，同じ問い合わせ数でもモデルがより重要だと推論する単語の問い合わせ回数を増やすことが出来る．そのため

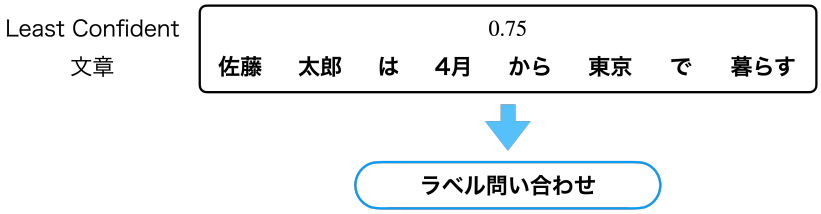

図 2.2: CRF におけるラベル問い合わせ

図 2.3: 点予測におけるラベル問い合わせ

効率的なモデルの学習が可能になると考えられる．

続いて，アノテータへのラベル問い合わせの方法について説明する．アノテータは固有表現全体を確認しなければ，ラベルの付与を行うことが出来ない．そこで次のようにクエリ問い合わせとアノテーションを行う．まず，クエリ戦略によって選択された単語 x_i をアノテータに問い合わせる．このとき x_i が固有表現の途中である時，固有表現全体にアノテーションを行えるよう単語 x_{i-1} も問い合わせる．これを固有表現の始まりが出現するまで繰り返し行う．また x_i が固有表現の始まりのとき，次の単語 x_{i+1} の問い合わせを行う．これも固有表現の終わりが出現するまで行う (図 2.4)．

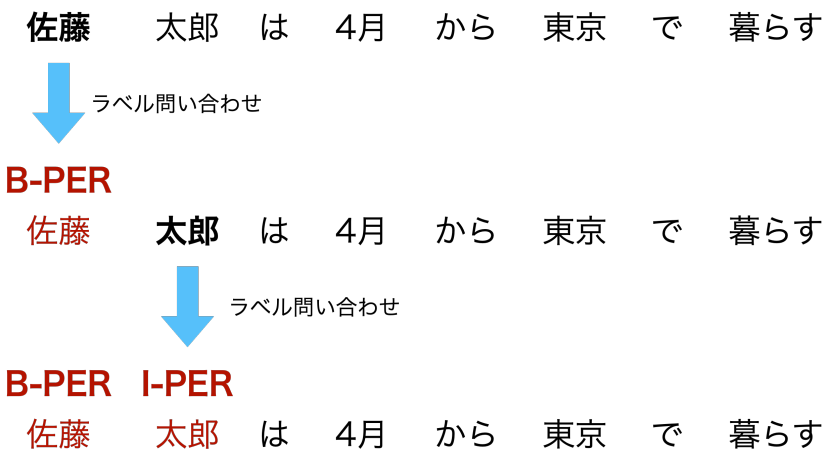

図 2.4: ラベル問い合わせの過程

2.4 実験

本研究では，点予測を用いた固有表現抽出における能動学習の有効性について既存手法との優位性を示すために二種類の実験を行う．

1 つ目の実験は点予測を用いた固有表現抽出に対し，能動学習が実際に有効であるかの検証である．大量のラベル無しデータ集合 U の中からランダムに単語を選択しラベル問い合

わせを行うモデルと Least Confident に基づいたクエリ戦略によって，選ばれた単語に対してラベルの問い合わせを行うモデルについて性能比較を行う．

2 つ目の実験では，比較手法と点推定に対して能動学習を適用した際の性能比較についての検証を行う．比較手法には固有表現抽出モデルである条件付き確率場と，Ma ら [19] が提案した深層学習モデルである LSTM-CNN-CRF を選択した．クエリ戦略は点予測と条件付き確率場には Least Confident を LSTM-CNN-CRF には Marginal Sampling[20] を採用した．

両実験において，今回は人手によるアノテーションは行わず，事前にラベルが分かっている固有表現抽出のデータセットである CoNLL-2003 を利用し仮想的な能動学習環境を構築した．この際データセットにおける教師データをランダムに選択し，少量のラベル付きデータ集合 L とラベルの情報を削除した大量のプール U の二つに分割し，そのときに作成されたラベル付きデータ集合 L を最初の学習時に利用した．また，本実験ではバッチ型能動学習を元に学習を行った．ハイパーパラメータであるバッチ数は 5,000 に設定した．ただし，文章単位でしか学習を行うことが出来ない条件付き確率場と LSTM-CNN-CRF については文章ごとのラベル問い合わせを行い，付与された単語数が 5,000 を超えた時点でその時行われているバッチを終了するという設定とした．

2.4.1 実験結果

点予測の固有表現抽出に対する能動学習の有効性

ランダムにプール U から単語を選択し，ラベルの問い合わせを行う点予測モデルと能動学習を適用し Least Confident に基づいてラベルの問い合わせを行う点予測モデルの比較を行った．抽出性能と単語に対するアノテーション数の関係を図 2.5 に示す．能動学習を適用した点予測では最初のプール U における一割強に対しアノテーションを行うだけで，全単語を用いて学習を行った能動学習を利用しないモデルと同等の抽出性能を実現した．すなわち，点予測に対して能動学習を適用することは，学習に必要な教師データの数を減らし，アノテータの負担を少なくすることに貢献出来ると思われる．

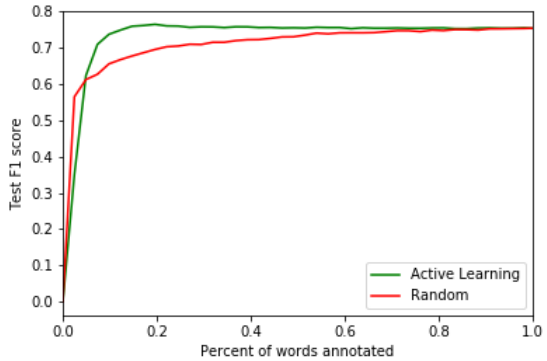

図 2.5: クエリ戦略による点予測の性能比較．x 軸はプール U におけるアノテーション済み単語の割合，y 軸は $F1$ 値

能動学習を適用した固有表現抽出モデルの性能比較

図 2.6 に既存手法と提案手法による固有表現抽出に対し，能動学習を適用したときの抽出性能とアノテーション単語数の関係を示した．提案手法である点予測は既存手法に比べ，ラベル付きデータ L が少ない場合において高い抽出性能を得られた．また本実験では LSTM-CNN-CRF モデルに対して Marginal Sampling を用いたが，点予測に対しても Marginal Sampling や更により多くの情報を得ることが出来る指標を適用することも可能であり，さらなる性能の向上も期待できる．しかし，提案手法はアノテーションされた単語が約 40%を超えると既存手法に比べ抽出性能が劣ることが確認できる．

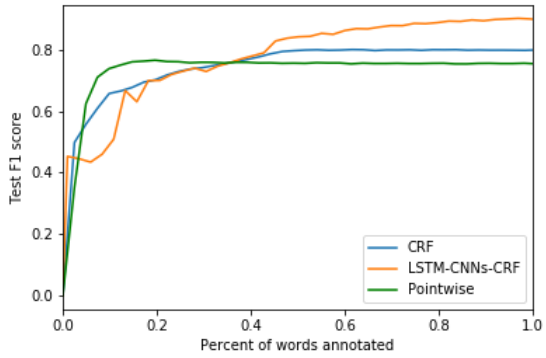

図 2.6: 既存手法と提案手法に対し，能動学習を適用した結果．x 軸はプール U におけるアノテーション済み単語の割合，y 軸は $F1$ 値

2.5 おわりに

本章では，点予測を用いた固有表現抽出モデルとそれに対する能動学習の適用手法の提案を行った．部分的アノテーションが利用可能な点予測に対して能動学習を適用することで，既存手法では不可能であった単語単位でのラベル問い合わせが可能になり，アノテーションコストを削減できた．また点予測は条件付き確率場や深層学習モデルに比べて，高速に学習が可能であるためバッチ型能動学習においてバッチのサイズを小さくしても，既存手法に比べてあまりコストを増加させることなく，より少ないラベル問い合わせ数で高い性能にたどり着けることが期待できる．

実験では無作為に選択した単語をアノテーションを行った点予測モデルに比べ，能動学習を適用した点予測モデルの方が少ないラベル付きデータで高い抽出性能を得られることを示した．また，既存手法の条件付き確率場や深層学習モデルに対して能動学習を適用した場合に比べても，提案手法はラベル付きデータが少ない状況において，比較的高い抽出性能を達成することが分かった．

しかし，ラベル付きデータの数が増加するに連れて，既存手法と比べ抽出性能が下がるという課題が本実験から分かった．能動学習を適用するかどうかに関わらず，固有表現抽出タスクにおいて十分な教師データが与えられる場合，点予測のようなラベルに対して独立性を仮定するモデルに比べて，マルコフ性を仮定する CRF や深層学習の方が抽出性能が高い傾向にある．このため，今後は部分的アノテーションコーパスが利用可能かつ能動学習に耐えうる高速な学習が可能な深層学習モデルについて検討していく必要がある．

第3章　遠距離教師あり学習

3.1　固有表現抽出における遠距離教師あり学習の問題

第一章でも述べたように一般的なドメインだけでなく専門分野においても，そのドメインに関する辞書が入手可能なことが多い．そのため近年辞書を学習資源として利用できるモデルの研究が行われている．辞書を利用した最も簡単な固有表現抽出手法として，文字列マッチングを用いた手法が挙げられる．この手法では抽出対象となる文章中の文字列が辞書に含まれる固有表現と一致する場合固有表現とし，辞書に含まれない単語については固有表現ではないとみなす抽出手法である．しかし，文字列マッチングによる固有表現抽出は辞書に含まれる固有表現しか抽出出来ないという問題に加え，間違った抽出を行うといった問題が存在する (図 3.1).

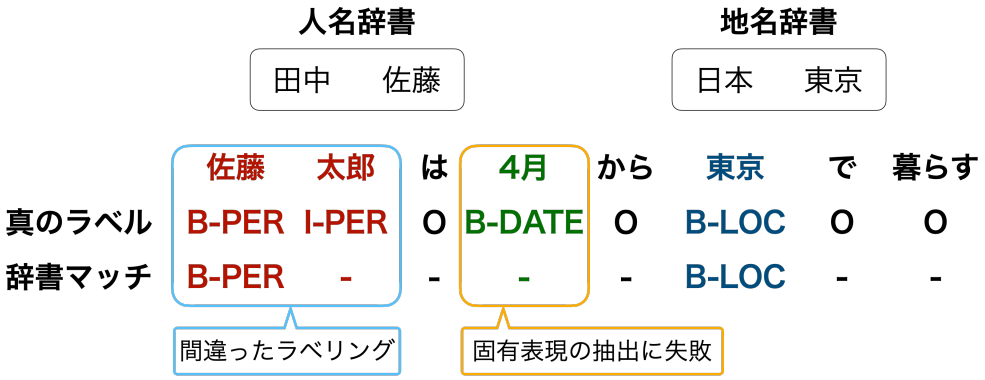

図 3.1: 辞書によるラベリングの失敗例

そのため，生コーパスに対して辞書に含まれる単語を文字列マッチングによってラベル付けを行い，遠距離教師データを作成する遠距離教師あり固有表現抽出モデル [21, 22] や PU 学習による固有表現抽出モデル [23] 等が提案されている．

固有表現抽出における遠距離教師あり学習の問題点は大きく分けて二つ存在する．1 つ目は表記ゆれや略称等が原因で文字列マッチングによるラベリングが失敗することによって再現率が低下するという点が挙げられる．この問題については Jie ら [24]，辰巳ら [25] が取り組んでいる．2 つ目の問題は，文字列マッチングによる間違ったラベリングが遠距離教師データに含まれてしまうによる適合率の低下である．本研究では，この 2 つ目の問題について着目し，複数のモデルを用いて学習と相互ラベリングを繰り返し行うことで，辞書マッチにおけるラベリングの誤りを減少させる手法を提案する．また提案手法による辞書マッチの誤りの考慮したモデルによる適合率の向上の可能性について検証する．

3.2 先行研究

一般的な CRF では，損失関数の算出に正解の系列のスコア $s(x, y)$ を利用する．そのため，一意に正しい系列が定まっていない部分的アノテーションコーパスを教師データとして利用することは出来ない．そこでラベル列の一部が欠損している場合においても，学習が行えるように一般化した Fuzzy CRF について説明する．

Fuzzy CRF は損失関数を拡張することで部分的アノテーションコーパスを学習可能にするモデルである．与えられた教師データにおいて，部分的アノテーションによって一部制約された取りうることが可能なラベル集合を $\mathcal{Y}_{possible}$ とする．ラベル集合 $\mathcal{Y}_{possible}$ における確率の和は次のように求めることが出来る．

$$\mathrm{score}_{\mathrm{FuzzyCRF}}(x, y) = P(y|x) = \frac{\sum_{y' \in \mathcal{Y}_{possible}} \exp\left(s(x, y')\right)}{\sum_{y' \in \mathcal{Y}_x} \exp\left(s(x, y')\right)} \tag{3.1}$$

この確率に対して，負の対数を取ったものが損失関数となる．

$$\begin{aligned} L_{\mathrm{FuzzyCRF}}(y) &= -\log \mathrm{score}_{\mathrm{FuzzyCRF}}(x, y) \\ &= \log \sum_{y' \in \mathcal{Y}_x} \exp\left(s(x, y')\right) - \log \sum_{y' \in \mathcal{Y}_{possible}} \exp\left(s(x, y')\right) \end{aligned} \tag{3.2}$$

部分的アノテーションコーパスに対して，学習した際の具体的な例を図 3.2 に示す．この図では「太郎は筑波出身だ」という文章中の「太郎」と「出身」に対してタグが付与されている．学習時にこれら二つのタグを通るように，それ以外のアノテーションがされていない単語に対しては全ての固有表現タグを取る可能性を加味して学習を行う．

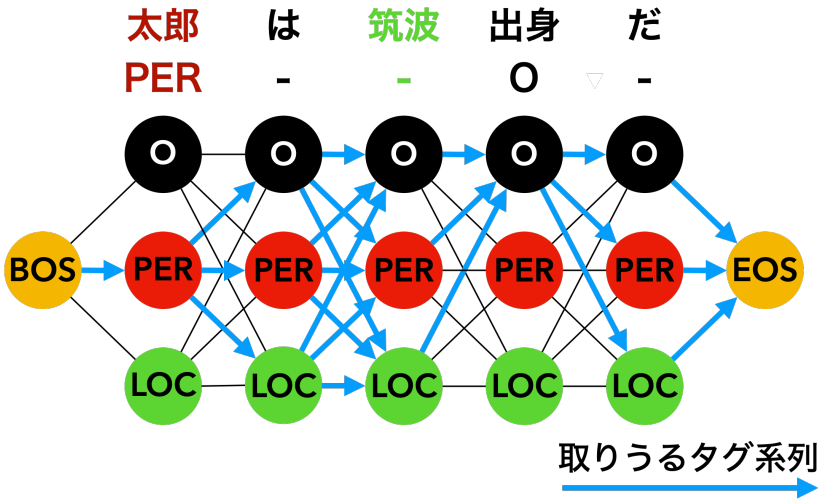

図 3.2: 部分的アノテーションコーパスにおける Fuzzy CRF の学習例

先程説明した Fuzzy CRF は古典的な機械学習モデルである CRF を単に拡張したものであり，LSTM-CRF の CRF 層に置き換えることが出来る．対して LSTM-CRF を用いて，部分的アノテーションコーパスを元に学習を行う深層学習モデルとして Jie ら [24] の手法について説明する．Jie らは人が現実でアノテーションを行うとき，固有表現ではないというタグを付ける行為は不自然であり，本来は固有表現にのみアノテーションが行われるという仮説を提唱した．Jie らの仮説の元で作成された教師データは，アノテータが知っている固有表現にのみアノテーションが付与されている部分的アノテーションコーパスであり，O タグ

は存在しない．このようなアノテーションコーパスを学習するために，Jie らは次のようなモデルを提案した (図 3.3)．このモデルは次のような手続きによって実行される．

1. 部分的アノテーションコーパスの欠損しているタグに対し，O タグを付与してアノテーションコーパスを作成する．
2. 作成したアノテーションコーパスを二分割する．
3. 分割したアノテーションコーパスそれぞれに対して，BiLSTM-CRF を構築し学習を行う．
4. BiLSTM-CRF を利用して，学習に利用しなかったのコーパスのラベルを予測し，新しいアノテーションコーパスとして利用する．この際，最初の部分的アノテーションコーパスに付けられたタグは必ず通るように制約付きビタビアルゴリズムを適用する．
5. 3, 4 を複数回実行する．
6. 分割したアノテーションコーパスを結合する．これを最終的なアノテーションコーパスとする．
7. 最終的なアノテーションコーパスを教師データとして，BiLSTM-CRF を学習する．

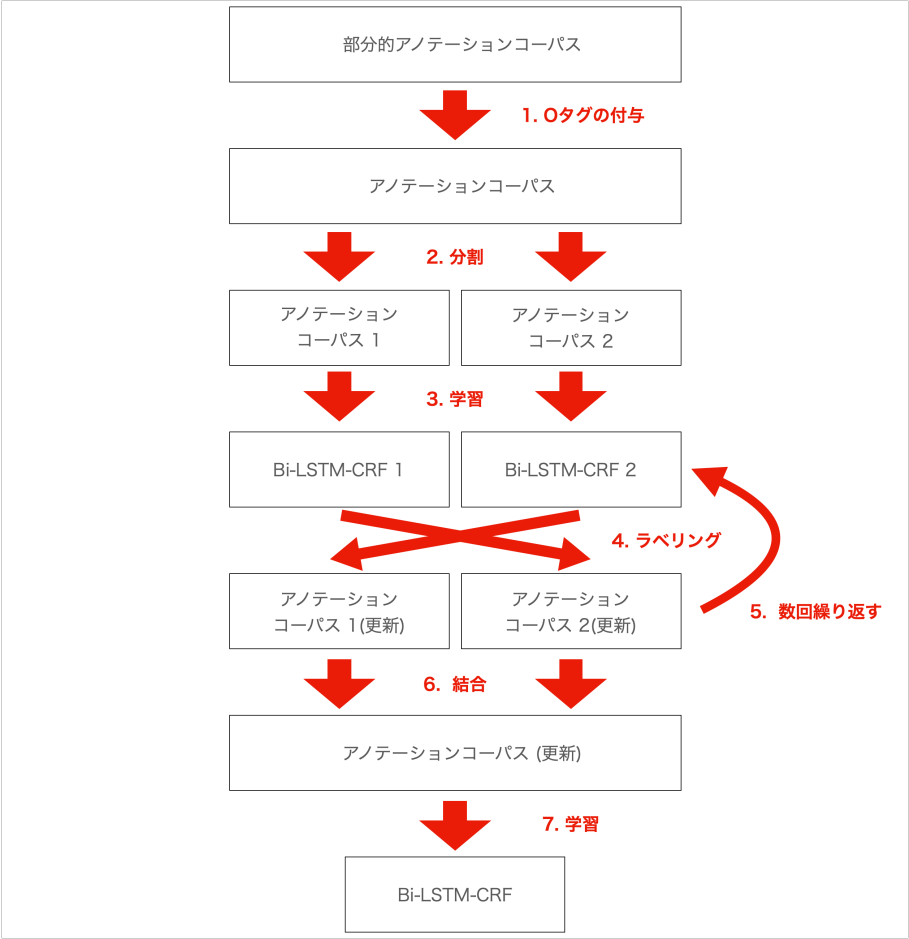

図 3.3: Jie らの提案手法のアーキテクチャ

3.3 提案手法

Jie らが提案したモデル（以下，Jie's method）は部分的アノテーションコーパスに対して学習可能だが，人手によってアノテーションが付与されていることを前提としている．そのため，アノテーションが失敗している可能性については考えられていない．辞書によって自動的にラベリングが行われることで作成された部分的アノテーションコーパスには，固有表現抽出における遠距離教師あり学習の問題でも述べたように，間違ったタグが付与されている可能性がある．そこで本章では辞書を用いて作成された部分的アノテーションコーパスに対して，間違ったラベルを付与する可能性を考慮したモデルを提案し，既存手法に対して遠距離教師ありデータを学習させたときの性能比較を行う．

制約付きビタビアルゴリズムを利用すると，各モデルによる相互アノテーションにおいても最初の辞書マッチによって間違ったタグが付与されたままになる．そこで提案するモデルでは，制約付きビタビアルゴリズムではなく通常のビタビアルゴリズムを利用する．しかし，通常のビタビアルゴリズムを用いると辞書に含まれるが，生コーパスには出現頻度が少ないような固有表現が学習中に失われてしまう可能性が高い．そのため，提案手法ではモデルのタグ予測に加えて，辞書による文字列マッチングを後処理を加えたものを最終的な予測とする．すでにモデルによってタグが付与された文章に対して後処理辞書マッチングを行った場合，単語に対する以下の 3 パターンのタグ競合が起こる．

- モデルによる予測では固有表現であり，辞書中にも同じ表現が含まれる．
- モデルによる予測では固有表現であるが，辞書中にその表現は存在しない．
- モデルによる予測では固有表現ではないが，辞書中にはその表現が存在する．

今回の実験では図 3.4 のように，最初のパターンのときのみ辞書マッチによる後処理によって付与されたラベルを採用し，それ以外の場合にはモデルによるタグ予測を採用することにした．

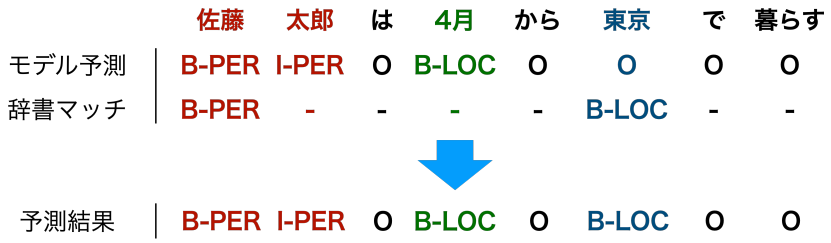

図 3.4: モデルによる予測と辞書マッチングの統合

まとめると，本手法は次のような手順によって固有表現抽出を行う．

1. 生コーパスに対して，辞書を用いてラベル付けを行い，部分的アノテーションコーパスを作成する．
2. 部分的アノテーションコーパスを遠距離教師ありデータとして，Jie's method を制約なしビタビアルゴリズムで実行する．
3. 後処理の辞書マッチングを行い，最終的なタグを決定する．

3.4 実験

3.4.1 データセット

今回は辞書が整備されてあるデータセットとそうではないデータセットである以下の二つを対象に実験を行った.

BC5CDR

BC5CDR は全 15,000 記事で構成された 15,935 個の Chemical（薬品名）と 12,853 個の Disease（病名），計二種類の固有表現を持つデータセットである．生コーパスと辞書については Shang ら [26] が公開しているものを利用した.

CoNLL-2003

CoNLL-2003 は新聞記事から作成された固有表現抽出データセットであり，LOC（地名）, ORG（組織名）, DATE（日時・時間表現）, MISC（その他）の四種類の固有表現がアノテーションの対象になっている．CoNLL-2003 では教師データと二つのテストデータが用意されている．今回の実験では教師データからアノテーションを削除したものを生コーパスとし，二つのテストデータのうち評価に利用しない方に含まれる固有表現を元に辞書を構築した.

3.4.2 比較手法

比較手法には次の 4 手法を選んだ．比較手法のいくつかには部分的アノテーションコーパスに対応出来ない手法も含まれるので，辞書と生コーパスによる遠距離教師データが活用出来るようにデータの生成過程に一部工夫を加えている.

文字列マッチング

辞書に含まれる固有表現を直接文字列マッチングによって抽出する手法である．文字列マッチングを用いる手法では，文章中に含まれる単語に対して複数の固有表現がマッチすることがある．この場合，マッチングした固有表現の中で最も長い文字列のものを選択する最長一致法を採用した.

LSTM-CRF

本来 Lample ら [27] の LSTM-CRF は部分的アノテーションコーパスを学習することが出来ない．そこでまず生コーパスに対して辞書マッチによるラベル付与を行い，部分的アノテーションコーパスを作成する．部分的アノテーションコーパスのラベルが欠損している部分は O タグとみなして，学習を行った.

Fuzzy-LSTM-CRF

LSTM-CRF と同様にまず辞書マッチを用いて生コーパスにラベル付与を行う．その後，作成した部分的アノテーションコーパスが学習出来る Fuzzy-CRF を用いて直接学習を行う.

Jie's method　先行研究にて説明した Jie's method を辞書マッチングによって生成した部分的アノテーションコーパスに対して適用する．また提案手法と同様にモデルによるタグ予測と辞書によるラベリングの組み合わせによる抽出性能の向上についても検証を行う.

BiLSTM-CRF のハイパーパラメータに関しては Lample ら [27] の論文を，提案手法と Jie's method のハイパーパラメータについては Jie ら [24] の論文を参考に設定した.

3.4.3 実験結果

表 3.1: 各モデルにおける性能比較

手法	BC5CDR			CoNLL-2003		
	Precision	Recall	F1	Precision	Recall	F1
辞書マッチング	**87.8**	61.7	72.4	49.2	29.3	33.8
BiLSTM-CRF	88.4	61.8	72.6	72.5	39.2	50.9
Fuzzy-LSTM-CRF	11.2	**85.5**	19.3	18.1	**80.0**	26.4
Jie's method	81.4	73.7	77.3	77.7	49.1	57.5
Jie's method + 後処理辞書マッチ	81.4	73.8	**77.3**	77.7	49.1	**57.5**
提案手法	83.3	69.3	75.6	78.7	47.8	57.0
提案手法 + 後処理辞書マッチ	83.4	71.7	77.1	**78.7**	47.8	57.0

実験結果は表 3.1 の通りである．まず辞書マッチングの結果を見ると，整備済みの辞書が存在する BC5CDR データセットでは高い適合率が確認できる．しかし CoNNL-2003 では辞書の性能が悪いため再現率が低い．その上 4 種類の固有表現がデータセット中に存在し，同じ単語でも文脈によって異なるタグが付与されることが多いため適合率も低いという結果になった LSTM-CRF は辞書マッチングに比べて，全体的に抽出性能が高い．特に CoNNL-2003 においては大きな性能向上が見られる．これは学習を行うことで教師データに出現しない固有表現を抽出出来たことや，文脈を考慮したタグ予測が行われた結果だと思われる．対して Fuzzy-LSTM-CRF は両データセットにおいて，適合率がかなり低い．これは辞書マッチによって生成された部分的アノテーションコーパスには O タグが存在しないため，ラベルを予測する際に O タグの可能性を殆ど考慮しなくなってしまったからだと考えられる．提案手法では，両データセットに対して Jie's method より提案手法のほうが適合率が高いという結果になった．これは部分的アノテーションコーパスの作成時に起きた辞書マッチングによる間違って付与されたタグが，ビタビアルゴリズムを用いた複数回にわたる相互アノテーションにより正しいタグに修正されたからだと考えられる．しかし，ビタビアルゴリズムにより辞書マッチングによる正しく付与されたタグに対しても修正が行われたため再現率が低下し，F1 値は Jie's method よりも低い結果となった．また Jie's method ではモデルの予測結果と辞書マッチングの統合を行っても抽出性能は殆ど変わらなかった．これは制約付きビタビによるラベリングの影響で辞書に含まれる固有表現を取り逃すことが少なかったからだと思われる．

各データセットについてみると，CoNLL-2003 においては提案手法の適合率は辞書マッチングより高いものになった．理由として CoNLL-2003 の実験で利用した辞書はテストデータから生成されており，LOC と ORG に同じ単語が含まれていることが多かった．そのため辞書マッチングによる抽出では異なる固有表現へのクラス分類が起こり，適合率が下がったと考察される．まとめると辞書マッチングの精度が不安定な場合，つまり辞書があまり整理されていないとき，適合率を重視した目的において提案手法は有効であると考えられる．ただし，BC5CDR のような辞書マッチングがある程度機能しているデータセットにおいては，提案手法はあまり有効的ではないと言える．

3.5 おわりに

本章では遠距離教師あり固有表現抽出手法において，辞書マッチの誤りを考慮することでパフォーマンスが向上するか検証した．その結果，辞書がきちんと整備されているときにはそもそも辞書マッチの誤りは起きにくく，あまり効果を発揮でないことが分かった．対して，辞書自体の性能があまり良くなく辞書マッチが失敗しやすいときに適合率の向上が見られた．

第4章　クラウドソーシング

4.1　固有表現抽出におけるクラウドソーシングの問題

近年，機械学習における教師データを作成する目的でクラウドソーシングを用いてアノテーション依頼を行う機会が増えている．クラウドソーシングとは，作業の品質が高いとは限らない不特定多数の人，ワーカに作業を委託することである [28]．その性質上高い採用コストと人件費を払い雇ったアノテータに比べるとアノテーションの品質は低く，ばらつきも大きい．そのため同一のタスクに対して複数のワーカを割り当て，ラベルを集約することで品質の高いラベルを得ようとする手法がいくつも提案されている．Dawid と Skene は [29] は複数の医者による診断結果からそれらをまとめた最終的な診断結果を導くという文脈で考案された．この手法はクラウドソーシングの文脈においてアノテーションの集約に応用されており，ワーカが付与したラベルを観測値，真のラベルを潜在変数とみなすことで生成モデルを仮定し，真のラベルを推定している．Whitehill ら [30] は真のラベルに加え，各ワーカの能力と各タスクの難易度も潜在変数として導入し，推論の対象とする GLAD モデルを提案した．しかし，これらの提案手法の殆どは二値分類や多値分類タスクへの適用を前提としている．対して，固有表現抽出は系列ラベリングタスクであり，多値分類を前提としたこれらの既存手法を適用しても系列の性質を捉えることは出来ない．そのため集約性能の低下が起こる．また，自然言語において同じ単語が異なる文章に出現することはよくあり，単語の持つ情報は大きい．なので固有表現抽出において，アノテーションだけを集約に使うのではなく，集約対象となる全文章を通した各単語とラベルの関係を見ることも重要だと考えられる． しかし，単純に多数決を取るようなモデルや Dawid らが提案した集約方法では，全単語について独立性を仮定するため図 4.1 のように集約したラベルと真のラベルが大きく異なってしまう可能性が高い．Nguyen ら [31] は Dawid らのモデル適用した後，Multinomial HMM を用いたアプローチを適用することでこれらの問題を解決し，系列ラベリングタスクにおいて多数決や Dawid らのモデルを用いた集約より高い性能を持つことを示した．しかし Multinomial HMM はパラメータ数が少なく，モデルとしての表現力は低い．

本研究では，このような系列ラベリングタスク固有のラベル列の集約問題について，いくつかの既存モデルと提案手法について比較を行い，モデルの複雑性と集約性能の関係について議論する．

4.2　先行研究

まず，多値分類タスクにおける集約モデルとして Dawid と Skene による手法 [29](以下，Dawid-Skene モデル) について説明する．K 個のクラスに属する N 個のデータが存在し，全 J 人のワーカによってアノテーションが行われたとする．このとき，データ $n \in \{1, \ldots, N\}$ に対して，ラベルを付与したワーカの集合を $\mathcal{J}_n \subseteq \{1, 2, \ldots, J\}$ とする．ここでワーカ j の能力を示す混同行列 $C^{(j)} \in \mathbb{R}^{K \times K}$ を潜在変数として導入する．行列の要素 $C^{(j)}_{a,b}$ は真のクラ

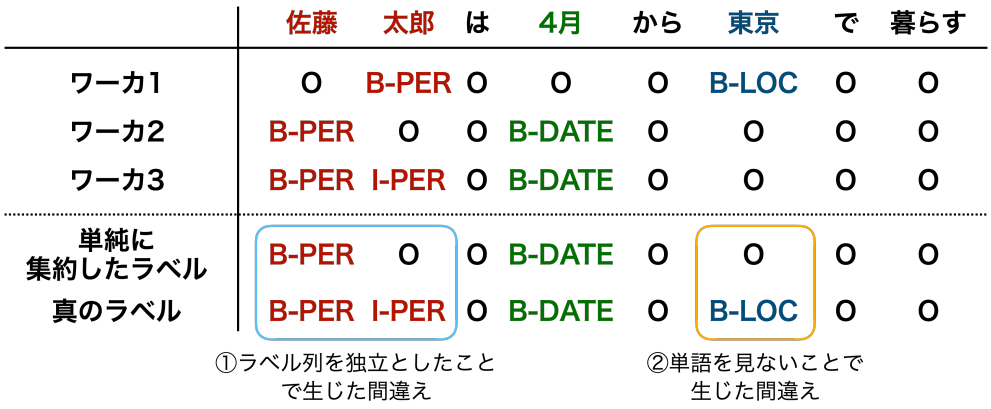

図 4.1: 固有表現抽出における集約の失敗例

スが a であるデータに対し，ワーカ j がクラス b とアノテーションを行う確率を表している（図 4.2）．

ワーカjが予測したラベル

真のラベル	B-LOC	I-LOC	B-PER	I-PER	O
B-LOC	0.8	0.0	0.1	0.0	0.1
I-LOC	0.15	0.75	0.0	0.0	0.1
B-PER	0.1	0.0	0.9	0.0	0.0
I-PER	0.0	0.0	0.4	0.6	0.0
O	0.1	0.0	0.1	0.0	0.8

図 4.2: ワーカ j の混同行列の例

例えば，混同行列における対角成分はワーカが各クラスに正しいラベルを付与する確率である．混同行列 $C^{(j)}$ は各ワーカ j と真のクラス k に対して，ディレクレ分布によってサンプリングされる．ここで $\boldsymbol{\alpha}$ はそれぞれの要素が正の実数値である K 次元のパラメータである．

$$C_{k,\circ}^{(j)} \sim \mathrm{Dirichlet}(\boldsymbol{\alpha}) \tag{4.1}$$

混合比率パラメータをディレクレ分布から生成される値 $\boldsymbol{t}$ としたとき，真のラベル $h_n \in \{1, \ldots, K\}$ はカテゴリカル分布から生成される．

$$
\begin{aligned}
h_n|\boldsymbol{t} &\sim \text{Categorical}(\boldsymbol{t}) \\
\boldsymbol{t} &\sim \text{Dirichlet}(\boldsymbol{\beta})
\end{aligned}
\tag{4.2}
$$

ただし，ここで $\boldsymbol{\beta}$ は $\boldsymbol{\alpha}$ と同じ制約を持つパラメータであり，このとき $\boldsymbol{t}$ は $\sum_{k=1}^{K} t_k = 1$ を満たす．これらの確率変数を利用すると，ワーカ j がデータ n に付与したラベル l_n^j は次のように表すことができる．

$$
l_n^j|h_n \sim \text{Categorical}(\boldsymbol{C}_{h_n}^{(j)}) \tag{4.3}
$$

このとき観測値はワーカから与えられたアノテーション $\{l_n^j\}_{j\in\mathcal{J}_n}$ であり，これを元に真のラベル h_n を推測するのが Dawid-Skene モデルの目的である．一連の生成過程をグラフィカルモデルにして表すと，図 4.3 のように書ける．

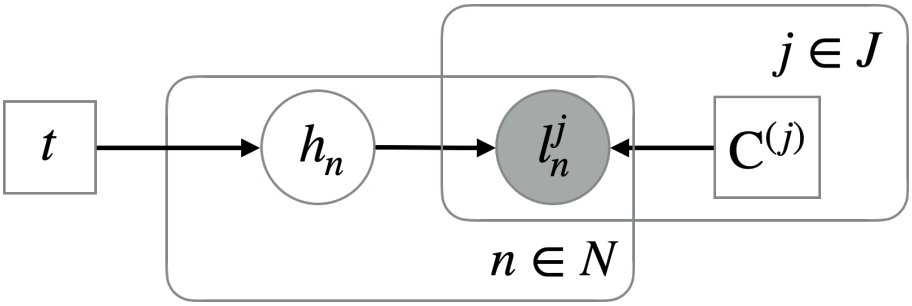

図 4.3: Dawid-Skene モデルのグラフィカルモデル

Dawid-Skene モデルはこれら真のラベル h_n とワーカの混同行列 C を推定するために，それぞれのパラメータを相互に固定し期待値を最大化させる EM アルゴリズムを用いて推論を行う．これは解析的に求めることが出来る．

Dawid-Skene モデルが各ラベルを独立に扱ったのに対して，Nguyen ら [31] は HMMs with Crowd Workers（以下，HMM-Crowd）は一次マルコフ性を仮定することにより系列の並びを考慮する遷移確率と単語の生成確率を導入したモデルである．ここで遷移パラメータを $\boldsymbol{\tau}_{h_i}$，出力パラメータを $\boldsymbol{\Omega}_{h_i}$ としたとき，遷移確率と生成確率は次のように生成される．

$$
\begin{aligned}
h_{i+1}|h_i &\sim \text{Categorical}(\boldsymbol{\tau}_{h_i}) \\
w_i|h_i &\sim \text{Multinomial}(\boldsymbol{\Omega}_{h_i})
\end{aligned}
\tag{4.4}
$$

遷移パラメータ $\boldsymbol{\tau}_{h_i}$，出力パラメータ $\boldsymbol{\Omega}_{h_i}$ の初期値は事前に Dawid-Skene モデルを用いて集約されたタグを元に設定される．これらの確率変数からワーカ j がデータ n に付与したラベル l_n^j は次のように表すことが出来る．これは Dawid-Skene モデルと同じように計算可能である．

$$
l_n^j|h_n \sim \text{Categorical}(\boldsymbol{C}_{h_n}^{(j)}) \tag{4.5}
$$

ただし Dawid と Skene[29] は混同行列を K 次正方行列としたのに対し，Nguyen ら [31] は HMM-Crowd における混合行列を固有表現かそうではないかだけを扱う二次正方行列として学習を行った（図 4.5）．

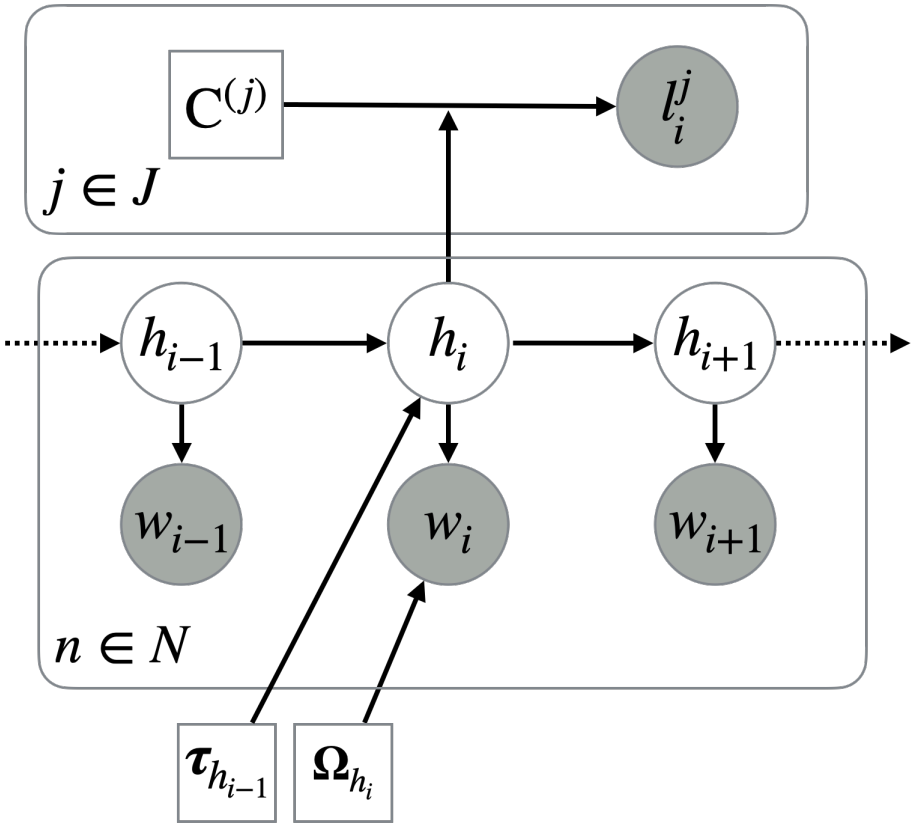

図 4.4: HMM-Crowd モデルのグラフィカルモデル

ここで Dawid-Skene モデルにおける混同行列との混乱を避けるため，以降 $C^{(j)} \in \mathbb{R}^{K\times K}$ である混合行列を Confusion Matrix，$C^{(j)} \in \mathbb{R}^{2\times 2}$ である混合行列を Binary Confusion Matrix と呼ぶことにする．また HMM-Crowd では EM アルゴリズムの代わりに変分ベイズ法を用いて各パラメータの推定を行っている．

4.3 提案手法

本研究では大きく分けて二つの実験を行った．1 つ目の実験では単語の生成確率に対して，二つのより複雑な確率分布を仮定したモデルを提案し，表現力と集約性能の関係性について議論を行う．2 つ目の実験にて集約の際に単語の確率分布を考慮することにどのような意味を持つのか HMM-Crowd の生成過程を一部変更し，性能の違いを確認する．

まず 1 つ目の実験で用いる提案手法として，単語埋め込みと多次元ガウス分布を利用したモデルである HMM-Crowd with Word Embedding について説明する．HMM-Crowd では単語の出力確率に多項分布を導入することで単語の生成確率を求めた．対する HMM-Crowd with Word Embedding では，単語 w_i に対応する単語埋め込み $v_{w_i} \in \mathbb{R}^d$ に対して多次元ガウス分布の生成を仮定することでより豊かな表現を期待する．真のラベル h_i に対する平均パラメータ $\boldsymbol{\mu}_{h_i} \in \mathbb{R}^d$ と共分散行列 $\boldsymbol{\Sigma}_{h_i} \in \mathbb{R}^{d\times d}$ を用いて，多次元ガウス分布による単語埋め込みの生成確率を次のように定義すると．グラフィカルモデルは図 4.6 のように表現できる．またここでの $\boldsymbol{\Omega}_{h_i}$ は HMM-Crowd と同様遷移パラメータを表す．

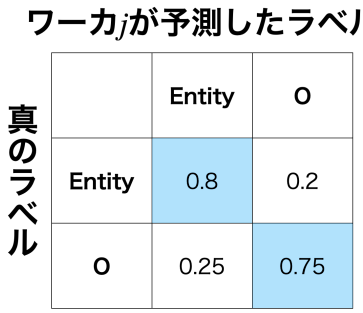

図 4.5: ワーカ j のバイナリ混同行列の例

$$v_{w_i}|h_i, w_i \sim \mathcal{N}(\boldsymbol{\mu}_{h_i}, \boldsymbol{\Sigma}_{h_i}) \tag{4.6}$$

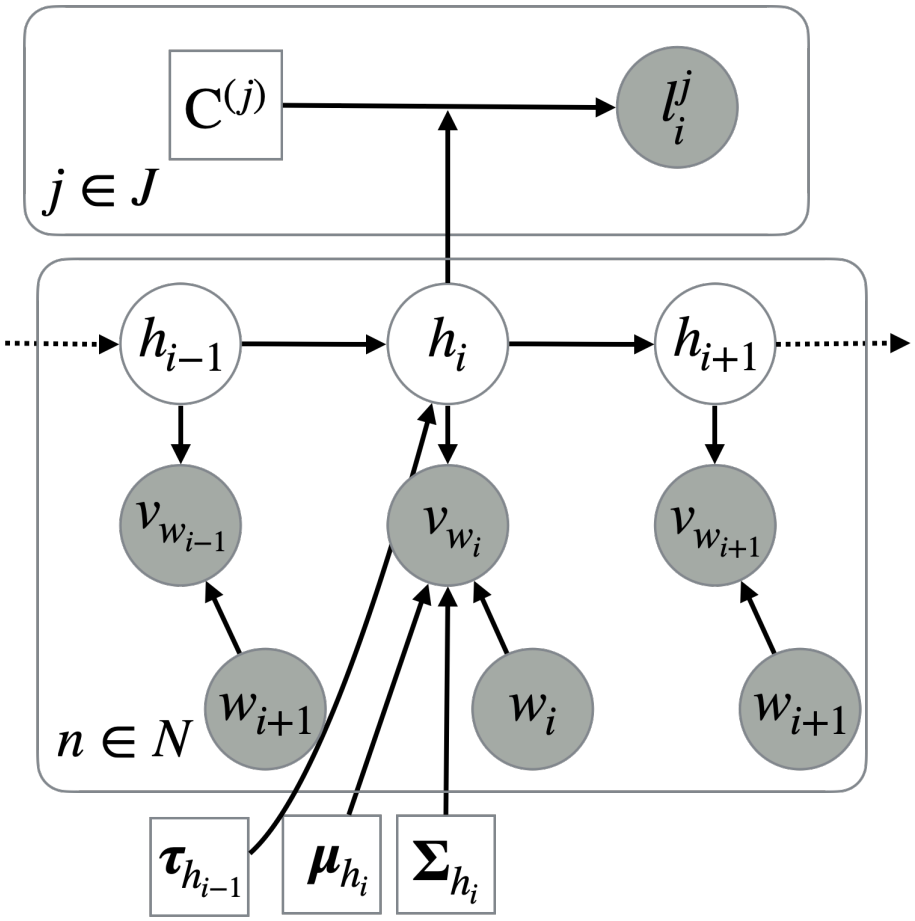

図 4.6: HMMs with Crowd Workers and Word Embedding モデルのグラフィカルモデル

もう一つの提案手法として，文字レベルの条件付き言語モデルを用いて単語の生成確率を計算する HMM-Crowd with Character Language Model について説明する．文字レベル言語モデル (Character Language Model) は再帰的ネットワークを用いたニューラル言語モデルである [32]．L 文字の単語 w_i に対応する文字列を $\boldsymbol{c}^{(i)} = (c_1^{(i)}, \ldots, c_L^{(i)})$ とし，それに対応する文字レベルの単語埋め込みを $\boldsymbol{v}_{c^{(i)}} = (v_{c_1^{(i)}}, \ldots, v_{c_L^{(i)}})$ とする．このとき真のラベル h_i

に対応する分散表現 v_{h_i} によって条件付けられた言語モデルは LSTM を用いて次のようにモデリングされる．

$$
\begin{aligned}
p(\hat{c}_{l+1}^{(i)}|\hat{c}_{1:l}^{(i)}, h_i) &= f(\mathrm{LSTM}(v_{\mathrm{cat}})) \\
v_{\mathrm{cat}} &= [v_{\hat{c}_l^{(i)}}; v_{h_i}] \\
\hat{c}_l^{(i)} &\sim p(\hat{c}_l^{(i)}|\hat{c}_{1:l-1}^{(i)}, h_i)
\end{aligned}
\tag{4.7}
$$

ここでの f は LSTM の状態から確率分布への写像である．c_0 を単語の始まりを表す特殊トークンとすると，単語の生成確率は Character Language Model を用いて計算できる．

$$
w_i|h_i \sim p(c_0|h_i) \prod_{l=1}^{L} p(\hat{c}_l^{(i)}|\hat{c}_{0:l-1}^{(i)}, h_i) \tag{4.8}
$$

Character Language Model のアーキテクチャは図 4.7 の通りであり，グラフィカルモデルを図 4.8 に示す．

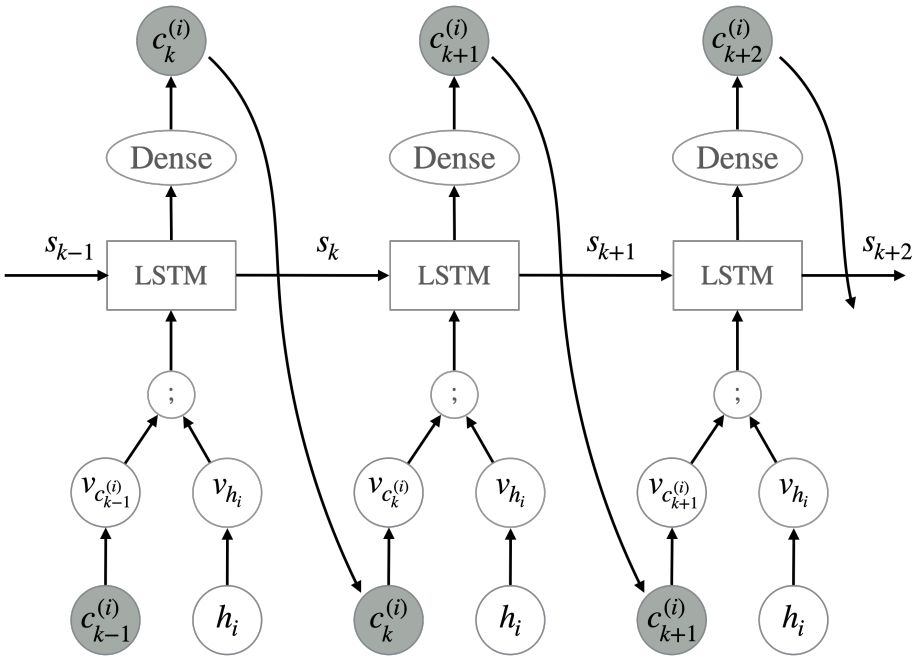

図 4.7: 条件付き文字レベル言語モデルのアーキテクチャ

Dawid-Skene モデルと同様 HMM-Crowd with Character Language Model でも EM アルゴリズムにて学習を行う．一般的な言語モデルの学習では損失関数に Cross Entropy Loss を用いる．このとき損失 J は単語 w_i の損失関数 $J^{(i)}$ の和として，次のように計算される．

$$
\begin{aligned}
J^{(i)} &= -\sum_{l=1}^{L} c_l^{(i)} \log \hat{c}_l^{(i)} \\
J &= \sum J^{(i)}
\end{aligned}
\tag{4.9}
$$

しかし，HMM-Crowd with Character Language Model は真のラベル h_u によって条件付けられており，今回のタスクにおいて真のラベル h_i は観測することが出来ない．そのため

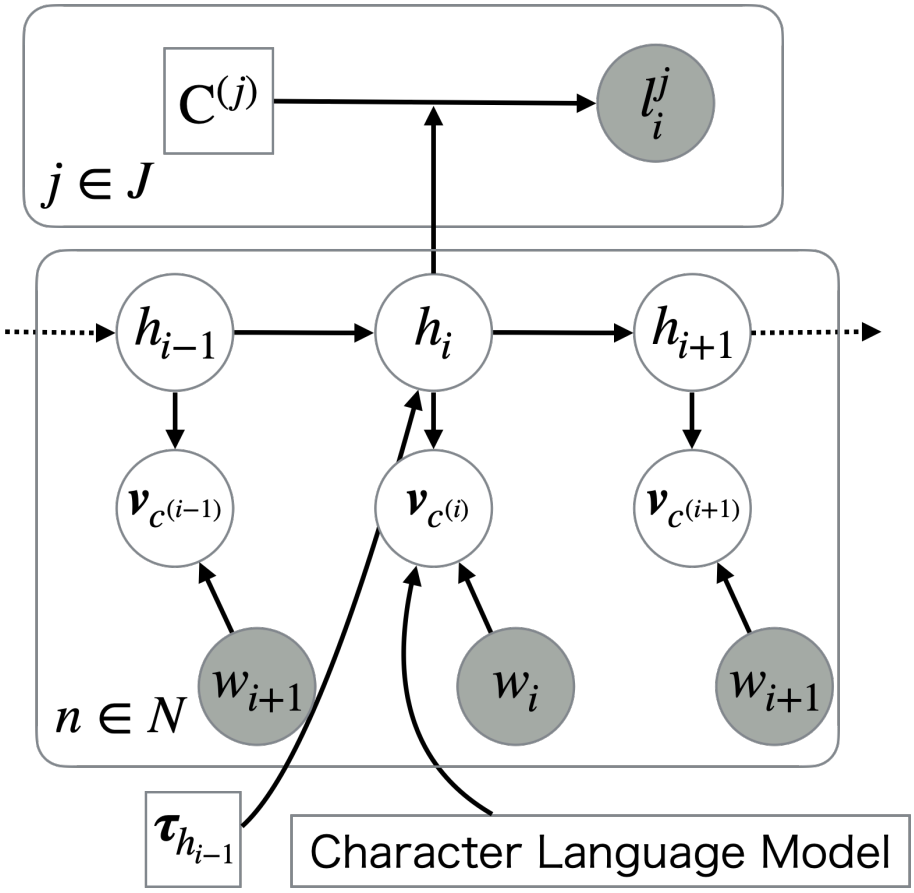

図 4.8: HMM-Crowd with Character Language Model のグラフィカルモデル

EM アルゴリズムにおける，M ステップにて推定された $p(h_i|w_i)$ によって重み付けした Cross Entropy Loss を損失関数とする．

$$
\begin{aligned}
J^{(i)} &= -\sum_{k=1}^{K} p(h_i|w_i) \sum_{l=1}^{L} c_l^{(i)} \log \hat{c}_l^{(i)} \\
J &= \sum J^{(i)}
\end{aligned}
\tag{4.10}
$$

次に，2 つ目の実験で用いるモデルについて説明する．Nguyen らが提案した HMM-Crowd では遷移パラメータ $\boldsymbol{\tau}_{h_i}$，出力パラメータ $\boldsymbol{\Omega}_{h_i}$ を用いることで一次マルコフ性を考慮している．これは各ワーカが単語 w_i を観測して付与したアノテーション l_i^j の情報に加え，更に真のラベル h_i によって単語 w_i の出力する確率を考慮しているということである．そのため単語 x_i が与える影響が，真のラベル h_i の遷移が与える影響に比べて大きく，性能を低下させている要因となっている可能性がある．そこで遷移確率を考慮した Dawid-Skene モデルを提案する．このモデルでは単語 w_i の出力確率 $p(w_i|h_i)$ は既にワーカによるアノテーションによって表現されていると考え導入しない (図 4.9)．また，HMM-Crowd では Binary Confusion Matrix を用いていたが，通常の Confusion Matrix を利用することでどの程度性能に影響が出るかも確認する．

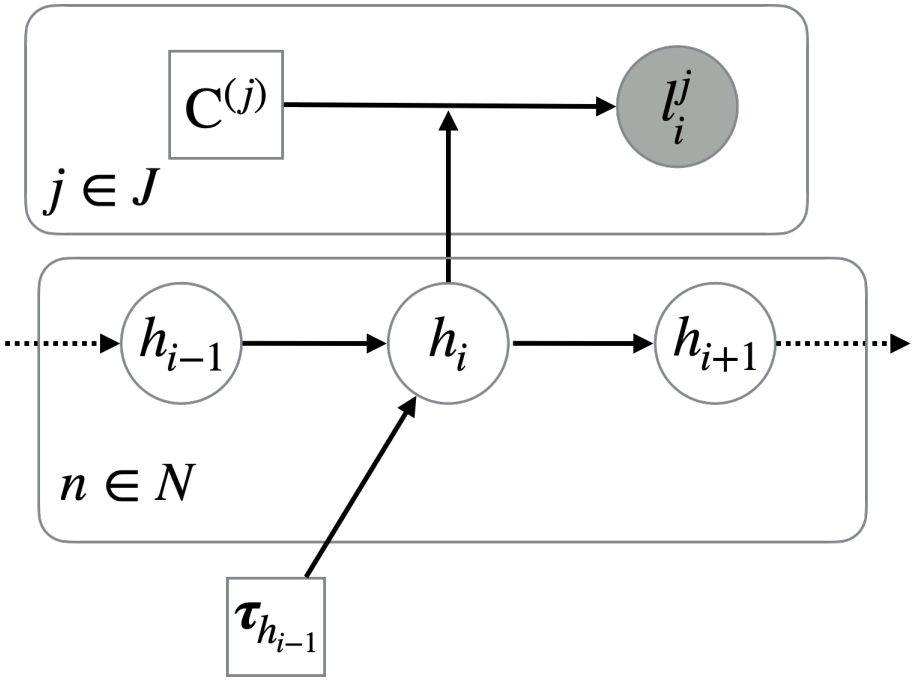

図 4.9: 遷移確率を考慮した Dawid-Skene モデル

4.4 実験

今回の実験では Rodrigues ら [33] が提供している計 47 人のワーカによってアノテーションされた CoNLL-2003 のデータセットを利用し，モデルの集約性能を計測した．まず 1 つ目の実験である既存手法と提案手法における集約性能についての調査の結果を述べる．HMM-Crowd with Word Embedding の学習では単語埋め込みの生成分布に多次元ガウス分布を仮定し，教師無し品詞予測を行った Lin ら [34] の実験を参考に平均パラメータ $\boldsymbol{\mu}_{h_i}$ のみを学習し，共分散行列 $\boldsymbol{\Sigma}_{h_i}$ は $\boldsymbol{\Sigma}_t(k,k) = 0.45 \quad (^\forall k \in \{1,\ldots,d\})$ で固定した．また事前学習済み単語埋め込みには 100 次元の Glove [35] を用いた．実験の結果を表 4.1 に示す．表からも分かるように再現率，適合率，F1 共に HMM-Crowd が高い．また単語の出力確率を扱う確率分布が複雑なモデルは，全体的に性能が低下していることが分かる．特に多次元ガウス分布を出力確率に利用したモデルの適合率は著しく低下している．

表 4.1: 各モデルにおける性能比較

手法	Precision	Recall	F1
Dawid-Skene	0.770	0.697	0.732
HMM-Crowd	**0**.777	**0**.745	**0**.760
HMM-Crowd with Word Embedding	0.006	0.007	0.007
HMM-Crowd with Character Language Model	0.373	0.499	0.427

2 つ目の実験では HMM-Crowd の出力確率とワーカーの混同行列の関係を確認する．実験の結果は表 4.2 の通りである．Dawid-Skene に対して Binary Confusion Matrix を導入するモデルは単語に対する出力確率の情報が殆ど無くなり意味をなさないため，今回の実験には含めていない．結果を確認すると，HMM-Crowd では Confusion Matrix を用いたモデルに比べて，Binary Confusion Matrix を混同行列に用いたモデルの方が性能が高かった．対

して Dawid-Skene モデルに対して遷移確率を導入したモデルは通常のモデルに比べて適合率が向上しているが，再現率は大きく下がっている．

表 4.2: Dawid-Skene モデルと HMM-Crowd における性能比較

手法	Emission	Transition	Confusion Matrix	Precision	Recall	F1
HMM-Crowd	✓	✓	$\mathbb{R}^{K\times K}$	0.759	0.740	0.750
	✓	✓	$\mathbb{R}^{2\times 2}$	0.777	**0**.745	**0**.760
Dawid-Skene	×	✓	$\mathbb{R}^{K\times K}$	**0**.781	0.579	0.665
	×	×	$\mathbb{R}^{K\times K}$	0.770	0.697	0.732

二つの実験から分かることとして，Dawid-Skene に対して遷移パラメータを導入すると，遷移確率の影響が大きく出てしまいワーカが行うアノテーションを無視した推定をしやすくなり性能が下がる．しかし，遷移付き Dawid-Skene に対して出力パラメータを加えた HMM-Crowd の性能は非常に高い．これは単語の出力確率は Dawid-Skene によって集約されたタグを元に初期値とするため，出力確率に多項分布を仮定したアノテーションから外れにくい上に遷移を考慮出来るモデルとなっていると考えられる．対して，HMM-Crowd with Word Embedding や HMM-Crowd with Character Language Model といった分散表現を利用したモデルは極めて性能が低い．これは両モデルにおいて出力確率の影響が大きすぎるため，ワーカの意見を無視した推定が増えたことが原因として考えられる．特に HMM-Crowd with Word Embedding では平均パラメータ $\boldsymbol{\mu}_{h_i}$ の学習だけでは十分な表現が獲得出来なかったことや，入力に対し事前学習済み分散表現を利用したことにより，分散表現中に存在しない未知語が出現してしまったという問題も原因の一つとして考えられる．

4.5 おわりに

本研究ではクラウドソーシングにおける系列ラベリング固有の集約問題についていくつかのモデルを提案し，調査を行った．その結果，単語の出力確率に多次元ガウス分布やニューラルネットワークのような複雑な確率分布を仮定すると遷移確率に比べ出力確率の影響が大きくなることで性能が低下することが分かった．しかし，これとは逆に出力確率に確率分布を仮定しない場合，遷移確率の影響が大きくなりすぎてワーカのアノテーションを無視した推定を行いやすいことが分かった．これらの点から HMM-Crowd の性能が優れているのは Dawid-Skene モデルによるラベル集約を用いて，安定しやすい多項分布の初期値を設定することによって，ワーカの意見から大きく外れないことが一つの要因だと考えられる．そのためワーカの意見から大きく離れすぎず，かつ豊かな表現が可能であるような分布を見つけることが今後の課題になると思われる．

第5章　おわりに

本研究では，専門家のアノテーション，外部知識を用いたラベル付与，クラウドソーシングの三つのアノテーション作成方法にて生じる，固有表現特有の問題に着目し，それらを解決するような手法の提案，調査を行った．第 2 章では固有表現抽出の学習には大量の教師データを必要とし，コーパスを作成するコストが負担になっているという問題について点予測による単語単位をクエリに用いた能動学習を提案し，有効性を示した．第 3 章では固有表現抽出における遠距離教師あり学習について，辞書を用いた生コーパスへの間違ったアノテーションの付与を問題として取り上げ，ラベリング誤りを考慮するモデルの提案と既存手法の比較を行った．実験の結果，辞書の整備が不十分で多くの誤りが発生する場合において，再現率の向上が見られた．最後に第 4 章にて固有表現抽出におけるクラウドソーシングでは，系列ラベリングタスク固有の集約で起きうる問題について述べ，生成モデルの複雑性と集約性能の関係について調査を行った．調査の結果，複雑なモデルを単語の出力分布に仮定すると出力確率が支配的になり遷移確率を無視したラベリングが増え，対して出力分布を導入しない場合遷移確率の影響が大きすぎるためアノテーションを無視することが分かった．教示方法のこれからの展開として BERT[36] を用いた転移学習や教師データと異なる言語にも利用可能な多言語に対応可能な固有表現抽出モデルによる教示コストの削減が考えられる．

謝辞

本研究の遂行においてご指導，ご鞭撻くださった若林啓准教授，研究に関する議論で大変お世話になりました手塚太郎准教授，指導教官を引き受けてくださった佐藤哲司教授に心から感謝します.

参考文献

[1] 戸田浩之, 中渡瀬秀一, 片岡良治. 特徴的な固有表現を用いたラベル指向ナビゲーション手法の提案. 情報処理学会論文誌データベース（TOD）, Vol. 46, No. SIG13(TOD27), pp. 40–52, sep 2005.

[2] Paul Thompson and Christopher C Dozier. Name searching and information retrieval. In *Second Conference on Empirical Methods in Natural Language Processing*, 1997.

[3] Y. Goldberg. 自然言語処理のための深層学習. 共立出版, 2019.

[4] D. Maynard, V. Tablan, C. Ursu, and Y. Wilks. Named entity recognition from diverse text types. 2001.

[5] J-D Kim, T Ohta, Y Tateisi, and J Tsujii. GENIA corpus–semantically annotated corpus for bio-textmining. *Bioinformatics*, Vol. 19 Suppl 1, pp. i180–2, 2003.

[6] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[7] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *CoRR*, Vol. abs/1603.01360, , 2016.

[8] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, Vol. 9, pp. 1735–1780, 1997.

[9] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. *Dive into Deep Learning*. 2020.

[10] 森信介, 中田陽介, Neubig Graham, 河原達也. 点予測による形態素解析. 自然言語処理, Vol. 18, No. 4, pp. 367–381, 2011.

[11] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[12] Ido Dagan and Sean P. Engelson. Committee-based sampling for training probabilistic classifiers. In Armand Prieditis and Stuart Russell, editors, *Machine Learning Proceedings 1995*, pp. 150 – 157. Morgan Kaufmann, San Francisco (CA), 1995.

[13] Hwanjo Yu. Svm selective sampling for ranking with application to data retrieval. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, p. 354–363, New York, NY, USA, 2005. Association for Computing Machinery.

[14] David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Machine Learning Proceedings 1994*, pp. 148–156. Elsevier, 1994.

[15] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pp. 287–294, New York, NY, USA, 1992. ACM.

[16] Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pp. 1070–1079, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[17] Yanyao Shen, Hyokun Yun, Zachary C. Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. *CoRR*, Vol. abs/1707.05928, , 2017.

[18] Graham Neubig. 点推定と能動学習を用いた自動単語分割器の分野適応. 言語処理学会年次大会, 2010, 2010.

[19] Xuezhe Ma and Eduard H. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *CoRR*, Vol. abs/1603.01354, , 2016.

[20] Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden markov models for information extraction. In *International Symposium on Intelligent Data Analysis*, pp. 309–318. Springer, 2001.

[21] Jason A. Fries, Sen Wu, Alexander Ratner, and Christopher Ré. Swellshark: A generative model for biomedical named entity recognition without labeled data. *CoRR*, Vol. abs/1704.06360, , 2017.

[22] Athanasios Giannakopoulos, Claudiu Musat, Andreea Hossmann, and Michael Baeriswyl. Unsupervised aspect term extraction with b-LSTM & CRF using automatically labelled datasets. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 180–188, 2017.

[23] Minlong Peng, Xiaoyu Xing, Qi Zhang, Jinlan Fu, and Xuanjing Huang. Distantly supervised named entity recognition using positive-unlabeled learning. In *Proc. ACL*, pp. 2409–2419. Association for Computational Linguistics, 2019.

[24] Zhanming Jie, Pengjun Xie, Wei Lu, Ruixue Ding, and Linlin Li. Better modeling of incomplete annotations for named entity recognition. In *Proc. NAACL*, p. 729–734, 2019.

[25] 辰巳守祐, 後藤啓介, 進藤裕之, 松本裕治. 辞書を用いたコーパス拡張による化学ドメインの distantly supervised 固有表現認識. Technical Report 7, 奈良先端科学技術大学院大学, 2019.

[26] Jingbo Shang, Liyuan Liu, Xiang Ren, Xiaotao Gu, Teng Ren, and Jiawei Han. Learning named entity tagger using domain-specific dictionary. In *Proc. EMNLP*, p. 2054–2064, 2018.

[27] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *Proc. NAACL*, pp. 260–270, 2016.

[28] 森嶋厚行. クラウドソーシングが不可能を可能にする―小さな力を集めて大きな力に変える科学と方法―. 2020.

[29] A P Dawid and A M Skene. Maximum likelihood estimation of observer Error-Rates using the EM algorithm. *J. R. Stat. Soc. Ser. C Appl. Stat.*, Vol. 28, No. 1, pp. 20–28, 1979.

[30] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, Vol. 22, pp. 2035–2043. Curran Associates, Inc., 2009.

[31] An Thanh Nguyen, Byron Wallace, Junyi Jessy Li, Ani Nenkova, and Matthew Lease. Aggregating and predicting sequence labels from crowd annotations. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 299–309, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[32] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-Aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*, March 2016.

[33] Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. Sequence labeling with multiple annotators. *Mach. Learn.*, Vol. 95, No. 2, pp. 165–181, May 2014.

[34] Chu-Cheng Lin, Waleed Ammar, Chris Dyer, and Lori Levin. Unsupervised POS induction with word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1311–1316, Denver, Colorado, 2015. Association for Computational Linguistics.

[35] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[36] 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.