NAIST-IS-MT1651056

修士論文

非即時的なタスク設定における固有表現抽出の改善

澤山 熱気

2018 年 3 月 9 日

奈良先端科学技術大学院大学
情報科学研究科

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士 (工学) 授与の要件として提出した修士論文である。

澤山 熱気

審査委員：

松本 裕治 教授　（主指導教員）
中村 哲 教授　（副指導教員）
新保 仁 准教授　（副指導教員）
進藤 裕之 助教　（副指導教員）
能地 宏 助教　（副指導教員）

非即時的なタスク設定における固有表現抽出の改善*

澤山 熱気

内容梗概

固有表現抽出は科学技術論文からの知識の自動抽出タスクにおいて中心的な役割を担っている．一般的に，自動抽出システムを構築する際には，固有表現抽出モジュールをシステム内部で独立に構築する．システムを利用する多くの場合において，リアルタイムな利用設定を仮定しているため，固有表現抽出モジュールはできるだけ早く入力の結果を返すことが必要となる．一方で，科学技術論文からの知識の自動抽出タスクにおいては，長い時間間隔の間に論文から徐々に知識を抽出することがタスクの目的であるため，多くの場合において即時的な結果の出力は不要である．このような背景から，本論文では非即時的なタスク設定における固有表現抽出について議論をする．本論文では，評価データを活用する方法に特に注目し，抽出精度向上を目指す．バイオインフォマティクス分野のデータセットを用いた実験により，提案手法が有用であることを示す．

キーワード

固有表現　固有表現抽出　知識抽出　科学技術論文，ニューラルネットワーク，部分単語

*奈良先端科学技術大学院大学 情報科学研究科 修士論文, NAIST-IS-MT1651056, 2018 年 3 月 9 日.

Improving Named Entity Recognition in Tasks with Non-immediate Response Setting*

Atsuki Sawayama

Abstract

Named entity recognition (NER) plays a central role in the task of automatic knowledge extraction from scientific and technical papers. Generally, we develop a NER module independently from the over-all system. At that time, we assume that the system works in a “real-time” setting, and thus, the NER module needs to respond inputs immediately as possible. However, in the actual scenario of the automatic knowledge extraction from scientific and technical papers, the immediate response is unnecessary in most cases since the purpose of the tasks to (gradually) accumulate knowledge from the papers over a relatively long time interval. From this background, we discuss approaches suitable for tackling the non-immediate response setting of NER. Moreover, this paper specifically focuses on an approach that utilizes a given test data for further improving the task performance. We conducted NER experiments in Bioinformatics field and showed that our proposed method improves the NER accuracy.

Keywords:

Named Entity, Named Entity Recognition, Scientific paper, Knowledge extraction, Neural Network, Subword

*Master's Thesis, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT1651056, March 9, 2018.

目 次

目 次

図 目 次

表 目 次

1. はじめに

インターネットの普及以降，膨大な数の科学技術論文を誰でも手軽に獲得できるようになった．また，最新の研究成果が日々投稿され公開されている．例えば，生命科学やバイオメディカル分野の論文検索エンジンで知られる Pubmed [1]のデータベースである MEDLINE は，公開されている情報に基づくと，おおよそ 40 万もの論文が毎年登録されている．これは 1 日あたり平均して 1000 本を超える計算になる．このような状況から，たとえその分野の専門家（或いは研究者）であっても，これらの大量の論文すべての内容に目を通し，理解することは非常に困難な状況にある．つまり，専門家に必須となる，分野の動向や最先端の研究成果を把握し続ける行為に多大な労力が必要となる．

この問題を緩和する方法として，自動知識抽出システムの利用が考えられる．実際に，これまでもいくつかの自動知識抽出システムが提案されてきた [1, 2] ．こういったシステムを利用することで，大量の論文から自分が必要とする情報を比較的容易に獲得することができるようになり，専門家の情報収集の労力が大幅に軽減されることが期待できる．

このような動機から，本論文では科学技術論文からの知識抽出に関して，更なる抽出精度の向上に向けた方法論の議論を行う．ただし，自動知識抽出システムは，固有表現抽出（Named Entity Recognition, NER）モジュール，関係抽出 (Relation Extraction, RE) モジュール，辞書構築モジュールといった多くのモジュールで構成される．その全てを一度に議論するのは困難であるため，本論文では，最初の取り組みとして固有表現抽出モジュールの抽出精度向上に焦点をあてる．これは，固有表現抽出モジュールは自動知識抽出システムの最初の処理にあたり，その性能が後続のモジュールの性能に大きく影響を与えると考えられるためである．

[1]https://www.ncbi.nlm.nih.gov/pubmed

1.1 研究の目的と意義

本論文では，科学技術論文からの知識抽出に向けた固有表現抽出の特性を考え，その特性に即して科学技術論文からの知識獲得に向けた固有表現抽出法を提案する．本論文では,「非即時的なタスク設定」と「頻出する規則性のある新しい用語」の二点に着目する．これらの特性に即した固有表現抽出を構築することで，従来の手法よりも高精度な固有表現の抽出の実現を目指す. 提案手法の詳細は第 4 章，第 5 章で示す.

1.2 本論文の構成

本論文の構成は以下の通りである. 第 2 章では，固有表現抽出の関連研究について概説する. 第 3 章では，非即時的なタスク設定における固有表現抽出について議論し，評価データを活用する手法である自己学習の予備実験を行う．第 4 章では，提案手法について概説し，提案手法を用いた実験と考察について記述する．第 5 章では，第 4 章で得た知見から，部分単語を用いた提案手法の実験と考察にについて記述する．第 6 章では，本研究で得た結論と将来の展望を記述する.

2. 関連研究

固有表現抽出（Named Entity Recognition, NER）とは，文中の固有表現（Named Entity, NE）と事前に定義した重要語をを抽出する技術 (図 1) であり，自動知識抽出システムをはじめ，対話システム，質問応答システムなどを構成する中で，最も基本となる言語処理技術である．一般的に，固有表現抽出技術では，文中の各単語に対し，事前に定義した固有表現クラスの中で最も適切であるクラスとその範囲ラベルを付与する，系列ラベリング問題として解かれ，ラベル付与方式として，BIO 方式 (Begin, Inside, Outside)，または BIOES 方式 (Begin, Inside, Outside, End, Single) [3] が一般的に用いられている (図 2)．系列ラベリングの手法としては，隠れマルコフモデル（HIdden Markov Model, HMM）[4]，条件付き確率場（Conditional Random Fields, CRF）[5]，サポートベクタマシン（Support Vector Machine, SVM）[6]，ニューラルネットワーク（Neural Network）[7, 8, 9] を用いた手法が提案されている．加えて，近年では，リカレントニューラルネットワーク（Reccurent Neural Network, RNN）をベースとし，条件付き確率場を組み合わせた用いた手法 [10, 11, 12] が，既存の CoNLL-2003 shered task datasets [2]などの標準的な固有表現抽出のベンチマークデータセットにおいて，高い精度を示している．標準的な固有表現抽出では，MUC[13] で定義された，情報としての単位がはっきりしている人名，組織名，地名，時間，日時，金額表現，割合表現の七種類の表現を抽出することを想定している．このことから分かるように，新聞などのフォーマルな文章から固有表現の抽出を行うことを想定し，抽出精度の向上を目指してきた．一方，近年では大量のテキストデータを処理することができるようになり，さまざまな分野においても固有表現抽出の抽出精度を向上させることで検索システムやシステム性能を改善させようとする動きが高まっていたり [14, 15, 16, 17]，SNS などのくだけた文からの固有表現抽出を目的としたコンペティションなども開催されている [18].

[2]https://www.clips.uantwerpen.be/conll2003/ner/

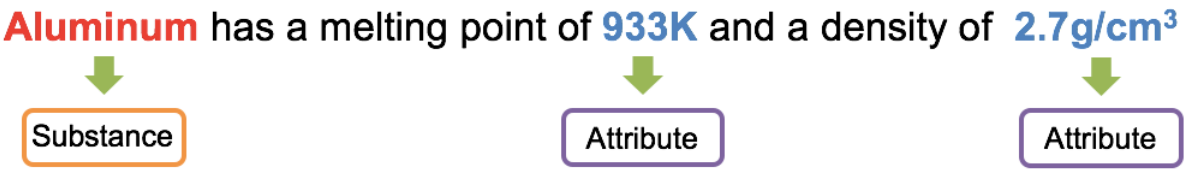

図 1　固有表現抽出の例

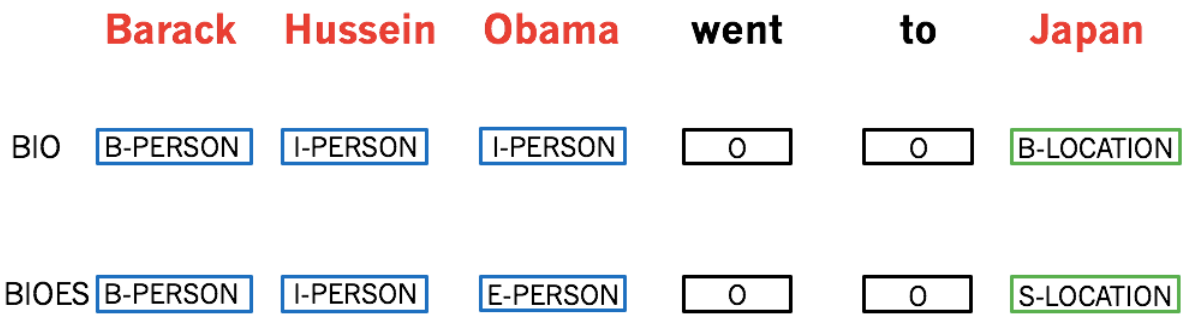

図 2　系列ラベリングによる固有表現抽出の例

3. 科学技術論文からの知識抽出設定における固有表現抽出

科学技術論文からの知識抽出における固有表現抽出では，文中の事前に定義された専門用語クラスの固有表現の抽出をおこなう．専門用語クラスの例として，バイオドメインではタンパク質，物質ドメインでは構成材質に加え，温度や融点などの属性値が該当する．前章で述べたように，近年では，リカレントニューラルネットワーク（Reccurent Neural Network, RNN）をベースとした条件付き確率場 (Conditional Random Field, CRF)[10, 11, 12] が標準的なベンチマークデータセットにおいて高い抽出精度を示しているため，本論文でも，RNN による条件付き確率場をベースライン手法と想定して議論を行う．

以下，まず科学技術論文（以降「論文」と略記）からの知識抽出に向けた固有表現抽出の特性を考え，その特性に即して論文からの知識獲得に向けた固有表現抽出法を提案する．ここでは，特に以降に述べる「非即時的なタスク設定」に着目する．

3.1 非即時的なタスク設定

論文からの知識抽出における固有表現抽出では，論文の収集・知識抽出から利用者がシステムを利用するまでの間に時間的な猶予がある．そのため，利用されるまでの時間的な猶予を活用し，獲得した論文全てを用いて繰り返し学習を行うことが可能である．つまり，学習の際に評価対象のデータを既に得ている前提で，評価データを学習に用いることができる．これは，従来，固有表現抽出タスクの利用シーンの想定となっている対話・QA(Question answering) システム等のリアルタイム性を求められる状況とは異なる．本論文では，対話・QA システム等のリアルタイム性を求めるタスク設定を「即時的なタスク設定」，論文からの知識抽出タスク設定を「非即時的なタスク設定」と呼ぶ．

即時的なタスク設定では，ユーザーがどのようなクエリを入力するかが不明であるため，システムは事前に評価すべきデータが不明である．一方，非即時的なタスク設定では，事前に抽出対象が定義されているので，抽出対象となるデー

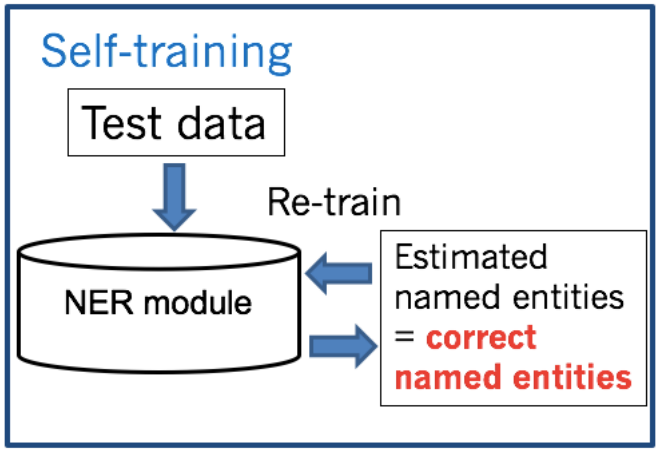

図 3　自己学習の概要

タが事前に得られる．抽出対象となるデータを学習時に既に得ているのであれば，学習用のデータのみではなく，評価データから得られる情報を有効に活用して固有表現抽出モデルを学習する方法論を選択することができる．例えば，機械学習の研究分野では，評価データを活用する学習法はトランスダクティブ学習（Transductive learning）と呼ばれ，多くの研究が行われている [19]．トランスダクティブ学習では，評価データを学習データの一部として活用する．固有表現抽出におけるトランスダクティブ学習では，学習した固有表現抽出モデルが評価データに対してつけた予想ラベルを正しいラベルとし，モデルの再学習に用いる方法が一般的である (図 3)．これは，評価データをラベルなしテキストとして用いた自己学習（Self-training）[20] としてみなすことができる．

3.2 予備実験：自己学習

トランスダクティブ学習を用いた手法の予備実験として，単純な自己学習を用いた固有表現の抽出を行った．本論文では，バイオインフォマティクス分野の論文からの知識抽出を想定する．ここでは，論文中からタンパク質の固有表現を抽出する実験として，ベースとなる固有表現抽出モデル (図 4) と評価データを用いた自己学習の比較実験を行った．実験に用いるデータセットとして，BioNLP 2011 shared task の Entity relation supporting task データセット [21][3]を用いた．データセットは論文アブストラクトで構成されており，学習データ (800 本)，開発データ (160 本)，評価データ (250 本) から成り立つ．

実験のベースとなる固有表現抽出モデルとして，NeuroNER [22] を用いた．このモデルのネットワークは Bi-directional LSTM-CRF で構築されている．実験には，NeuroNER の標準パラメータを用い，学習エポック数のみ 40 に設定した (表 1)．NeuroNER の標準の単語ベクトル (図 4 中の Word embedding) には，学習済みの単語埋め込みベクトルが用いられ，文字ベクトルは，Bi-RNN を用いて構築される (図 4 中の Character embedding)．その後，単語ベクトルと文字ベクトルを連結し，モデルの入力として与えられる (図 4 中の Token embedding)．実験では，それぞれの単語の出力に BIOES [23] タグを適用するため，タグの種類は 5 種類となる．したがって，5 個のタグからどのタグが該当するかを選択する系列ラベリング問題として固有表現抽出モデルを構築した．実験の性能評価には，フレーズごとの F 値 [24] を用いた．

最初に，学習データを用いてベースモデルを構築し，評価データに対して学習したモデルでタグ付けを行う．その後，自己学習の設定における教師なし学習データとして，タグ付けをした評価データを用いて，ベースモデルの再学習を行った．

[3]http://2011.bionlp-st.org/home/entity-relations

表 1 実験設定

パラメータ	値
文字埋め込みベクトルの次元数	50(25 × 2)
文字 LSTM の隠れ層の次元数	25
ドロップアウト率	0.5
学習率	0.005
学習済み単語埋め込みベクトル	Glove.6B.100d
学習済み単語埋め込みベクトルの次元数	100
勾配法	sgd
早期終了エポック数	10 エポック
単語ベクトルの次元数	150
単語 LSTM の隠れ層の次元数	100
バッチサイズ	1
学習エポック数	40

3.3 結果と考察

実験の結果を表 2, 図 5, 図 6 に示す．縦軸が F 値で，横軸が学習エポック数を示している．表 2 中の学習 (train)・開発 (valid) データの F 値は学習中の最大値であり，評価データの F 値は，学習中の最大値 (test max) と，開発データが最も高い値となったエポックのモデルを用いて評価した際の値 (test best) である．自己学習では，ベースモデルと比べ，抽出精度の向上が見られた.

予備実験として行った単純な自己学習などのトランスダクティブ学習では，モデルに与えられる評価データの正解は既知ではないことが一般的である．そのため，単純な自己学習では，固有表現抽出モデルが予測した固有表現タグの結果を，正誤に関わらず，正解であるものとして再学習に活用する問題が知られている．表 2 の自己学習後の精度からも分かるように，再学習によって新しい情報がほとんど得られていないことが容易に予想される．加えて，単純な自己学習では，一度間違えてしまった固有表現タグをモデルが間違い続ける問題が発生する．このことから，再学習の際に，予測した固有表現タグ結果をそのままモデルに与える

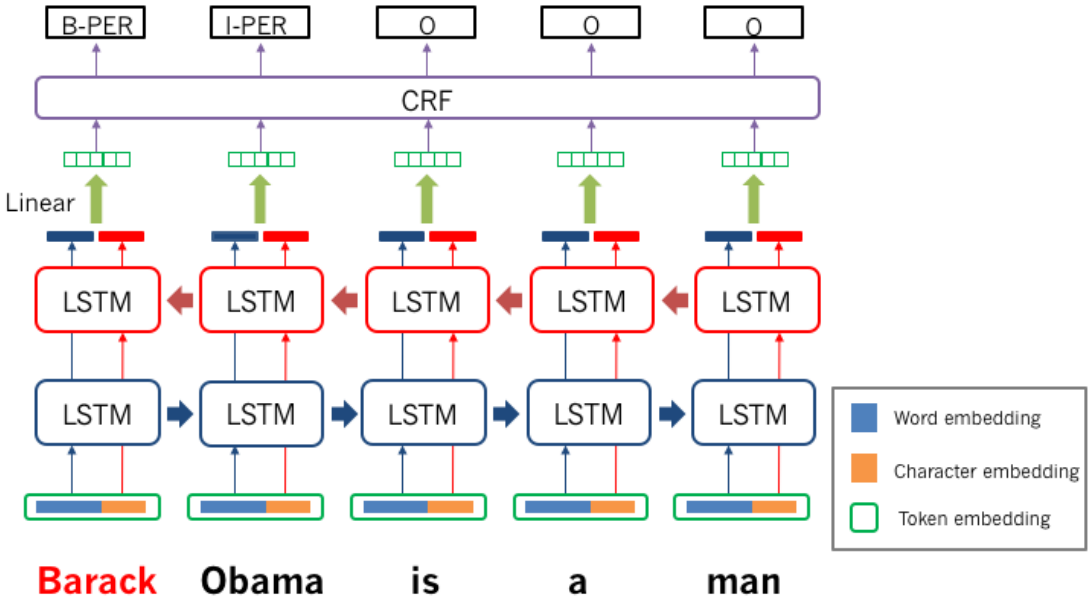

図 4　ベースモデル (Ib-model) のモデル構造

のではなく，何らかの別の情報としてモデルに与える方法論が求められる．

表 2　ベースモデルと自己学習の F 値の比較

ベースモデル (F 値)	自己学習 (F 値)
82.25	83.00

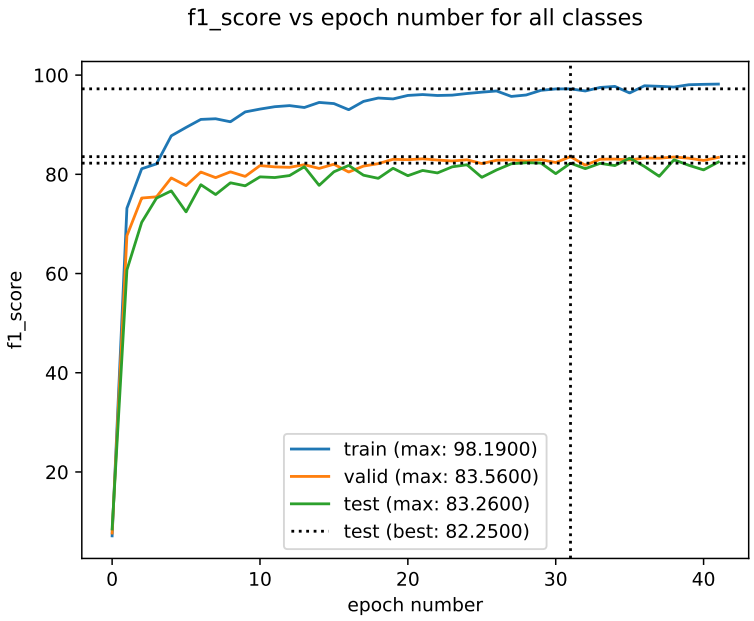

図 5　ベースモデル

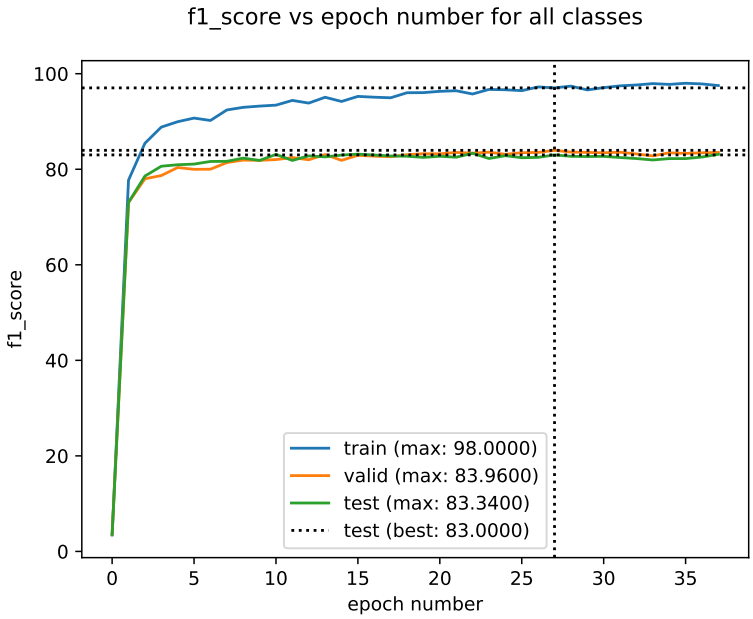
f1_score vs epoch number for all classes
f1_score
epoch number
train (max: 98.0000)
valid (max: 83.9600)
test (max: 83.3400)
test (best: 83.0000)
0
20
40
60
80
100
0
5
10
15
20
25
30
35

図 6　自己学習

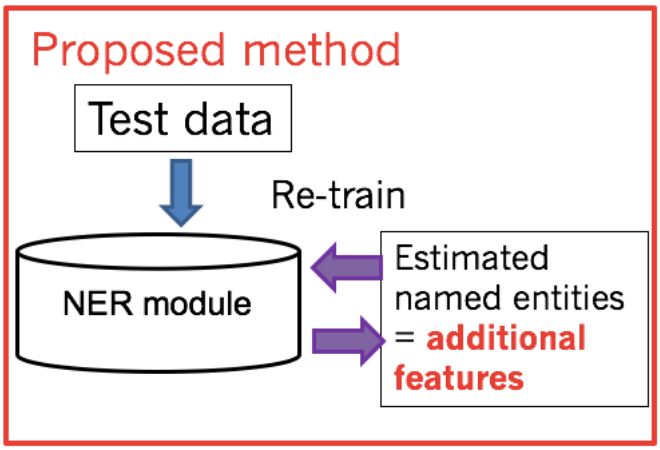

図 7　提案手法の概要

4. 提案手法

4.1 評価データの予測分布の活用

提案手法では，第 3 章で議論した特性を効果的に利用するため，評価データが予測した固有表現タグを正解としてモデルに与える代わりに，評価データの予測分布を追加の特徴として活用し，モデルに与える方法を考える (図 7)．提案手法の基本となるアイディアは，入力データに対するタスク依存の類似度を計算し，追加の特徴として用いる方法となる．基本的な学習の枠組みは一般的な自己学習と似ているが，自己学習と提案手法との大きな違いは，モデルが予測した固有表現をどのように再学習に用いるかである．

以下，学習の枠組みを説明するために，まず，提案手法の概要図を図 8 に示す．提案手法では，最初に初期ベースモデル（Initial base model, Ib-model と呼ぶ）を一般的な固有表現抽出と同じ設定で構築する（図中 1）．次に，Ib-model を用いて，与えられた評価・開発データに対してタグに対する予測分布を計算する（図中 2）．その後，予測分布を追加情報とし評価・開発データを用いて，類似度素性モデル（Similerity feature model, Sf-model と呼ぶ）を学習する（図中 3）．次に，Sf-model を用いて，学習データの類似度素性 (similerity feature) を算出する

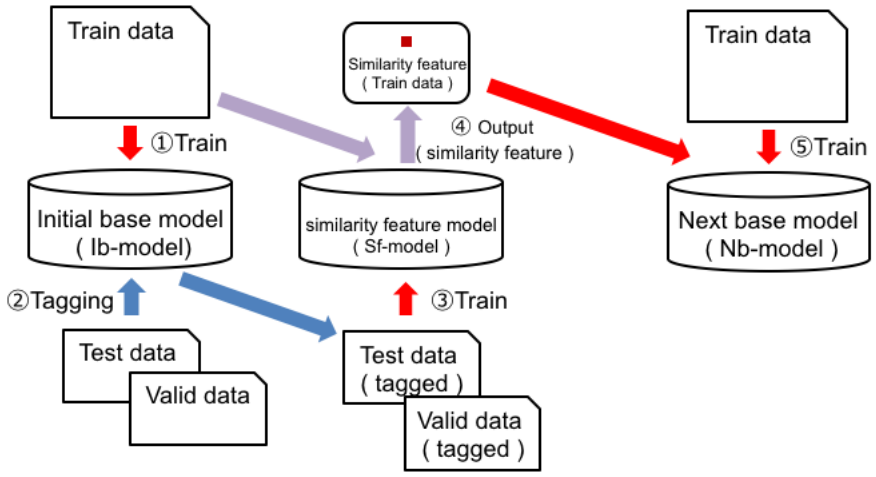

図 8　提案手法の概略図

（図中 4）．最後に，類似度素性を活用して Ib-model を再学習した次期ベースモデル（Next base model, Nb-model と呼ぶ）を構築する（図中 5）．

Ib-model は，一般的な固有表現抽出モデルと同様のネットワーク構造でモデル構築し，評価･開発データに対して固有表現タグの予測分布を計算する役割を担っている．

Sf-model は，データ間の単語それぞれの類似度を計算する役割を担っている．そのため，Sf-model は，固有表現抽出モデルのネットワーク構造で構築するが，モデルの役割は固有表現抽出ではない．モデル学習には，Ib-model によってタグづけされた評価・開発データを用いる．Sf-model では，評価・開発データの観点で，Ib-model の予測分布と同じ予測分布となるように学習し，学習データの類似度素性を算出する．ここでの類似度とは，評価・開発データの単語が，Ib-model の観点で固有表現と似ているかどうかを予測分布で出力したものとなる．そのため，学習データの類似度素性の算出時，学習データの固有表現が評価・開発デー

タ中の固有表現と似ているならば，似た予測分布を出力することを期待している．次に，Sf-model における，類似度素性の算出方法について説明する．一般的な固有表現抽出モデルでは，各固有表現クラスの確率値を出力し，その値のうちの最も高い確率のタグを予測タグとして用いている．提案手法の類似度素性として，各固有表現タグの確率値である BIOES [3] の 5 次元の値を用いた．注意すべき点として，Sf-model では点ごとの推定結果を算出するために，CRF を用いず，RNN でのモデル構築を行う．

Nb-model では，Ib-model に Sf-model から算出した類似度素性を活用し，再学習を行う．実験では，類似度素性の活用方法として，二つの手法の実験を行った．

一つ目は，RNN モデルの出力に類似度素性を足しこむ手法 (Nb-model 手法 1 と呼ぶ) である．手法 1 では，RNN の線形変換後の値に類似度素性を足しこむことで，類似度素性にバイアスのような役割を持たせ，モデルの学習に役立てる．手法 1 のモデルの構造を図 10 に示す．次に，計算式を式 1 に示す．単語ベクトル $\boldsymbol{X}$ を Nb-model の入力として与え，RNN の線形変換後の出力値である BIOES の 5 次元の値 $\Phi(\boldsymbol{X})$ に対して，同様に Sf-model から算出した類似度素性 $\Psi(\boldsymbol{X})$ を足し合わせる．その値を $\boldsymbol{O}$ とし，CRF への入力とする (式 1)．パラメータ α は類似度素性の用いる度合い，ϵ は log スケールの補正のために導入している．

$$\boldsymbol{O} = \Phi(\boldsymbol{X}) + \alpha \log(\Psi(\boldsymbol{X}) + \epsilon) \tag{1}$$

ただし，α は [0,1.0] から設定し，$\epsilon = 1.0 \times 10^{-6}$ とする．

二つ目は，単語ベクトルに類似度素性を連結する手法 (Nb-model 手法 2 と呼ぶ) である．手法 2 のモデルの構造を図 11 に示す．手法 2 では，モデルに入力する単語ベクトルに類似度素性を入力し，単語の類似度素性を学習に活用する．

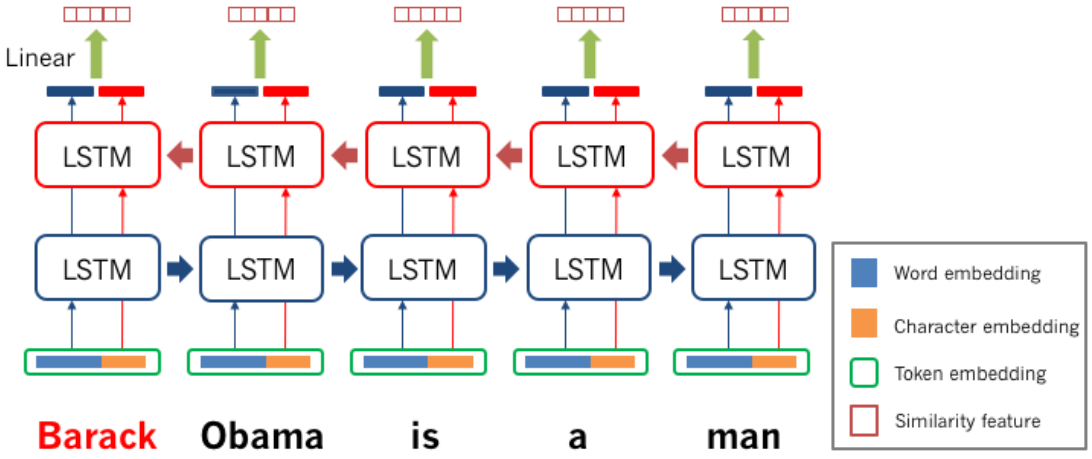

図 9　Sf-model のモデル構造

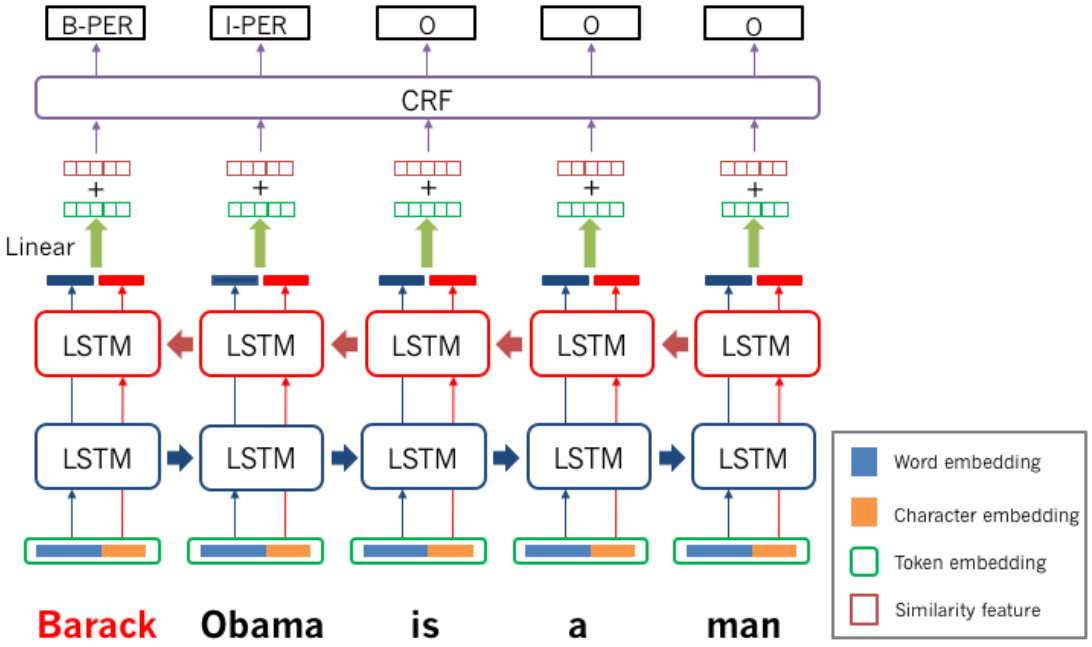

図 10　Nb-model(手法 1) のモデル構造

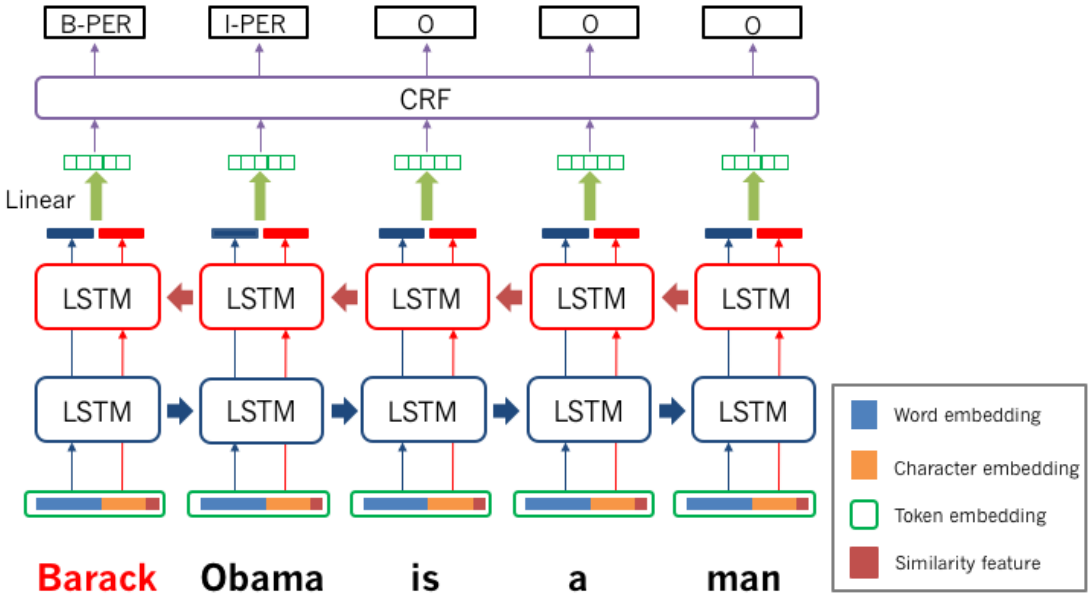

図 11　Nb-model(手法 2) のモデル構造

4.2 実験

バイオインフォマティクスの知識抽出を想定し，Ib-model と Nb-model の比較実験を行った．実験に用いるデータセット，固有表現抽出モデルは第 3 章と同様のもの [21, 22] を用いた．実験でのパラメータを図 3 に示す．新たに追加したパラメータ α は [0,1.0] の値を手動で決定して用いた．各モデルの構造は図 4, 図 9, 図 10, 図 11 で示したものを用いる．Sf-model では，Ib-model でタグをつけた評価データと開発データを用いてモデルの構築を行った．注意すべき点として，Sf-model では点ごとの推定結果を算出するために Bi-LSTM のみでモデルの構築を行っている．Nb-model の手法 1(図 10) では，学習済みの Ib-model に対して類似度素性を追加して再学習を行った．Nb-model の手法 2(図 11) では，Ib-model と Nb-model で入力の単語埋め込みベクトルの次元数が異なるため，類似度素性を単語埋め込みベクトルに連結したのち，新たに学習を行った．

4.3 結果と考察

実験結果を図 12，図 13，図 14 に示す．実験の結果，Nb-model の手法 1(図 10) において，α の値が高い設定の際に学習した Ib-model に類似度素性を足しこむ (表中の 0 エポック) と，Ib-model よりも F 値が高くなった (図 13，表 4)．Ib-model の結果と比較し，recall の向上に役立ったことが分かった (表 5, 表 6)．しかし，そこから再学習をしていくと，精度の大幅な上昇が見られなかった．このことから，類似度素性を足しこむことでモデルの改善することができるが，学習データのみで再学習するため，再学習時の精度向上に貢献できなかったのではないかと考えられる．一方で，手法 2(図 11) では，精度の向上がみられなかったものの，Ib-model と比べ，学習速度の改善には効果があった (表 14)．これは，学習の際に単語ベクトルに連結した類似度素性の情報が，単語ベクトルの情報よりも重視されなかったために，学習時にうまく活用されなかったのではないかと考えられる．

今回の実験では，モデル構築，データセットそれぞれにおいて，改善すべき点がある．モデル構築の改善においては，以下の三つの点が挙げられる．一つ目は，Sf-model の構築である．今回の Sf-model の構築には評価データと開発データを

表 3 実験設定

パラメータ	値
文字埋め込みベクトルの次元数	50(25 × 2)
文字 LSTM の隠れ層の次元数	25
ドロップアウト率	0.5
学習率	0.005
学習済み単語埋め込みベクトル	Glove.6B.100d
学習済み単語埋め込みベクトルの次元数	100
勾配法	sgd
単語ベクトルの次元数（Ib-model）	150
単語ベクトルの次元数（Sf-model）	150
単語の類似度素性の次元数	5
単語ベクトルの次元数（Nb-model 手法 1）	150
単語ベクトルの次元数（Nb-model 手法 2）	155
単語 LSTM 隠れ層の次元数	100
バッチサイズ	1
学習エポック数	100
α	[0,1.0]

用いたが，より正しい類似度を計算するために，少量の評価データのみではなく，大量の論文を用いてモデルの学習をすることが必要であると考えられる．二つ目は，専用の単語埋め込みベクトルを用意することである．今回用いた単語埋め込みベクトルは新聞などの一般分野で学習されたものを使用しているため，バイオインフォマティクス専用の単語埋め込みベクトルを用意することで，精度の改善に繋がると考えられる．三つ目は，パラメータ α の自動学習を行うことである．今回の実験では，パラメータ α の値の範囲を手動で決定して適用しているが，今後は自動で学習できるようにすることで，適切な α の値を取れるようにする必要があると考えられる．

また，使用したデータセットを調査したところ，学習データと評価データでは，

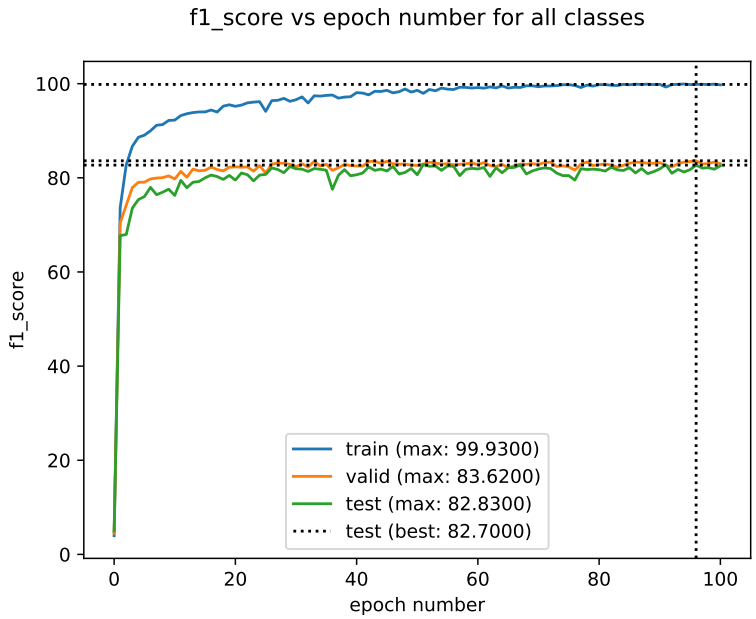

図 12　Ib-model の実験結果

語彙の異なりが大きいことが分かった (表 7)．すなわち，使用したデータセットには多くの未知語が存在する問題がある．一方で，データセット中の固有表現には規則性のある文字列が頻出し，同じ部分文字列を持つ固有表現が存在する．このことから，出現する固有表現の規則的な部分文字列を情報として活用することに加え，共通する文字列単位で単語分割をおこない，未知語や語彙の異なりを低減させることができれば，精度の向上に繋がる可能性があると考えられる．

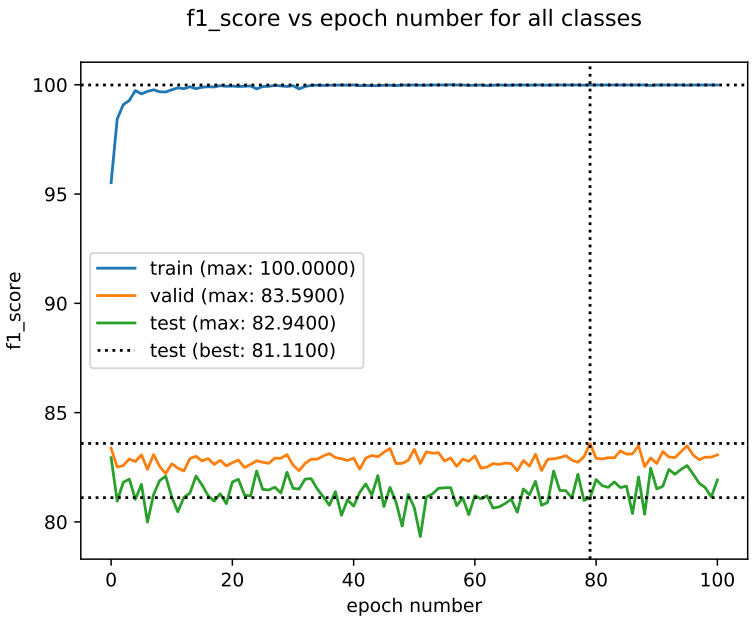

図 13　Nb-model(手法 1) の実験結果（α=1.0）

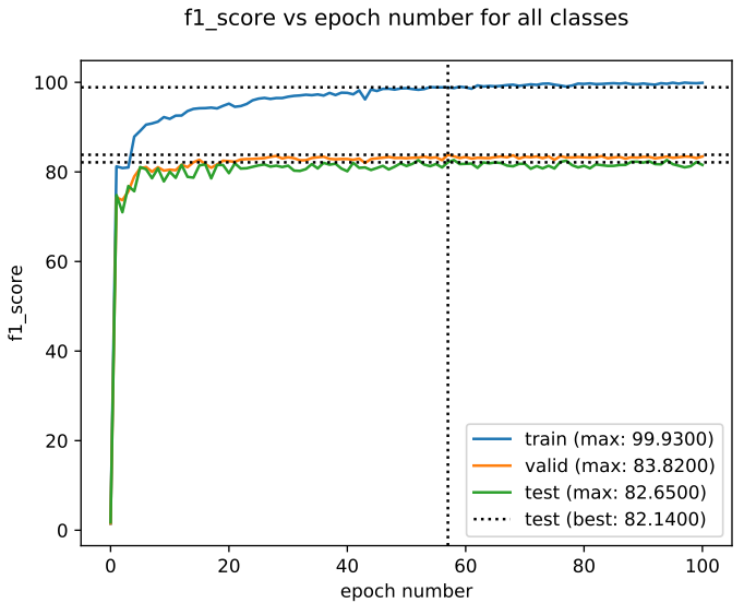

図 14　Nb-model(手法 2) の実験結果

表 4 α の変化による Nb-model(手法 1) の F 値の変化

α	F 値	F 値 (0 エポック)
0.0	82.33	82.70
0.001	82.03	82.61
0.005	81.80	82.63
0.01	81.94	82.61
0.05	81.91	82.63
0.1	82.64	82.64
0.2	82.63	82.63
0.5	**82.92**	**82.92**
1.0	81.11	**82.94**

表 5 Ib-model の分類結果

固有表現クラス	Precision	Recall	F 値
B-Protein	84.88	74.66	79.44
E-Protein	84.20	73.98	78.76
I-Protein	86.28	77.37	81.58
S-Protein	87.29	82.80	84.98
micro-avg	86.08	78.63	82.16

表 6 Nb-model(手法 1) の分類結果 (α=1.0)

固有表現クラス	Precision	Recall	F 値
B-Protein	84.15	75.24	79.45
E-Protein	84.29	75.53	79.77
I-Protein	85.67	77.41	81.33
S-Protein	87.12	83.54	85.29
micro-avg	85.77	79.36	82.42

5. 部分単語を用いた精度の改善

5.1 頻出する規則性のある新しい用語

第 4 章をふまえ，本章では，論文からの知識抽出における特性である「頻出する規則性のある用語」に着目する．論文からの知識抽出を想定した固有表現抽出では，比較的規則性のある専門用語が固有表現として多く出現する．例えば，バイオ分野や化学分野におけるタンパク質名や化学式などが挙げられる．これらの専門用語では，物質構造や化学変化，構成材質に沿って用語の命名がされるため，文字列に規則性が存在する．また，こういった専門用語は日々新しく生み出されている状況にある．こういった新しく生み出された用語は機械学習の文脈では未知語となる．一般的に，未知語は抽出精度の悪化に大きく影響を与えると考えられるため，これらの未知語をうまく取り扱う方法論が求められる．

近年，盛んに研究が進んでいるニューラルネットワークを用いた言語処理においては，未知語を扱う方法の一つとして文字列情報を組み込む方法がしばしば用いられている．本章では，前述したように論文中には規則性のある専門用語が固有表現として出現しやすいという特性に着目し，その分野の用語として高頻度で出現する特有の部分文字列に分解して利用する方法も合わせて活用することとする．

5.2 部分単語への分割

前節で述べた頻出する規則性のある新しい用語に適した方法として，固有表現抽出の前処理として，与えられた文章の各単語に対して部分単語（Subword）による分割を行う．このとき，部分単語への分割方法にはいくつかの方法論が考えられる．例えば，もっとも単純には人手作成のルールによる方法が考えられる．しかし，新しい用語の増加によるルールの網羅性の低下，少数の例外処理，ルール追加のメンテナンスコストなどが必要となり，あまり良い方法とは言えない．

近年，統計量に基づく部分単語への分割方法がニューラル機械翻訳の研究分野でよく用いられるようになってきた [25]．本章では，統計的な方法を用いて部分

単語への分割を行う．

例として，タンパク質の固有表現の分割処理を図 15，図 16 に示す．図 15 に示すように，学習データには GM-CSF というタンパク質の固有表現が存在し，評価データでは G-CSF というタンパク質の固有表現が存在する状況を仮定する．図中の青枠が一度に固有表現抽出モデルへ入力する，1 処理単位とする．

単語単位（図中 Word Level）では，学習データに存在しない G-CSF はそのまま 1 処理単位として処理されるため，評価データを予測する際，G-CSF は未知語扱いとなる．一方，文字単位（図中 Character Level）にあるように，1 文字ずつ分割して処理単位とすると，学習時に，GM-CSF を文字単位で学習するため，G-CSF を構成する全ての文字が既知である．そのため，評価データの予測時に G-CSF は未知語ではなくなる．しかし，単語を 1 文字ずつ分割することにより，モデルに入力する一文あたりの処理単位が増加する．そのため，モデルの一文あたりの予測回数が増え，予測精度が下がる可能性がある．部分単語（図中 Subword Level）は，単語単位と文字単位の中間にあたる．部分単語による分割をおこなった場合では，評価データの G-CSF が未知語とならないことに加え，学習データの GM-CSF と評価データの G-CSF に共通する規則性のある文字列 CSF を捉えることができる．そのため，G-CSF の抽出が向上できることが期待できる．加えて，文字単位よりも処理単位を比較的少なくすることができるため，モデルの予測回数増加による抽出精度の低下を抑えられる可能性がある．

一点注意として，図 16 に示すように，固有表現抽出では，機械翻訳のデータとは違い，固有表現タグと単語間の対応関係を部分単語へ分割する際に保持する必要がある．つまり，単語を部分単語へ分割，及び，逆変換に相当する元の分割単位に復元する際に，タグの対応関係も同様に変換可能する必要がある．しかし，これは単純な規則で容易に変換可能であるため，部分単語への分割を固有表現抽出へ適用することは問題とはならない．

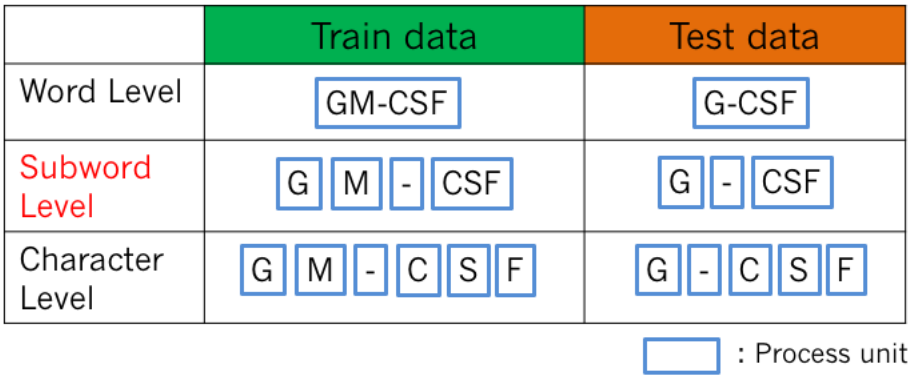

図 15　単語分割による分割単位の違い

5.3　実験

5.3.1　部分単語による前処理

予備調査として，第 4 章の実験に用いたデータセット [21] の語彙の比較を行った．表 7 に各データ間を比較した語彙の異なりを示す．この表からデータ間の語彙が大きく異なることが分かる．よって，部分単語を用いて評価データの未知語を特有の部分文字列へ分解して既知の単語として扱う必要があり，これによって，抽出精度の向上に繋がる可能性がある．

部分単語へ分割するために既存のライブラリである，Subword-NMT [25] を利用した．学習データの単語の語彙が，1000, 2000, 3000（Sub 1k，2k，3k と略記する）程度になるよう分割し，その分割方針を用いて，評価データ，開発データも同様に分割した．

データセットに対して部分単語を適用した結果を表 7，表 8 に示す．部分単語

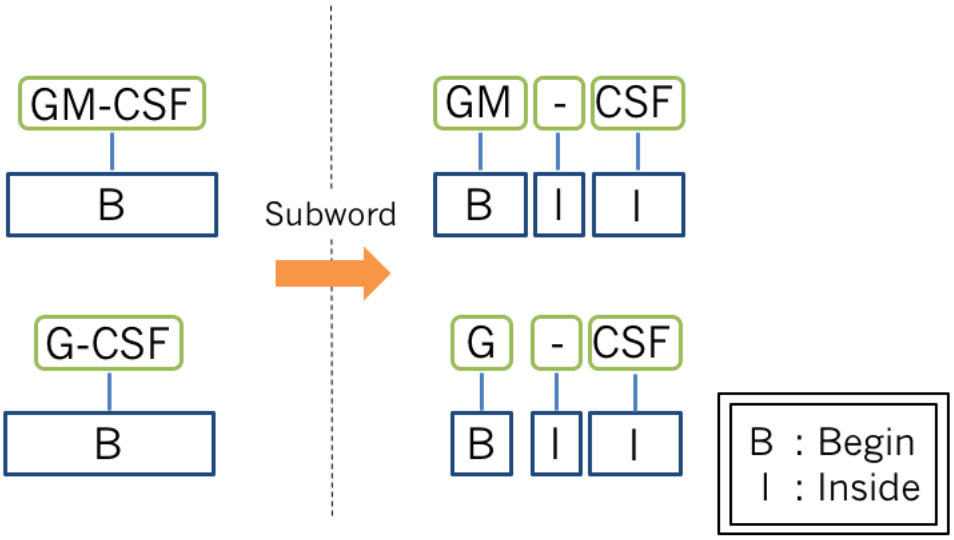

図 16 部分単語による単語の部分文字列への分割と固有表現タグの分割例

を用いたことで，各データセット間の語彙の異なりは低下し，固有表現の平均単語数は増加した．このことから，評価データに出現する未知語を低減できたことが分かる．

5.3.2 分割されたデータによる固有表現の抽出

固有表現抽出モデルとして前章と同様に NeuroNER [22] を用いた．モデルのネットワークは Bi-directional LSTM-CRF で構築されている．実験には，NeuroNER の標準パラメータを用い，標準で用いられている学習済みの単語埋め込みベクトルのみ使用しなかった．これは，バイオインフォマティクス分野のテキストで学習されていないベクトルであること，データセットに部分単語を適用する効果を明確にするためである．Ib-model では，通常の固有表現抽出モデルと同様に NeuroNER [22] を学習し，開発・評価データにタグをつけた．その後，タグをつけた開発・評価データを用いて Sf-model を Bi-directional LSTM でモデル学習し

表 7 部分単語化によるデータの語彙の比較

	標準	Sub 1k	Sub 2k	Sub 3k
学習データ	16918	1172	2189	3193
開発データ	6009	1130	2058	2878
評価データ	9432	1163	2136	3048
学習・評価の比較				
学習・評価の和集合	21176	1177	2196	3205
学習・評価の積集合	5174	1158	2129	3036
学習データのみ出現	11744	14	60	157
評価データのみ出現	4258	5	7	12
学習・開発の比較				
学習・開発の和集合	18931	1173	2191	3201
学習・開発の積集合	3996	1129	2056	2870
学習データのみ出現	12922	43	133	323
開発データのみ出現	2013	1	2	8

たのち，各データセットの類似度素性を算出した．Nb-model では，学習済みの Ib-model に対して類似度素性を追加して再学習を行った．パラメータ α は，[0,1.0] の値を手動で決定して用いた．実験では，それぞれの単語の出力に BIOES タグを適用するため，タグの種類は 5 種類となる．実験の性能評価には，フレーズごとの F 値 [24] を用い，Ib-model，Nb-model に加え，同一設定で Ib-model を追加学習 (100 エポック) したモデルと精度比較を行った.

5.4 結果と考察

部分単語に分割したデータセットを用いた実験結果を表 9 に示す．学習・開発データの F 値は，学習 (100 エポック) 中で最も高い値である．一方，評価データの F 値は，開発データが最も高い値となったエポックのモデルを用いて評価した際の結果である．学習済みの単語埋め込みベクトルを使用しない設定において，部

表 8 部分単語化によるデータごとの固有表現の平均単語数の変化

	標準	Sub 1k	Sub 2k	Sub 3k
学習	1.34	3.46	2.86	2.47
開発	1.32	3.40	2.88	2.49
評価	1.32	3.84	3.12	2.76

表 9 Subword-NMT 適用による効果 (F 値の変化)

	標準	Sub 1k	Sub 2k	Sub 3k
学習	99.98	99.94	99.97	99.92
開発	80.58	80.45	78.62	78.69
評価	77.42	**80.94**	79.56	78.38

分単語へ分割しない設定（表中の標準）よりも部分単語を用いた設定のほうが F 値が高い結果となった．このことから，固有表現の単語数が増えることによる難しさよりも，未知語であることの難しさの方が上回るのではないかと考えられる．

部分単語を用いたことで，抽出できるようになった固有表現，できなくなった固有表現の例を表 11，表 12 に示す．表 11 中の最初の例では，分割前の設定では JAK1 のみ抽出できたが，分割後では後ろの JAK2 も同じ部分文字列が続いていることから抽出できるようになったのではないかと考えられる．二つ目の例では，分割によって，データ中に頻出する alpha の文字列が処理単位となったことで，抽出できるようになったのではないかと考えられる．三つ目の例では，部分単語によって分割されたことで，正式名称中に存在する文字列を捉えることができたため，後ろの略称名部分も抽出することができるようになったのではないかと考えられる．一方，表 12 の例では，分割し過ぎたことで，比較的長い文字列で構成されていた特有の文字列構造が失われてしまったため，抽出できなかったのではないかと考えられる．

次に類似度素性を用いた提案手法の結果を表 10 に示す．表は学習中で開発データの F 値が最も高かったエポック時の評価データの F 値である．表内の (ベスト) は，α の各設定中で，最も開発データが高かったエポック時の評価データの F 値

表 10 類似度素性による F 値の変化

	標準	Sub 1k	Sub 2k	Sub 3k
Ib-model	77.42	80.94	79.56	78.38
Ib-model を追加学習	77.60	81.65	79.51	78.32
Nb-model(ベスト)	78.02	**82.28**	79.80	79.06

表 11 部分単語によって抽出できるようになった固有表現の例

分割前	分割後 (Sub 1k)
JAK1 and JAK2	**J A K 1** and **J A K 2**
C /EBPalpha	**C / EB P alpha**
CD138/syndecan-1 (syn-1)	**CD1 3 8 / syn d ec an -1** (**syn -1**)

を比較し，最も高かった α の設定（標準: 1.0，Sub 1k: 0.2，Sub 2k: 0.2，Sub 3k: 0.2）の F 値である．提案手法では，Ib-model，並びに Ib-model を追加学習した結果よりも抽出精度の改善が見られた．このことから，評価データから得られた類似度素性がモデルの改善に役立ったことが分かった．

今回の手法の改善点として，以下の三つの点が挙げられる．一つ目は，専用の単語埋め込みベクトルを用意することである．バイオインフォマティクス分野のテキストで学習された部分単語用の単語埋め込みベクトルを用意することで精度の改善に繋がると考えられる．二つ目は，Sf-model の構築である．今回の Sf-model の構築には評価データと開発データを用いたが，より正しい類似度を計算するために，少量の評価データのみではなく，大量の論文を用いてモデル学習が必要であると考えられる．部分単語を適用した論文を用いることで，分割しない設定よりもさらに正しい類似度が算出できると考えられる．三つ目は，Nb-model の改善である．類似度素性の与え方は他にも考えられる．別の手法でも試すことで，より効果のある活用方法を検討すべきである．加えて，今回の実験ではパラメータ α を手動で決定していたため，より高い精度になるように α の最適値をモデルが自動学習できるようにする必要がある．

表 12 部分単語によって抽出できなくなった固有表現の例

分割前	分割後 (Sub 1k)
glucocorticoid receptor	g luc oc or ti co id receptor

6. 結論

本論文では，科学技術論文からの知識抽出を想定した固有表現抽出の精度改善のために，二つの提案を行った．一つ目は，非即時的なタスクにおける固有表現抽出の抽出精度を改善するため，評価データの活用に着目した．提案手法では，評価データを用いて各データの類似度を計算し，その類似度を追加の特徴として学習する方法を用いた．実験では，現在，固有表現抽出で高い抽出精度を示しているニューラルネットによる条件付き確率場による手法の抽出精度を改善することができた．二つ目は，前処理として部分単語へ分割する方法である．これはタンパク質の固有表現によく現れる部分文字列を捉えること，未知語を低減する目的で，部分単語によるデータセットの部分文字列への分割を行った．実験では，部分単語の前処理を用いたことで，抽出精度の向上することができた．

今後のタスクとして二つの方針が考えられる．一つ目は，Sf-model にあったモデル構造を構築することである．提案手法では単純化のため，Ib-model，Sf-model，Nb-model の基本的なモデル構造は同じものを用いている．しかし，Ib,Nb-model では固有表現抽出を，Sf-model では類似度計算を目的とするため，それぞれのモデルの目的が異なる．そのため，類似度計算に最適な Sf-model の構造を考える必要がある．二つ目は，提案手法に能動学習を取り入れることである．一般的には，長い時間間隔を繰り返して知識抽出を行うため，自動知識抽出システムの開発の間に人手によってアノテーションされた追加データが得られることが期待できる．それらは大量には得られないものの，モデルの性能改善には大変有用である．そのため，能動学習を用いることで，モデルの性能改善に効果があるデータを選択し，人間からの正解のフィードバッグを活用する必要がある．近年では，ニューラルネットワークと能動学習を組み合わせた手法 [26, 27] も提案されているため，能動学習を知識抽出システムの一部として取り入れることで拡張したい．

謝辞

本論文を執筆するにあたり，多くの方々のご協力とご支援を賜りました．深く感謝いたします．主指導教員である松本裕治教授には，論文解析勉強会，関係抽出勉強会において，研究に関するご指導，ご鞭撻いただきました．副指導教員である新保仁准教授には，定例研究会での進捗報告の際，有益なアドバイスをしていただきました．進藤裕之助教には，研究テーマや，手法に関する相談に乗っていただきました．能地宏助教には，勉強会において，研究方法に関するアドバイスをしていただきました．後藤啓介客員研究員には，関係分野の論文紹介や，手法に関するアドバイスをしていただきました．NTT コミュニケーション科学基礎研究所鈴木潤主任研究員には，論文の執筆から研究の指導まで，多くのご尽力とご協力をいただき，深く感謝申し上げます．北川裕子秘書をはじめ，研究室の同期，先輩，後輩，スタッフ，研究室の皆様に深く感謝申し上げます．松本研で過ごした修士学生としての二年間は，自分が学びたかった知識や技術を学ぶことができ，よい研究生活を送ることができました．加えて，新しい研究成果を共有し，活発に議論できたため，自分のアイディアや実験について，多角的な視点から考察することができました．お世話になった皆様に心から感謝いたします．

参考文献

[1] Kenji Yoshimura, Toru Hitaka, Sho Yoshida, et al. Automatic extraction system of technical terms in japanese science and technology sentences. *Journal of Information Processing Society of Japan*, Vol. 27, No. 1, pp. 33–40, 1986 (in Japanese).

[2] Fuchun Peng and Andrew McCallum. Information extraction from research papers using conditional random fields. *Information processing & management*, Vol. 42, No. 4, pp. 963–979, 2006.

[3] Lev Ratinov and Dan Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pp. 147–155. Association for Computational Linguistics, 2009.

[4] Christopher D Manning and Hinrich Schütze. *Foundations of statistical natural language processing*, Vol. 999. MIT Press, 1999.

[5] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *the 18th International Conference on Machine Learning 2001*, pp. 282–289, 2001.

[6] 山田寛康, 工藤拓, 松本裕治. Support vector machine を用いた日本語固有表現抽出. 情報処理学会論文誌, Vol. 43, No. 1, pp. 44–53, 2002.

[7] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Machine Learning Research*, Vol. 12, pp. 2493–2537, 2011.

[8] James Hammerton. Named entity recognition with long short-term memory. In *the seventh conference on Natural language learning at the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2003-Volume 4, pp. 172–175. Association for Computational Linguistics, 2003.

[9] Jason PC Chiu and Eric Nichols. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, Vol. 4, pp. 357–370, 2016.

[10] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*, 2016.

[11] Shotaro Misawa, Motoki Taniguchi, Yasuhide Miura, and Tomoko Ohkuma. Character-based bidirectional lstm-crf with words and characters for japanese named entity recognition. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pp. 97–102, 2017.

[12] Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. Semi-supervised sequence tagging with bidirectional language models. *arXiv preprint arXiv:1705.00108*, 2017.

[13] Ralph Grishman and Beth Sundheim. Message Understanding Conference-6: A Brief History. In *International Conference on Computational Linguistics*, Vol. 96, pp. 466–471, 1996.

[14] 新里圭司, 関根聡, 吉永直樹, 鳥澤健太郎. 固有表現抽出手法を用いたレストラン属性情報の自動認識. 言語処理学会第 12 回年次大会発表論文集, pp. 93–96, 2006.

[15] 森信介, 山肩洋子, 笹田鉄郎, 前田浩邦. レシピテキストのためのフローグラフの定義. 情報処理学会研究報告. 自然言語処理, Vol. 2013, No. 13, pp. 1–7, 2013.

[16] Suzushi Tomori, Takashi Ninomiya, and Shinsuke Mori. Domain specific named entity recognition referring to the real world by deep neural net-

works. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Vol. 2, pp. 236–242, 2016.

[17] Jason Fries, Sen Wu, Alex Ratner, and Christopher Ré. Swellshark: A generative model for biomedical named entity recognition without labeled data. *arXiv preprint arXiv:1704.06360*, 2017.

[18] Bo Han, Alan Ritter, Leon Derczynski, Wei Xu, and Tim Baldwin. Proceedings of the 2nd workshop on noisy user-generated text (wnut). In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, 2016.

[19] Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, Vol. 1. Wiley New York, 1998.

[20] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pp. 189–196. Association for Computational Linguistics, 1995.

[21] Sampo Pyysalo, Tomoko Ohta, and Jun'ichi Tsujii. Overview of the entity relations (rel) supporting task of bionlp shared task 2011. In *Proceedings of the BioNLP Shared Task 2011 Workshop*, pp. 83–88. Association for Computational Linguistics, 2011.

[22] Franck Dernoncourt, Ji Young Lee, and Peter Szolovits. NeuroNER: an easy-to-use program for named-entity recognition based on neural networks. *Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2017.

[23] Lance A Ramshaw and Mitchell P Marcus. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pp. 157–176. Springer, 1999.

[24] Erik F Tjong Kim Sang and Sabine Buchholz. Introduction to the conll-2000 shared task: Chunking. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7*, pp. 127–132. Association for Computational Linguistics, 2000.

[25] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

[26] Yanyao Shen, Hyokun Yun, Zachary C Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*, 2017.

[27] Maolin Li, Nhung Nguyen, and Sophia Ananiadou. Proactive learning for named entity recognition. *BioNLP 2017*, pp. 117–125, 2017.

表 13 α の変化による F 値の変化 (学習に使用したデータ：評価,Sf-model のネットワーク構造：Bi-LSTM-CRF)

α	F 値	F 値 (0 エポック)
0.0	82.14	82.70
0.001	82.70	82.70
0.005	82.02	82.70
0.01	81.48	82.70
0.05	81.55	82.70
0.1	81.89	82.73
0.2	82.75	82.75
0.5	82.41	82.80
1.0	81.74	82.75

付録

A. Sf-model の構築方法の違いによる精度の変化

第 4 章で行った実験では，Sf-model の設定として，評価データと開発データを用いてモデル学習し，モデルのネットワークに Bi-LSTM を用いた．本章では，Sf-model を別設定 (評価データのみでのモデル学習，学習・評価・開発データでのモデル学習)，別ネットワーク構造 (Bi-LSTM-CRF モデル) で構築し，得られた類似度素性を用い，手法 1 の Nb-model を構築した予備実験の結果を示す．

実験結果を表 13，表 14，表 15 に示す．類似度素性の構築には，評価データのみでのモデル構築や，評価・開発データでのモデル構築でなく，持っている全てのデータも用いることが Nb-model の精度改善に役立つことが分かった．このことから，さらに大量のテキストデータを用いて Sf-model を構築することで，さらに正確な類似度素性の計算を行う必要があり，これによって，さらなる Nb-model の精度改善が期待できる．

表 14　α の変化による F 値の変化 (学習に使用したデータ：評価・開発, Sf-model のネットワーク構造：Bi-LSTM-CRF)

α	F 値	F 値 (0 エポック)
0.0	82.13	82.70
0.001	81.87	82.70
0.005	82.23	82.73
0.01	82.47	82.79
0.05	81.91	82.63
0.1	81.34	82.83
0.2	81.90	82.86
0.5	82.96	82.96
1.0	83.07	83.07

表 15　α の変化による F 値の変化 (学習に使用したデータ：学習・評価・開発, Sf-model のネットワーク構造：Bi-LSTM)

α	F 値	F 値 (0 エポック)
0.0	82.50	82.70
0.001	81.59	82.61
0.005	81.97	82.64
0.01	82.61	82.61
0.05	82.11	82.58
0.1	81.64	82.65
0.2	82.72	82.72
0.5	82.67	83.01
1.0	83.23	83.23

表 16 Subword-NMT 適用による効果 (Ib-model，単語埋め込みベクトルあり)

	標準	Sub 1k	Sub 2k	Sub 3k
学習	99.93	99.86	99.90	99.88
開発	83.62	81.69	81.31	81.16
評価	82.70	81.95	81.72	81.07

B. 部分単語を用いた固有表現抽出 (単語埋め込みベクトルあり)

第 5 章の実験では，固有表現抽出モデルで標準で用いられる学習済みの単語埋め込みベクトルがバイオインフォマティクス分野のテキストで学習されていないベクトルであること，加えて，データセットに部分単語を適用する効果を明確にするため，学習済みの単語埋め込みベクトルを使用しなかった．本章では，標準で用いられる学習済みベクトルを使用して Ib-model と Nb-model の各手法 (手法 1，手法 2) の予備実験を行った結果を示す．第 5 章と同様に，手法 1 は Ib-model のベストモデルを用いて，α の各パラメータごとに再学習し，手法 2 はモデルに入力する単語ベクトルの次元数が Ib-model と異なるため，類似度素性を単語ベクトルに連結して，新たに学習を行う．

実験結果を表 16，表 17，表 18，表 19，表 20 に示す．実験結果から，標準での固有表現抽出モデルの精度は大きく向上したものの，部分単語で分割された単語用の学習済みのベクトルがないために，部分単語に分割した設定では，標準よりも精度の伸びが低い結果となった．一方で，手法 2 で構築を行った設定では，部分単語に分割した設定のほうが，精度の伸びがよい結果となった．

表 17 類似度素性による F 値の変化 (標準)

モデル	F 値	F 値 (0 エポック)
Ib-model	82.70	-
Nb-model 手法 1(α)		
0.0	82.33	82.70
0.001	82.03	82.61
0.005	81.80	82.63
0.01	81.94	82.61
0.05	81.91	82.63
0.1	82.64	82.64
0.2	82.63	82.63
0.5	82.92	82.92
1.0	81.11	82.94
Nb-model 手法 2	81.14	-

表 18 類似度素性による F 値の変化 (Sub 1k)

モデル	F 値	F 値 (0 エポック)
Ib-model	81.95	-
Nb-model 手法 1(α)		
0.0	82.57	81.95
0.001	82.52	81.98
0.005	82.51	81.97
0.01	82.50	81.95
0.05	82.40	82.06
0.1	82.87	82.26
0.2	83.14	82.67
0.5	82.57	82.88
1.0	81.27	82.71
Nb-model 手法 2	83.26	-

表 19 類似度素性による F 値の変化 (Sub 2k)

モデル	F 値	F 値 (0 エポック)
Ib-model	81.72	-
Nb-model 手法 1(α)		
0.0	82.60	81.72
0.001	82.34	81.75
0.005	81.92	81.77
0.01	82.50	81.84
0.05	82.69	82.02
0.1	82.10	82.32
0.2	82.69	82.69
0.5	82.74	82.74
1.0	81.85	82.57
Nb-model 手法 2	82.81	-

表 20 類似度素性による F 値の変化 (Sub 3k)

モデル	F 値	F 値 (0 エポック)
Ib-model	81.07	-
Nb-model 手法 1(α)		
0.0	81.89	81.07
0.001	81.64	81.07
0.005	81.87	81.08
0.01	81.92	81.12
0.05	82.03	81.25
0.1	80.88	81.41
0.2	81.18	81.70
0.5	81.60	81.82
1.0	81.74	81.85
Nb-model 手法 2	81.33	-

表 21 類似度素性による F 値の変化 (標準)

モデル	F 値	F 値 (0 エポック)
Ib-model	77.42	-
Nb-model 手法 1(α)		
0.0	77.60	77.42
0.001	77.29	77.40
0.005	77.40	77.40
0.01	77.76	77.40
0.05	77.62	77.43
0.1	77.62	77.62
0.2	77.77	77.77
0.5	77.86	78.86
1.0	78.02	78.02
Nb-model 手法 2	79.81	-

C. 部分単語を用いた固有表現抽出 (単語埋め込みベクトルなし)

第 5 章の表 10 では，Nb-model の手法 1 の α の各設定中で，最も開発データが高かったエポック時の評価データの F 値を比較し，最も高かった α の設定（標準: 1.0，Sub 1k: 0.2，Sub 2k: 0.2，Sub 3k: 0.2）の F 値のみを表示している．本章では，第 5 章で省略した，各 α の設定での F 値と手法 2 で Nb-model を構築した際の F 値を示す．手法 1 は Ib-model のベストモデルを用いて，α の各パラメータごとに再学習し，手法 2 はモデルに入力する単語ベクトルの次元数が Ib-model と異なるため，類似度素性を単語ベクトルに連結して，新たに学習を行っている．

実験結果を表 21，表 22，表 23，表 24 に示す．結果から，手法 1 では，Ib-model に類似度素性を足した状態では，F 値が上昇するものの (表中の 0 エポック)，再学習をしていくと，α の値が適切な場合には，F 値が上昇することが分かる．そのため，α の自動学習によって再学習を行う必要があると考えられる．

表 22 類似度素性による F 値の変化 (Sub 1k)

モデル	F 値	F 値 (0 エポック)
Ib-model	80.94	-
Nb-model 手法 1(α)		
0.0	81.65	80.94
0.001	81.05	80.94
0.005	81.38	80.95
0.01	81.85	81.03
0.05	81.82	81.21
0.1	81.31	81.40
0.2	82.28	81.66
0.5	81.21	81.66
1.0	81.84	81.33
Nb-model 手法 2	81.23	-

表 23 類似度素性による F 値の変化 (Sub 2k)

モデル	F 値	F 値 (0 エポック)
Ib-model	79.56	-
Nb-model 手法 1(α)		
0.0	79.51	79.56
0.001	79.42	79.56
0.005	79.07	79.60
0.01	78.86	79.62
0.05	79.53	79.78
0.1	79.70	79.95
0.2	79.80	79.92
0.5	79.19	80.26
1.0	79.05	80.05
手法 2	78.76	-

表 24 類似度素性による F 値の変化 (Sub 3k)

モデル	F 値	F 値 (0 エポック)
Ib-model	78.38	-
Nb-model 手法 1(α)		
0.0	78.32	78.38
0.001	78.55	78.38
0.005	78.43	78.43
0.01	78.36	78.44
0.05	78.22	78.58
0.1	78.18	78.79
0.2	79.06	79.06
0.5	78.38	78.99
1.0	77.84	78.80
Nb-model 手法 2	78.48	-

D. 外部発表一覧

国際会議論文

[1] Atsuki Sawayama, Jun Suzuki, Hiroyuki Shindo, Yuji Matsumoto. Improving Named Entity Recognition in Tasks with Non-immediate Response Setting. Second International Workshop on SCIentific DOCument Analysis (SCIDOCA 2017), Tokyo, Japan, November 15, 2017.

全国大会

[2] 澤山 熱気，鈴木 潤，進藤 裕之，松本 裕治．非即時的なタスク設定における固有表現抽出の改善．　言語処理学会第 24 回年次大会 (NLP2018)，2018(発表予定)．