Distant Supervision における 事前学習によるノイズ削減法

須藤 広大　　能地 宏　　松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
{sudo.kodai.rx2, noji, matsu}@naist.is.jp

1 はじめに

関係抽出とは, 文書から構造化された情報を抽出する情報抽出の分野の 1 つであり, 現実世界の実体 (**エンティティ**) 間の関係を表すような文から実体間の意味関係を抽出する処理である. 例えば, Apple, Steve Jobs といった実体間の意味関係を表す文から, founders という関係を抽出する. 近年, 関係抽出の分野では, 従来の教師あり学習, 教師なし学習, 半教師あり学習の手法とは異なる Distant Supervision と呼ばれる, コストをかけずに大量のラベル付きデータを生成して, 学習を行う手法が盛んになっている. [3, 4, 5, 6, 7, 8]

Mintz らが提案した Distant Supervision [5] は Freebase [1] などの知識ベースに存在する事実 (実体間の関係) を教師として, 知識ベースの実体を参照している実体の言及 (**エンティティメンション**) が一文に二つ含まれているとき, 知識ベースの事実に付与されている関係ラベルをヒューリスティックに文に付与することで疑似ラベルが付与されたデータを生成して, 学習を行う手法である. 図 1 に, Distant Supervision によるデータ生成過程の例を示す. 知識ベースに存在する事実 (例: founders(Apple, Steve Jobs)) を用いて, 二つの実体言及のペア (例: e1: Apple, e2: Steve Jobs) の両方を含んだ文 (**インスタンス**) を抽出する. このとき, インスタンスの集合を**バグ**と呼び, 知識ベースに存在する事実の関係ラベル (例: founders) を, バグの関係ラベルとして, 疑似ラベルデータを生成する.

しかし, ヒューリスティックな手法のために, 正解ではない関係ラベルをインスタンスに付与してしまうことがあり, 学習過程で関係ラベルの不確かさについて考慮しなければならない.

本研究の貢献は以下の二点にまとめられる.

1. 本タスクで最も用いられているデータセットにおける, バグレベルのノイズ (正解のインスタンスを含まないバグ) による問題点を分析する.
2. 事前学習によるノイズ削減法をおこなうことで, 従来手法よりも高い性能の関係抽出器を実現する.

Riedel ら [6] は, Distant Supervision で置かれた仮定を緩和してモデル化することで, Mintz ら [5] のモデルを上回る性能を報告した. Zeng ら [8] は緩和した仮定を用いて, 素性の設計を行わない区分的畳み込みニューラルネットワーク (Piecewise Convolutional Neural Network; PCNN) を提案しており, 関係抽出において, 高い性能を出している. [1]

[1]報告されている中では, Yankai ら [4] によるモデルが現在の最高精度とされているが, 公開されているソースコードを用いても, 再現性がなかったため, 本実験では Zeng ら [8] の手法をベースラインとした.

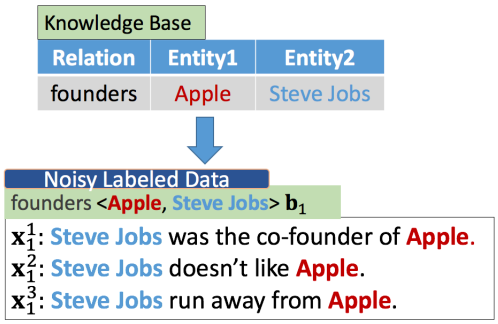

図 1: Distant Supervision によるデータ生成過程の例

Zeng ら [8] の手法では, バグ内の予測確率が一番高いインスタンスのみを学習に用いるマルチインスタンス学習 [2] を組み合わせることで, バグ内で文レベルのノイズ削減を行い, 性能向上を目指していた. しかし, バグレベルのノイズについては考慮しておらず, 正しくない関係ラベルを持つインスタンスのみで構成されているバグも学習に用いていた.

本研究では, バグレベルでのノイズ削減を目指した事前学習を行うことで, 従来手法より高い性能を実現した. 本研究の構成は以下の通りである. 第 2 節では, ベースラインとなる PCNN を用いた関係抽出について, 概説する. 第 3 節では, 本研究で使用したデータセットの分析と, 従来手法の問題点の考察を述べる. 第 4 節では, 本研究で提案する事前学習によるノイズ削減法について述べる. 第 5 節では, 実験設定と結果の考察を述べる. 第 6 節では, まとめを述べる.

2 PCNN を用いた関係抽出

2.1 Distant Supervision を用いた関係抽出タスク

以降, 本稿では, データセットを $\mathcal{D} = \{(\mathbf{b}_i, \mathbf{y}_i) \mid i = 1, ..., T\}$ と定義する. T はデータセットに存在するバグの数を表す. データセット $\mathcal{D}$ はバグ $\mathbf{b}_i$ と対応するラベル $\mathbf{y}_i$ で構成されている. バグ $\mathbf{b}_i$ は $\mathbf{b}_i = \{\mathbf{x}_i^j \mid j = 1, ..., M\}$ のように, 文 $\mathbf{x} \in \mathbb{R}^S$ で構成されている. S は最大の文の長さを表し, 最大長に足りない文にはゼロパディングを行う. M は各バグに存在する文の数を表し, 各バグ

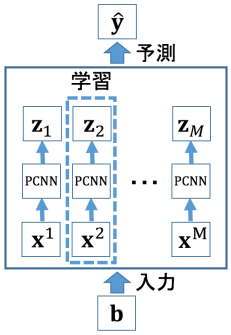

図 2: 全体のモデルの概要

によって, その数は異なる. このとき, 文 $\mathbf{x}$ は, $\mathbf{x} = (x_1, ..., x_S)$ のように, 単語の ID x で構成される.

このデータセットには, ℓ 種類の関係ラベルが存在し, $\mathbf{b}_i$ のラベルを $\mathbf{y}_i \in \{0,1\}^{(\ell+1)}$ と表す. このベクトルの k 次元目 y_i^k が 1 であれば, i 番目のバグが k 番目の関係ラベルを持っていることを表し, 0 であれば, k 番目の関係ラベルを持っていないことを表す. ただし, $y_i^{(\ell+1)}$ が 1 であれば, このバグは何の関係ラベルも持たないことを表す.

本タスクは, バグ $\mathbf{b}$ を入力として, $\hat{\mathbf{y}} \in [0,1]^{(\ell+1)}$ を出力するタスクとして, 定式化する. $\hat{\mathbf{y}}$ は, バグ $\mathbf{b}$ をモデルに入力したときの, モデルが示す関係ラベルの予測確率を表す.

2.2 PCNN

本研究では, Zeng ら [8] による PCNN をベースラインとして用いる. バグ $\mathbf{b}$ を入力とした全体のモデルの概要を図 2 に示す. 詳細は後述するが, PCNN は文 $\mathbf{x}$ を入力として, 関係ラベルの予測確率値 $\mathbf{z} \in [0,1]^{(\ell+1)}$ を出力する.

入力層

入力層では, PCNN の入力で与えられる文 $\mathbf{x}$ の要素である単語 ID x_s をそれぞれ素性ベクトル $\mathbf{f}_s$ に変換する. 文 $\mathbf{x}$ を入力として, 出力する行列 $\mathbf{F}$ は, $[\mathbf{f}_1, ..., \mathbf{f}_S]$ で構成される.

x_s に割り当てられている単語 ID から, 単語ベクトルと e1, e2 の位置ベクトルを全て結合したベクトル $\mathbf{f}_s$ に変換する. 位置ベクトルとは, 文中における e1, e2 の相対距離を表したベクトルである. 単語ベクトルと位置ベクトルは, それぞれ単語分散行列 $\mathbf{E} \in \mathbb{R}^{d_w \times W}$ と位置分散行列 $\mathbf{P} \in \mathbb{R}^{d_p \times P}$ から得る. 単語行列 $\mathbf{E}$ は, W 個の d_w 次元単語ベクトルから成り, 位置分散行列 $\mathbf{P}$ は, P 個の d_p 次元位置ベクトルから成る. よって, 一つの単語を表現するベクトル次元は, $d = d_w + d_p \times 2$ となり, 文の素性行列は, $\mathbf{F} \in \mathbb{R}^{d \times S}$ となる.

畳み込み層

畳み込み層で用いる重み行列 $\mathbf{W}_{conv} \in \mathbb{R}^{(d+w) \times H}$ は, $\mathbf{W}_{conv} = [\mathbf{w}_1, ..., \mathbf{w}_H]$ と表し, H はフィルターの数と定義する. 文の素性行列 $\mathbf{F}$ は, $\mathbf{F} = [\mathbf{f}_1, ..., \mathbf{f}_S]$ と表すことが出来る. 畳み込み層の式は,

$$c_{hs} = \mathbf{w}_h \mathbf{f}_{s-w+1:s} \ (\ 1 \leq h \leq H\ )\ (\ 1 \leq s \leq S\ ) \quad (1)$$

となる. $\mathbf{f}_{s-w+1:s}$ は, $\mathbf{f}_{s-w+1}$ から $\mathbf{f}_s$ までのベクトルを結合したベクトルを表す. w はフィルターのウィンドウ幅を表す. 畳み込み層の出力行列 $\mathbf{C} \in \mathbb{R}^{(S+w-1) \times H}$ を, $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_H]$ と表すことが出来る.

区間的プーリング層

区間的プーリング層では, $\mathbf{c}_h = (\mathbf{c}_{h1}, \mathbf{c}_{h2}, \mathbf{c}_{h3})$ の三つの区間に分ける. ($e1$ の前, $e1$ と $e2$ の間, $e2$ の後) といった三つの区間に分けることで, 通常の最大プーリング層より幅広い表現を獲得することが出来る.

その後, それぞれの区間に対して, 最大プーリングを行い,

$$p_{ht} = max(\mathbf{c}_{ht})\ (\ 1 \leq h \leq H\ )\ (\ 1 \leq t \leq 3\ ) \quad (2)$$

それぞれのベクトルを結合して, $\mathbf{p}_h \in \mathbb{R}^3$ である, $\mathbf{p}_h = (p_{h1}, p_{h2}, p_{h3})$ を得る. その後, 全てのベクトルを結合して, $\mathbf{p}_{1:H}$ を得て, 活性化関数 $tanh$ をかける.

$$\mathbf{g} = tanh(\mathbf{p}_{1:H}) \quad (3)$$

このようにして出来たベクトル $\mathbf{g} \in \mathbb{R}^{3H}$ は, 文の素性行列 $\mathbf{F}$ を畳み込んだベクトルを表す.

出力層

それぞれの関係ラベルの予測確率値 $\mathbf{z}$ を計算するため, 線形写像を行い, ソフトマックス関数を用いる.

$$\mathbf{o} = \mathbf{W}_{out}\mathbf{g} + \mathbf{b}_{out} \quad (4)$$

このとき, $\mathbf{W}_{out} \in \mathbb{R}^{(\ell+1) \times 3H}$ であり, $\mathbf{o} \in \mathbb{R}^{(\ell+1)}$ と表す.

最後に, ソフトマックス関数を用いて,

$$\mathbf{z} = p(\mathbf{y} \mid \mathbf{x}, \theta) = softmax(\mathbf{o}) \quad (5)$$

とすることで, 文 $\mathbf{x}$ が与えられたときの, 関係ラベルの予測確率値 $\mathbf{z}$ を出力することが出来る. この出力結果を用いて, モデルの学習と予測を行う.

2.3 バグに対する予測と最適化

クロスエントロピーを用いて, 目的関数 $J(\theta)$ を以下のように定義する. 以下の目的関数を最小化することで, パラメータ集合 θ を学習する. パラメータ集合 θ は ($\mathbf{E}$, $\mathbf{P}$, $\mathbf{W}_{conv}$, $\mathbf{W}_{out}$, $\mathbf{b}_{out}$) のパラメータである.

$$J(\theta) = \sum_{i=1}^{T} \log p(\mathbf{y}_i \mid \mathbf{b}_i, \theta) \quad (6)$$

モデルの学習時には, バグ $\mathbf{b}_i$ の中で最も正解ラベルの予測確率が高いインスタンス $\mathbf{x}_i^j$ のみを学習に用いるマルチインスタンス学習を行う. 学習に用いるインスタンスは, 以下のようにして選択する.

$$j^* = \arg\max_{j \in (1...M)} p(\mathbf{y}_i \mid \mathbf{x}_i^j; \theta) \quad (7)$$

前述した図 2 の例では, バグのラベル $\mathbf{y}_i$ に対して, 最も確率予測値が高かった $\mathbf{x}^2$ のみが学習に使われている. 学習の詳細設定については, 5.2.2 節で述べる.

表 1: 間違った関係ラベルが付与された文の例

関係ラベル: 「/people/person/nationality」
Ayesha Khan, the daughter of Yasmin Nighat Khan of New York and the late Karam Dad Khan, was married at the Yale Club of 671 New York yesterday to Dr. Ali Omer Farooqi, a son of Dr. Shabnam o. Farooqi and Dr. m.sultan Farooqi of Karachi, **Pakistan.**

また, モデルの予測時には,

$$\hat{y}_k = \max_{j \in (1 \ldots M)} (z_{jk}) \tag{8}$$

を得る. z_{jk} は j 番目の文 $\mathbf{x}_j$ が入力されたときの k 番目に対する予測値であり, $\hat{y}_k$ は, バグ $\mathbf{b}$ が与えられたときの k 番目に対する予測値となる. $\hat{\mathbf{y}} = (\hat{y}_1, ..., \hat{y}_{(\ell+1)})$ とすることで, バグ $\mathbf{b}$ が入力されたときの, 関係ラベルへの予測確率 $\hat{\mathbf{y}}$ を得る.

3 分析と問題点

本タスクで用いたデータセットに対して, 人手による分析を行った. データセットの詳細は 5.1 節に記述する.

訓練データからランダムにサンプリングした 35 個のバグの中に, 正解ラベルになりうるインスタンスがあるかどうかの分析を行った. 20 個のバグが正解のインスタンスを持っており, 12 個のバグが不正解のインスタンスのみで構成されており, 3 個のバグが判断が難しいバグとなっていた. つまり, サンプリングしたデータの中では約 60%のバグを学習に用いることができるが, 40%弱のバグは学習に用いることができないと考えられる.

具体的な例を表 1 で示した. 上記の例は, 「/people/person/nationality」という関係ラベルが付与された文の例である. 「Pakistan」という実体は別の実体である「Dr. m.sultan Farooqi of Karachi」に対して意味関係を持っているが, 対象である実体「Ayesha Khan」とは何の意味関係も持っていないことが分かる.

4 事前学習によるノイズ削減法

本研究では, 前述した問題点を解決するために, Riedel ら [6] によって,

> 「もし何らかの意味関係を持つ二つのエンティティが事実として存在するならば, それらのエンティティを含むような文は, **少なくとも一つの文は**その意味関係を表す」

と置かれていた仮定を,

> 「もし何らかの意味関係を持つ二つのエンティティが事実として存在するならば, それらのエンティティを含むような文は, **少なくとも一つの文は**その意味関係を表す. **ただし, 例外が存在する.**」

と新たに仮定を置きかえて, モデルの最適化手法を提案する.

本手法のアイデアはノイズを含むデータでの学習時に, 信頼度の高いデータを用いて事前学習をすることで, ノイズを取り除いた学習を行うことである. 本研究では, 以下の三種類の手法でのルールベースによるフィルタリングを行うことで, 事前学習に用いる信頼度の高いデータを得る.

ベースフィルタリング

関係ラベルの名前を用いたフィルタリングを行う. 例えば, 「/location/location/contains」といった関係ラベルを持っている場合, 小分類である「contains」という単語から編集距離が閾値 Q 以内の単語を含む文だけをフィルタリングする.

CBOW フィルタリング

関係ラベルの名前と, CBOW を用いて学習した単語分散表現から得た類似単語 U 個を用いたフィルタリングを行う. 例えば, 「/location/location/contains」といった関係ラベルを持っている場合, 小分類である「contains」という単語から「includes」や「contained」といった類似単語 U 個から編集距離が閾値 Q 以内の単語を含む文だけをフィルタリングする.

Skip-gram フィルタリング

関係ラベルの名前と, Skip-gram を用いて学習した単語分散表現から得た類似単語 U 個を用いたフィルタリングを行う.

編集距離の閾値 Q と類似単語 U については, あらかじめ決められた値を設定し, 類似単語 U 個は NYTimes のコーパスを用いて学習した単語分散表現により得る.

5 実験

5.1 データセット

本研究で用いたデータセットは Riedel ら [6] によって作成されたデータセットを用いる. このデータセットは Hoffman ら [3] や Zeng ら [8] 等によって幅広く使われているデータセットである.

Freebase の事実は訓練データ, 評価データの 2 つのパートに分かれている. 訓練データを用いて, 2005-2006 年の NYTimes のコーパスにラベル付けしたものを訓練コーパスとして, 評価データを用いて, 2007 年の NYTimes のコーパスにラベル付けしたものを評価コーパスとしている. 53 個の関係ラベルの中には NA と呼ばれる実体間に関係が存在しないことを表すラベルが含まれている. 訓練コーパスで使用されている文は 522,611 文あり, 実体のペアは 281,270 ペアあり, 18,252 の事実が存在する. 評価コーパスで使用されている文は 172,448 文あり, 実体のペアは 96,678 ペアあり, 1950 の事実が存在する. また, 本実験では, コーパスは全て小文字処理とトークナイズ処理が行われているデータであり, Yankai ら [4] が実験に用いたデータを使用する.

5.2 実験設定

5.2.1 評価手法

Mintz ら [5] や Zeng ら [8] によって, 用いられている評価手法で評価を行う. テストコーパスから発見した事実の上位 2000 件, つまり, テストコーパス $\mathcal{D}$ を入力としたときのモデルの予測 $\hat{\mathbf{y}}$ の中で, 予測確率が高いもの上位 2000 件が, 評価データの Freebase 中に存在するかどうか, 実験を行い, 適合率-再現率曲線による評価と AUC での評価を行った. この評価手法は, 人手での評価のように時間がかからず, アノテーションも必要がないため, 適切な精度の評価をコストなしに行う事が出来る.

表 2: ハイパーパラメーター一覧

パラメーター	値
ウィンドウ幅 w	3
フィルター数 H	230
単語ベクトル次元 d_w	50
位置ベクトル次元 d_p	5
ドロップアウト率 p	0.5
バッチ数 $Batch$	160
学習率 λ	0.003
最適化手法 Opt	SGD

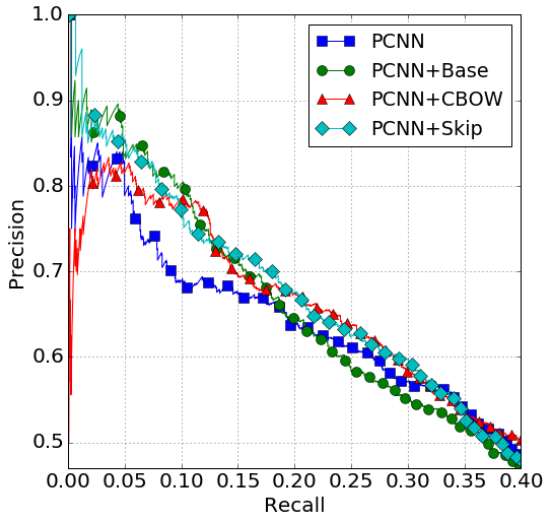

図 3: 適合率-再現率曲線の比較結果

5.2.2 実装詳細

モデルの実装には, TensorFlow[2] を用いて, CPU(Intel(R) Xeon(R) CPU E5-2699 v3 @ 2.30GHz) の環境下で実行した. 本研究の実験は, 配布されている訓練データを 9:1 の割合で訓練データと開発データに分割している. 実験結果は全てエポック数 30 に設定しており, 開発データの AUC 値が最も高いエポックで, 評価データに対する評価を行った.

CBOW フィルタリング, Skip-gram フィルタリングに用いた単語分散表現の学習には, gensim [3] を用いた.

使用したハイパーパラメータの一覧は表 2 に示す. 各ハイパーパラメータは, Zeng ら [8] と Yankai ら [4] の実験設定を参考に選択した. 編集距離の閾値 Q と類似単語個数 U については, $Q \in \{2,3\}$, $U \in \{1,4,6\}$ と設定した. また, 重みパラメータ行列は, それぞれ [-0.2, 0.2] から正規分布に従ってサンプリングした値を設定し, モデルに用いた単語分散表現の初期値には, CBOW を用いて, NYTimes コーパスを学習した Yankai ら [4] が実験に用いた単語分散表現を使用した.

5.3 結果と考察

提案手法と比較手法の適合率-再現率曲線を 図 3 に示す. 図 3 の関係ラベルはそれぞれ従来手法 (□ 印), ベースフィルタリング (○ 印), CBOW フィルタリング (△ 印), Skip-gram フィルタリング (◇ 印) の中で最も高い精度のものを示している. 提案手法はほとんどの領域において, 従来手法より高い性能を達成している. 特に前半部分において, CBOW によるフィルタリングを除いた全ての提案手法が, 従来手法より高い性能を達成している.

次に, 表 3 にそれぞれの AUC 値の結果を示した. 事前学習を行う事で, AUC 値が最大で 1.85 ポイント向上している. また, 平均ではベースフィルタリングが 1.17 ポイント, CBOW フィルタリングが 0.76 ポイント, Skip-gram フィルタリングが 1.3 ポイントづつ AUC 値が向上している.

[2]https://www.tensorflow.org/
[3]https://radimrehurek.com/gensim/

表 3: AUC 評価による比較結果

モデル	編集距離 Q 以内	類似単語 U 個	文の数	AUC 値
PCNN	-	-	-	27.85
+Base	2	1	2876	28.55
	3		8309	**29.49**
+CBOW	2	4	41555	28.94
		6	53648	27.68
	3	4	65000	28.82
		6	113096	**29.00**
+Skip-gram	2	4	30560	29.08
		6	43740	28.86
	3	4	62805	**29.70**
		6	74235	29.00

6 おわりに

本研究では, Distant Supervision における, 事前学習によるノイズ削減法を提案した. データセットの中で信頼度の高いデータを抽出して, 信頼度の高いデータを用いて学習した後に通常の学習を行うことで, 予測時に高い性能が得られることがわかった. 今後の課題として, 信頼度の高いデータの抽出方法の工夫, また, 他のタスクにおいて, データ拡張と組み合わせた半教師あり学習手法としての応用の検証などが考えられる.

参考文献

[1] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD*, pages 1247–1250, 2008.

[2] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1):31–71, 1997.

[3] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. In *ACL*, pages 541–550, 2011.

[4] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun. Neural relation extraction with selective attention over instances. In *ACL*, pages 2124–2133.

[5] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *ACL*, pages 1003–1011, 2009.

[6] S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In *ECMLP-KDD*, pages 148–163, 2010.

[7] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning. Multi-instance multi-label learning for relation extraction. In *EMNLP*, pages 455–465, 2012.

[8] D. Zeng, K. Liu, Y. Chen, and J. Zhao. Distant supervision for relation extraction via piecewise convolutional neural networks. In *EMNLP*, pages 17–21, 2005.