1J1-3

分散表現を用いたトリプル抽出

Triple Extraction from Texts using Distributed Representations of Words

蛭子　琢磨 *1*2　　市瀬　龍太郎 *2*1
Takuma Ebisu　　Ryutaro Ichise

*1総合研究大学院大学　　*2国立情報学研究所
SOKENDAI　　National Institute of Informatics

Knowledge graphs have been shown useful to many AI tasks these days. Triples of knowledge graphs are traditionally structured by editors or extracted from semi-structured information, however human resources for editing are expensive and semi-structured information is not common. On the other hand, most of information are stored in the form of texts. Recently, distributed representations of words have become a hot topic because of their ability that can capture meanings of words from texts. The models for them need no labels on texts, but Mikolov showed it can deal well with the word analogy task or the analogy task. This can be considered one of knowledge extraction task for finding a missing entity of a triple from texts. However accuracy rate is low if it is straightly applied to relations of Knowledge graphs because the method uses only one triple as positive example. In this paper, we extend the analogy task to use more positive example and propose quite new method to extract knowledge from texts. Experiments show that the proposed method achieves considerable improvement as compared with the baselines.

1. はじめに

現実世界の知識をどのようにコンピュータで表現するかは，AI の分野における重要な課題である．Knowledge Graph は，そのような知識をコンピュータに扱いやすい形で表現するための手段として頻繁に用いられている．Knowledge Graph とは，ノードがエンティティを表し，エッジがノード間の関係を表す，有向グラフである．通常，Knowledge Graph はエッジの始点であるヘッドエンティティh，エッジが表す関係 r，エッジの終点であるテイルエンティティt の 3 つ組であるトリプル (h, r, t) の集合として表現される．その有用性から，Knowledge Graph は様々な AI タスクに応用されてきた．現在，大量のトリプルを作るために，半構造化されたデータが用いられるが，半構造化されたデータを作るためには多くの人手が必要であり，簡単に作ることはできない．

ところで，近年，文章からの情報抽出技術の一つとして，単語を低次元ベクトルで表現する手法である，単語の分散表現が自然言語処理の分野で注目を集めている．単語の分散表現は，ラベル付けられていない文章を，ニューラルネットワーク言語モデルで処理することで得ることができる．文章は人々が知識を表現する手段として，最も頻繁に用いるものであることから，容易に集めることができるため，大量の学習データを用意することが可能である．ニューラルネットワーク言語モデルは本来，文章の穴埋め問題を解くために設計されたものであるが，それから得られる単語の分散表現を用いることにより，単語間の類似度評価や，アナロジータスクを上手くおこなえることが特に注目を集め，様々な AI タスクに利用されている．ここで，アナロジータスクとは，(a, b) と同じ関係にある $(c, ?)$ について，?に当てはまるものを答えよ，というタスクである．このアナロジータスクは，文章から新たなトリプルを探し出す，知識抽出の一つとして捉えることができる．しかしながら，この手法を直接，Knowledge Graph に用いられるような関係に応用しても，高い正答率は得られにくい．なぜなら，この方法では，一つのトリプルしか学習データとして用いることができないからである．

この論文では，より多くの正例を用いるように従来のアナロジータスクを拡張する．それにしたがって，単語の分散表現を用いて，より正確にトリプルの予測を行う手法を提案する．次に実際の Knowledge Graph を実験データとして用い，提案手法による文章からのトリプル予測の精度を評価する．

連絡先: 蛭子琢磨，総合研究大学院大学複合科学研究科情報学専攻，〒 101-8430 東京都千代田区一ツ橋 2-1-2，E-mail:takuma@nii.ac.jp

2. 単語の分散表現

単語の分散表現は，文章からニューラルネットワークを用いて学習された，単語のベクトル表現である．それらは，単語についての豊かな情報を含んでいることから，様々なタスクに用いられている．本論文では，Mikolov らによって提案された最も有名な単語の分散表現モデルである，CBOW モデル [Mikolov 13] によって得られる単語の分散表現を用いる．この章では，CBOW モデルと，それによって得られる単語の分散表現の性質について概説する．

2.1 CBOW モデル

CBOW モデルは，単語の列である学習用のコーパス $w_1, w_2, \ldots, w_n$ を，文脈と，それから予測される単語の組の正例の集合として捉える．具体的には，文脈 c_i として，単語 w_i を中心とする，サイズ L のウィンドウ内の単語の集合 $\{w_{i-L}, w_{i-L+1}, \ldots, w_{i+L}\}$ を取り，それから予測される単語として，w_i を取る．各単語 w はそれぞれ 2 つのベクトルで表現されており，それらは，文脈ベクトルの計算に用いられる $\vec{w}$ と，文脈から予測される単語の確率を計算する際に用いられる $\vec{pred}_w$ である．文脈 $c_i = \{w_{i-L}, w_{i-L+1}, \ldots, w_{i+L}\}$ について，それを表すベクトル $\vec{c_i}$ は，$\vec{w_{i-L}}, \vec{w_{i-L+1}}, \ldots, \vec{w_{i+L}}$ の平均で与えられる．ここで，文脈 c_i に対する，単語 w の予測確率は，$P(c_i, w) = \sigma(\vec{c_i} \cdot \vec{pred}_w) = \frac{1}{1+e^{-\vec{c_i} \cdot \vec{pred}_w}}$ と定められている．コーパスで (c_i, w_i) が観測されるたびに，$P(c_i, w_i)$ が高くなるように $\vec{w_i}$ と $\vec{c_i}$ は調整される．逆に，ランダムに選ばれた単語 w_N について，$P(c_i, w_N)$ は低くなるように調整される，これをネガ

ティヴサンプリングと呼ぶ．そして，CBOW モデルを学習する際の目的関数は次の通りである．

$$F = \sum_{n=1}^{n} (\log P(c_i, w_i) - k \cdot \mathbb{E}_{w_N \sim P_N}(\log P(c_i, w_N)))$$

ここで，k はネガティヴサンプリングを行う回数を定めるハイパーパラメーターであり，w_N はネガティヴサンプリングのためにランダムに選ばれた単語であり，分布 P_N に従う．通常，ハイパーパラメーター α と単語 w のコーパスにおける出現回数 $\#(w)$ を用いて，$P_N(w) \propto \#(w)^\alpha$ である．

文脈計算に用いられたベクトル $\vec{w}$ は，単語 w の分散表現と呼ばれ，様々なタスクに利用されている．

2.2 単語の分散表現を用いたアナロジータスク

Mikolov らの研究 [Mikolov 13] において，単語の分散表現を用いて，高い精度でアナロジータスクを行えることが示されている．ここで，アナロジータスクとは次に示すタスクである．

アナロジータスク: 関係 r にある単語の組 $(w_{1,1}, w_{1,2})$ を正例として用いて，与えられた $w_{2,1}$ と関係 r にある単語 $w_{2,2}$ を予測．

単語の分散表現には，このタスクを行うのに便利な特徴が備わっている．それは，単語の組 $(w_{1,1}, w_{1,2})$ と $(w_{2,1}, w_{2,2})$ が同じ関係 r にあるならば，それらの単語の分散表現の差はほぼ等しくなるということである．つまり，$\vec{w_{1,2}} - \vec{w_{1,1}} \simeq \vec{w_{2,2}} - \vec{w_{2,1}}$ が成り立つ．この性質を用いることで，単語 $w_{2,2}$ の予測は，$\vec{w_{1,2}} - \vec{w_{1,1}} + \vec{w_{2,1}}$ に近い分散表現を持つ単語を探すことで行われる．

アナロジータスクは，文章からのトリプル抽出タスクの 1 つだと考えられるが，直接既存の Knowledge Graph にこの手法を適用した場合，正答率は十分ではない．そこで本論文では，アナロジータスクを多くの正例を用いるように拡張したものである拡張アナロジータスクを定義し，それに対する新たな手法を提案し，Knowledge Graph に関する文章からのトリプル抽出に応用する．

3. 拡張アナロジータスク

アナロジータスクをそのまま既存の Knowledge Graph に適用しても，高い正答率は得られにくい．そこで，より多くの正例を使用する拡張アナロジータスクを定め，それについての新たな手法を提案する．

3.1 拡張アナロジータスク

アナロジータスクを，より多くの正例を用いるタスクに拡張する．

拡張アナロジータスク: 関係 r にある単語の組の集合 $S_r = \{(w_{i,1}, w_{i,2}) \mid i = 1, \ldots, N\}$ を正例として用いて，与えられた w_1 と関係 r にある単語 w_2 を予測．

アナロジータスクの手法を，単純に拡張アナロジータスクへ対応させた手法は，関係 r を表すベクトルとして，2 つの単語の差のベクトルの平均を取ることである．つまり，$\vec{r} = \sum_{i=1}^{N}(w_{i,1} - w_{i,2})/N$ と定め，w_2 の予測は，$\vec{w_1} + \vec{r}$ と近い分散表現を持つ単語を探すことで行う手法である．この手法は，通常のアナロジータスクより大幅に正答率を改善する．本研究では，この手法をベースラインとする．

3.2 提案手法

CBOW モデルのアルゴリズムに従うと，単語の分散表現 $\vec{w_i}$ と $\vec{pred_{w_j}}$ の内積は，ウィンドウ内で共起するほど大きくなり共起しなければネガティヴサンプリングにより小さくなる．つまり，分散表現の各成分は，他の単語との共起頻度を表している．全ての単語についての共起頻度の情報が，アナロジータスクに必要であるとは考えがたい．そこで，分散表現から，アナロジータスクに重要だと思われる成分だけを抜き出し，特定の単語との共起の情報のみを抜き出すことを考える．これを行うために，分散表現の空間 U から部分空間 U' への射影関数 $proj_{U'}$ を用いる．$proj_{U'}$ は次のように定義される．

$$proj_{U'}(\vec{v}) = \sum_{i=1}^{d-d'} (\vec{v} \cdot \vec{b_i})\vec{b_i} \quad (\vec{v} \in U)$$

ここで，d は U の次元，$d - d'$ は U' の次元，$b_1, \ldots, b_{d-d'}$ は U' の正規直交基底である．$proj_{U'}$ を用いることで，U' と直交する成分を 0 にすることができ，それに関連した単語の情報を無視することができる．

アナロジータスクでは，同じ関係にある単語の組 $(w_{1,1}, w_{1,2})$ と $(w_{2,1}, w_{2,2})$ について，$\vec{w_{1,2}} - \vec{w_{1,1}} \simeq \vec{w_{2,2}} - \vec{w_{2,1}}$ が成り立つという性質を用いていた．そこで，U' は，$proj_{U'}$ によって写像された単語の分散表現が，その条件をより満たすように定める．つまり，拡張アナロジータスクで与えられる正例 $S_r = (w_{i,1}, w_{i,2}) \mid i = 1, \ldots, N$ について，$proj_{U'}\vec{w_{1,2}} - proj_{U'}\vec{w_{1,1}}$ の分散が最小になるように U' を定める．したがって目的関数は次の通りである．

$$F = \sum_{i=1}^{N} \|proj_{U'}(\vec{w_{i,2}} - \vec{w_{i,1}} - \vec{r})\|_2^2 \tag{1}$$

この用にして求められた U' を用いて，w_2 の予測は，$proj_{U'}(\vec{w_1} + \vec{r})$ と近い，$proj_{U'}$ で射影された分散表現を持つ単語を探すことで行われる．

4. 実験

この章では，Freebase の人に関するトリプルを集めたデータセットである，FB13 を用いて，提案手法を用いた拡張アナロジータスクの実験を行う．

4.1 実験目的

U と U' の次元の差である d' は，提案手法のハイパーパラメータであった．そこで，実験 1 として，d' が，どのように提案手法に影響を与えるかを調べるために，d' の値を変更しながら，拡張アナロジータスクの正答率を，10 点交差法を用いて測定する．

実際に，文章からのトリプル抽出として提案手法を用いるためには，前もって適切な d' を定めておく必要がある．そこで，実験 2 として，適切な d' を訓練データから予測し，定めた場合について，提案手法を用いた拡張アナロジータスクを行う．それにより，提案手法のハイパーパラメータである d' を前もって定めることが可能かについて調べる．

4.2 実験データ

4.2.1 単語の分散表現

CBOW モデルの実装として，word2vec[*1] を用いる．英語版 wikipedia を処理して学習コーパスとして用いる．総単語数は

[*1] https://code.google.com/archive/p/word2vec/

表 1: FB13W の詳細

関係	# トリプル	関係	# トリプル
gender	11452	cause of death	3935
nationality	10171	religion	2614
profession	12674	parents	446
place of death	3935	children	435
place of birth	7978	ethnicity	1243
location	7170	spouse	621
institution	1820		

1.8 億である．人名は基本的に複数の単語で構成されているため，英語版 Wikipedia をそのままコーパスとして用いることはできない．そこで，word2vec と一緒に配布されているプログラムである，word2phrase を用いてコーパスに事前処理を行う．word2phrase は各単語の出現頻度と比べて，共起する頻度が高い連続する 2 つの単語をまとめて一つの熟語にするシステムである．word2phrase を用いて，英語版 Wikipedia を 2 回処理したものをコーパスとして用いる．これにより，コーパス内には，最大で 4 単語からなる熟語が含まれている．このコーパスを word2vec で学習することで，単語の分散表現を得る．分散表現空間の次元 d は 300 に設定した．分散表現を得られた単語数は，86 万であった．Mikolov らが実験に用いたものと同じように，この分散表現を正規化して実験に使用する．

4.2.2 実験に用いるトリプル

FB13 は，Freebase に含まれるトリプルのうち，人々に関するものを 13 個のリレーションについて抽出したものである．これらのトリプルのうち，ヘッドエンティティとテイルエンティティが単語の分散表現として得られたもののみを用いる．このデータセットの詳細は表 1 の通りである．

このデータを，FB13W と表す．

4.3 実験 1

FB13W のそれぞれの関係について，d' を 0 から 300 まで変更しながら 10 点交差検定を用いて拡張アナロジータスクを行い，提案手法における d' の影響を評価する．

4.3.1 実験手順

それぞれの関係について，d' を 0 から 300 まで変更しながら拡張アナロジータスクを行い，提案手法の評価を行う．

10 点交差検定のそれぞれのステップにおいて，90%のトリプルを学習データとして使用し，残りの 10%をテストデータとして使用する．学習データを用いて U' を，目的関数 1 にしたがって定める．その後，それぞれのテストデータに含まれる単語の組 (h,t) について，t を分散表現が得られた単語 w に置き換えることで，候補となる単語の組を約 86 万個作成する．その後，候補となる単語の組全てについてスコア関数 S を用いてスコアを計算し，スコアが高い順に並べる．S は次のように定められる．

$$S(h,w) = -\|proj_{U'}(\vec{w} - \vec{h}) - proj_{U'}(\vec{r})\|_2^2$$

得られたランキングにおいて，元となった単語の組が 1 位となったものの割合と 10 位以内にあるものの割合を評価基準とし，それぞれ HITS@1 と HITS@10 と表す．

比較手法として，テイルエンティティとヘッドエンティティの分散表現の差のベクトルを関係を表現するベクトルとして用いる方法を用いる．スコア関数は次のとおりである．

$$S_{AVE}(h,w) = -\|(\vec{w} - \vec{h}) - \vec{r}\|_2^2$$

表 2: CBOW モデルによる分散表現を用いた実験 1 結果

評価基準	HITS@1(%)			HITS@10(%)		
関係	AVE	提案手法	d'	AVE	提案手法	d'
gender	88.6	93.1	10	100	100	0
nationality	60.2	68.7	10	86.7	96.7	30
profession	19.8	28.1	30	51.1	67.8	30
place of death	10.1	18.6	30	27.3	42.8	20
place of birth	3.2	9.1	20	12.2	25.5	20
location	2.8	9.3	20	11.8	25.8	30
institution	7.7	15.2	10	34.4	45.7	20
cause of death	1.0	11.1	20	9.0	56.3	50
religion	19.8	34.1	20	52.5	79.5	70
parents	10.8	12.3	40	37.2	47.8	100
children	9.9	11.5	10	36.6	40.7	40
ethnicity	7	50.8	20	17.5	89.1	140
spouse	5.3	72.1	230	15.3	89	210

この方法を AVE と表記する．AVE は提案手法において $d' = 0$ とした場合と一致する．

4.3.2 実験結果

実験結果は表 2 に示す．ここで，提案手法の正答率は，d' を変更しながら測定した際のもっとも良い結果であり，そのときの d' の値を一緒に表記している．

実験により，次の結果を得た．

1. 提案手法の正答率について，AVE による正答率を全ての関係について上回る結果を得た．”religion”と”ethnicity”と”spouse”の改善率は大きく，HITS@1 において 10%以上改善し，特に”spouse”については 65%改善した．”gender”についての予測は，全ての関係の中で最も良い結果を出し，93.1%の正答率であった．これらの結果は，提案手法が不要な部分空間を無視することで，予測精度を上げることが可能であることを示している．

2. HITS@1 における d' の範囲は 10 から 40 程度であり，HITS@10 における d' の範囲は 20 から 40 程度であった．したがって，分散表現空間の次元の 300 のうち，大半の 250 程度はそのままであった．”spouse”についての d' は非常に高く，HITS@1 で 230，HITS@10 で 210 であった．

提案手法の結果が，全ての関係において AVE の結果を上回ったことから，関係を予測するために必要な単語の情報は一部であり，それを訓練データを用いて選ぶことで，より正確に拡張アナロジータスクを行うことができることが示された．

4.4 実験 2

実験 1 は実際の Knowledge Graph について，それぞれの d' について拡張アナロジータスクの実験を行った．この実験では，学習データを用いて適切な d' を予測した上で，その d' を用いて提案手法を評価する．

4.4.1 実験手順

10 点交差検定のそれぞれのステップにおいて，学習データに対して，さらに d' を変更しながら 10 点交差検定を適用し，そこで最もよい結果を得た d' を用いてテストを行う．

4.4.2 実験結果

実験結果を表 3 に示す. 実験により，次の結果を得た．

表 3: 実験 4 結果

評価基準	HITS@1(%)		HITS@10(%)	
関係	AVE	提案手法	AVE	提案手法
gender	88.6	93.1	100.0	100.0
nationality	60.2	68.7	86.7	95.6
profession	19.8	27.4	51.1	65.7
place of death	10.1	17.0	27.3	42.2
place of birth	3.2	8.4	12.2	25.4
location	2.8	9.0	11.8	24.1
institution	7.7	13.2	34.4	43.8
cause of death	1.0	10.2	9.0	54.5
religion	19.8	33.8	52.5	78.0
parents	10.8	11.0	37.2	45.3
children	9.9	9.2	36.6	38.4
ethnicity	7.0	51.5	17.5	85.7
spouse	5.3	67.6	15.3	85.8

1. 提案手法の正答率はほぼ全ての関係について AVE のものより高い．実験 1 の表で示した最も適切な d' の値を用いたときより少し低いが，それでも十分な改善を見せた．

2. ”children”についての HITS@1 の実験結果のみ AVE と比べて低い．これは，十分な学習データがないために起こったことだと考える．実際，”children”についてのトリプルの数が，データセットの中で最も少ない．HITS@10 については，提案手法の正答率のほうが AVE より高い．

3. HITS@10 における正答率の上昇は著しい．”cause of death”については 45%程度改善し，”ethnicity”については 68%, ”spouse”については 70%改善した．

”chiledren”を除いて全ての関係について提案手法の結果が従来の手法である AVE の結果を上回った．したがって，学習データから適切な d' を予測して設定することが可能であることが示された．

5. 関連研究

最後に，提案手法と，Knowledge Graph のトリプルの予測や文章からの抽出に用いられる先行研究との違いについて述べる．

5.1 Knowledge Graph Embedding モデル

Knowledge Graph Embedding モデルは，Knowledge Graph Completion を行うために，現在最も盛んに研究が行われている方法である．これの基本となるモデルは TransE[Bordes 13] である．単語の分散表現において，関係が単語の分散表現の差で現れていたことから着想を得て作られたモデルであり, 多くの亜種が存在する．Knowledge Graph Embedding モデルは，予測したいトリプルに含まれる両方のエンティティについての情報が，学習データの中に十分に含まれている必要がある．つまり，学習データのトリプルの中には存在しないエンティティを探してくるようなタスクを行うことはできない．提案手法は，一つの関係について集めたトリプルを用いて，文章の中から適切な新しいエンティティを探してくるというタスクを行うことができ，この点に違いがある．

提案手法と Knowledge Graph Embedding モデルは必要とするデータの種類が大きく異なっており，Knowledge Graph Completion のために相補的に用いることができると考える．

5.2 OpenIE

Open Information Extraction (OpenIE) は，その名の通り，文章からさまざまな関係についての情報を抽出することを表す．OpenIE モデルは，与えられた 1 つの文をトリプルに変換するモデルである．提案手法にはラベル無しの文章と予測したい関係についてのトリプルのみが必要であるが，現在最も優れた OpenIE のモデルである，Stanford OpenIE[Angeli 15] は学習の際に，大量のラベル付きのデータが必要である．抽出したい関係に合わせて，ラベル付きデータを作成するのはあまり現実的ではない．また，OpenIE によりアウトプットされたトリプルそのままでは，雑多な状態であるため，エンティティや関係をマッチングさせる処理が必要である．そして，OpenIE は一つの文章をトリプルに変換するシステムであるがゆえに，文章として明確に述べられた事実しか抜き出すことはできない．例えば，提案手法は，(Musashi Miyamoto,gender,male) というトリプルを予測することができたが，”Musashi Miyamoto”と”male”は同一の文章に出現しておらず，OpenIE のシステムでは，これを抽出することは不可能であり，また，単語間の文法的な関係を捉えることも不可能である．

6. まとめ

今日，Knowledge Graph は，様々な AI タスクや，情報の公開に利用されており，ますます重要性がましている．だが，現在存在している Knowledge Graph はまだまだ完全には程遠く，既存の方法では，様々な分野にわたって Knowledge Graph を作ることは不可能である．今後の情報科学の発展のためには，Knowledge Graph を作る新たな手法が不可欠である．

本研究では，ニューラルネットワーク言語モデルを用いて，ラベルの無い文章から有用な単語の分散表現が得られることに注目し，もともと単語の分散表現が得意としていた，アナロジータスクの手法を拡張することで，トリプルを文章から抽出する新しい手法を提案した．提案手法は，従来の方法と比べてトリプルの予測精度を大幅に向上させることができる．

参考文献

[Angeli 15] Angeli, G., Premkumar, M. J. J., and Manning, C. D.: Leveraging Linguistic Structure For Open Domain Information Extraction., in *ACL (1)*, pp. 344–354, The Association for Computer Linguistics (2015)

[Bordes 13] Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O.: Translating Embeddings for Modeling Multi-relational Data, in Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. eds., *Advances in Neural Information Processing Systems 26*, pp. 2787–2795, Curran Associates, Inc. (2013)

[Mikolov 13] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J.: Distributed Representations of Words and Phrases and their Compositionality, in Burges, C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. eds., *Advances in Neural Information Processing Systems 26*, pp. 3111–3119, Curran Associates, Inc. (2013)