深層学習によるナレッジグラフの実体の推定手法の検証

1　はじめに

近年, 人工知能技術は急速な発展を遂げている. その中でも, ナレッジグラフが注目を集めており, 人工知能の実用的な基盤技術としてさまざまな分野で活用されている. ナレッジグラフを用いることにより, 自然文や SNS の投稿, 音声などの非構造化データに対して高度な知識処理が可能となった. そのため, さまざまな分野の知識を表現するナレッジグラフが開発されている.

また, 推理小説における凶器や犯人を推定する「ナレッジグラフ推論チャレンジ」というコンテストでは, ナレッジグラフを用いて人物の関係や行動の意図, 時系列などを基に凶器や犯人の推定の結果を競っている.

そこで本実験では, ナレッジグラフ推論チャレンジにおいて小説の内容のナレッジグラフとして公開されているデータに関して, 深層学習の手法のひとつである TransE を用いてナレッジグラフの実体を推定し, その精度を検証する.

2　要素技術

2.1　ナレッジグラフ

ナレッジグラフ（ Knowledge Graph ）[1] とは, さまざまな知識を体系的に連結し, その関係をグラフ構造で表した知識のネットワークのことである. Google により実世界のオブジェクトの検索を可能にする基盤技術として紹介され, さまざまな研究や産業分野に普及された.

図 1 にナレッジグラフの例を示す. ナレッジグラフは, それぞれが意味をもつエンティティ集合（ Entity ）と, そのエンティティ同士の関係性を表現するリレーション集合（ Relation ）によって構成されており, 図 1 のようにエンティティ集合の各要素をノード, リレーション集合の各要素をエッジとする有向グラフとして表される. 2 つのエンティティ間に複数の関係がある場合はそこに複数のリレーションが付与されるため, 多重グラフとなる. なお, Relation はリレーションの種類の集合であり, Relation の総数はグラフのエッジの総数と一致しない.

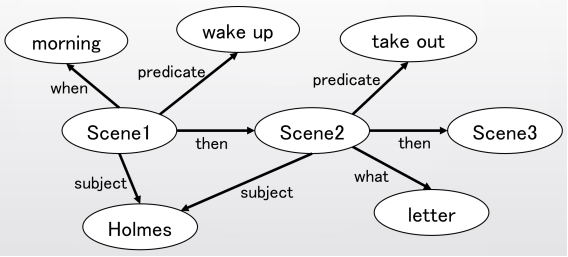


図 1: ナレッジグラフの例

また, ナレッジグラフにはグラフと異なる表現方法があり, 2 つのエンティティとそれらを結ぶリレーションを 1 つにまとめた 3 つ組構造（ triple ）として表すことができる. triple は, $head \in$ Entity と $tail \in$ Entity が $relation \in$ Relation で結ばれていることを表す $(head, relation, tail)$ として表現される. ナレッジグラフはさまざまな種類のデータを統合的にグラフとして扱うことにより, データ間の複数のつながりを学習することが可能である.

2.2　TransE

ナレッジグラフの代表的な埋込み（ Embedding ）手法として TransE [2] が挙げられる. TransE では, $\mathrm{D} = \{(h_i, r_i, t_i)\}_{i=1}^{n}$ という triple の集合（ナレッジグラフ）に対して, 以下の式を満たすように学習する.

$$v(h) + v(r) \simeq v(t) \tag{1}$$

$v(x)$ は x をベクトル化したものである. 入力としてナレッジグラフを与えると, すべてのエンティティ, リレーションに対してそれぞれ k 次元のベクトルを出力する.

$v(h), v(r), v(t) \in \mathbb{R}^k$ を $h, t \in$ Entity, $r \in$ Relation の Embedding として, 下記の目的関数を最小化する.

$$Loss(V) = \sum_{(h,r,t)\in \mathrm{D}} \max(0, f(h,r,t) + \gamma - f(h',r,t')) \tag{2}$$

ここで, $\gamma \geq 0$ はハイパーパラメータのマージンである. また, $f(h,r,t)$ は triple (h,r,t) の結びつきの良さを表すスコア関数であり,

$$f(h,r,t) = ||v(h) + v(r) - v(t)|| \tag{3}$$

表 1: 配布データ例

head	relation	tail
Scene1	subject	Holmes
Scene1	predicate	wake up
Scene1	when	morning
Scene1	then	Scene2
⋮	⋮	⋮

表 2: 加工データ例

head	relation	tail
Holmes	wake up	morning
Holmes	wake up	Scene2
⋮	⋮	⋮

として定義される. 右辺はノルムを表している. また, (h', r, t') は与えられた (h, r, t) に対して, h または t を入れ替えて得られる triple のことである.

このように, TransE はデータとして与えられた triple (h, r, t) が (1) 式を満たすように Embedding し, $v(h), v(r), v(t)$ を学習するモデルである.

2.3 データセット

ナレッジグラフ推論チャレンジにおいて, シャーロック・ホームズの 8 作品の推理小説をナレッジグラフとして表現したデータセットが公開されている. 本実験ではこのデータセットを使用する. 表 1 に配布されているナレッジグラフデータの例を示す. 本データに関して, エンティティとしてストーリーシーンや登場するキャラクター, オブジェクトの名前, 属性名, 自然言語文や数値情報をもち, リレーションはそれらの関係を表している.

また, triple 集合はストーリーの展開を説明するもの, 登場するキャラクターやオブジェクトを説明するものに分類される. ストーリーの展開を説明する triple では, head としてストーリーシーンをもっている. ストーリーシーンとは物語上の順番ごとに id が割り振られたものであり, 同一のストーリーシーンごとに triple 集合を分割すると triple の系列データとしてとらえることができる. また, 同一のストーリーシーンには複数の triple が存在し, そのシーンに対する主語や述語は relation と tail によって関係別に表されている.

3 実験

3.1 データの加工

本実験では, ナレッジグラフ推論チャレンジにおいて配布されたデータを加工して使用する. 加工データは主にシーンについて説明された triple 集合に対して, 以下の処理を施すことで得られる.

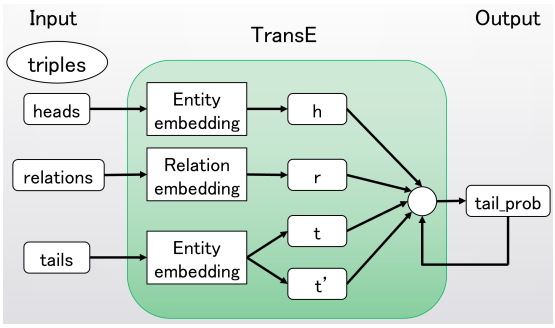


図 2: TransE モデルの概略図

1. head にあるストーリーシーンごとに triple 集合を分割する.
2. relation が ”subject” の triple の tail, relation が ”predicate” の triple の tail を抽出することでストーリーシーンにおける主語と述語を得る.
3. 得た主語と述語をそれぞれ head, relation とし, tail に他の tail をもつ新たな triple を作成する.

このようにして SVO 形式の triple 集合を作成する. 表 2 に, 表 1 の配布データを加工した例を示す.

3.2 TransE モデル

図 2 に本実験で使用する TransE モデルの概略図を示す. 本モデルの入力は triple を複数並べたデータである. 入力された各 triple の head, relation, tail を Embedding したあと, それらを結合して推定した tail のベクトルを出力する. この推定した tail のベクトルを深層学習により正解の tail のベクトルに近づけていく. つまり, (1) 式を満たすように学習していく.

3.3 実験概要

本実験では, 加工したデータのナレッジグラフに関して, TransE を用いて Embedding し, その tail の推定の精度を検証した. 扱う triple 集合のデータを 9 : 1 に分割し, それぞれを訓練データ, テストデータとした. また, 交差検証として 5 種類のデータを作成し, それらに対してモデルの学習をしてその精度を評価した. 表 3 に本実験のパラメータを示す.

表 3: 本実験のパラメータ

パラメータ	値
データ数	5843
エンティティ数	2487
リレーション数	583
埋め込み次元	32
バッチサイズ（訓練時）	5259
バッチサイズ（テスト時）	584
学習率	1.0×10^{-2}
エポック数	15000

4 実験結果

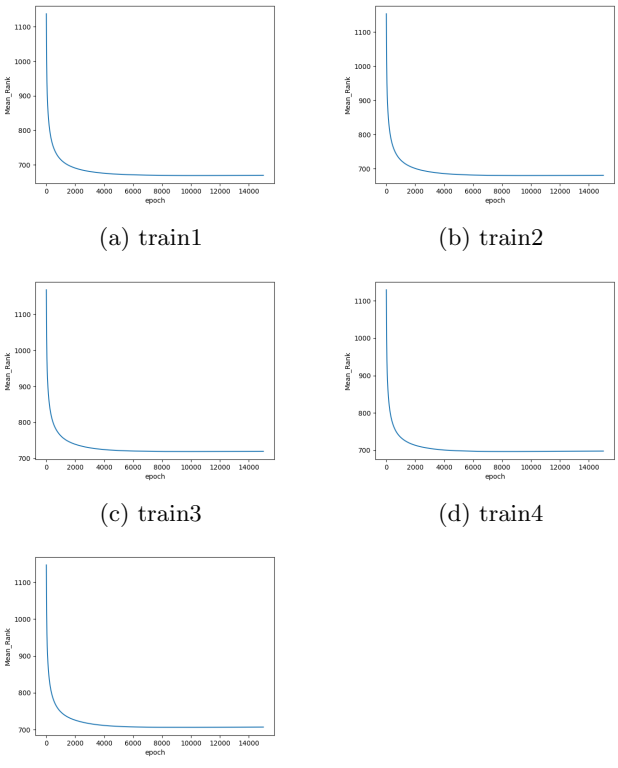
図 3, 4 はそれぞれ訓練データ, テストデータにおける loss の変動を表しており, 正解の tail とどれだけ離れているかを表すランクの平均値を縦軸, epoch 数を横軸としている.

図 3 より, 交差検証として用いた 5 種類のデータにおいて, どれも epoch 数に対して loss が減少していることがわかる. しかし, 5000 epoch あたりで loss が約 700 に収束し始めていることから, 訓練データ内に正解の tail を正確に推定できていない triple が多く存在することがわかる.

また, 図 4 より, 5 種類のデータにおいて, 訓練データと同様に, どれも epoch 数に対して loss が減少していることがわかる. さらに, 5000 epoch あたりで loss が約 1000 に収束し始めており, 正解の tail を正確に推定できていない triple がかなり多く存在することがわかる.

図 5 は累積 top 数に対するデータ数の割合の変動を表しており, 累積 top 数を縦軸, 累積 top 数に対する triple の割合を横軸としている. 累積 top 数とは, 例えば「 top3 」において, 推定した tail と正解の tail との距離が 3 未満であるデータの個数を表している. つまり,「 top1 」は tail の推定が成功したデータの個数となる. なお, top1 で数えられるデータは top3 にも含まれている. 図 4 より, 5 種類のデータにおいてどのグラフも似た形をしている. また, 累積 top 数が 500 のときに約 30 % となっていることから, 推定した tail と正解の tail との距離であるランクが 500 以内にあるものは約 175 (= テストデータ数 × 0.3) 程度であることがわかる.

表 4 はそれぞれ訓練データ, テストデータにおいて, 5 種類のデータにおける累積 top 数である top1, top3, top10 をまとめたものである. 訓練データにおいて, 5 つ目のデータが top1 = 17 で他のデータと比べると最も良い結果が得られている. 一方, テストデータにおいてはどのデータも似た結果になっていることがわかる.



(a) train1 (b) train2

(c) train3 (d) train4

(e) train5

図 3: 訓練データにおける loss の変動

表 4: 累積 top 数

(a) train データ

	top1	top3	top10
train1	6	30	102
train2	12	41	131
train3	8	34	140
train4	8	37	159
train5	17	43	151

(b) test データ

	top1	top3	top10
test1	3	7	20
test2	0	4	15
test3	1	6	20
test4	3	6	18
test5	2	5	13

表 5 は, 5 種類のテストデータに対する tail の推定において正解した例をまとめたものである. 表 5 より, head もしくは tail に ”Holmes” をもつ triple において正しく推定できていることがわかる. また, 正しく推定できた triple の head, tail はどれも小説の登場人物の名前となっている. これは, 小説において人物の名前は一般的な単語より多く登場しており, そのデータ数が比較的多くなったためであると考えられる.

以上のことから, 本実験における TransE モデルを

表 5: tail の推定において正解した例

test	head	relation	tail
1	Jack_Croker	believe	Holmes
	Holmes	ask	Percy_Trevelyan
	Gregory_Inspector	arrest	Fitzroy_Simpson
2	-	-	-
3	Holmes	sympathy	Jack_Croker
4	Henry	help	Holmes
	Blessington	invite in	Holmes
	Windibank	greet	Holmes
5	Abe_Slaney	pull out	Elsie
	Cubitt	visit	Holmes

用いて tail の推定の精度を良くするためには, 登場の少ない単語が head や tail となる triple においても考慮する必要がある. そのため, そのようなデータに対して新たな情報を triple として付加することでデータ数を増やせばよいと考えられる.

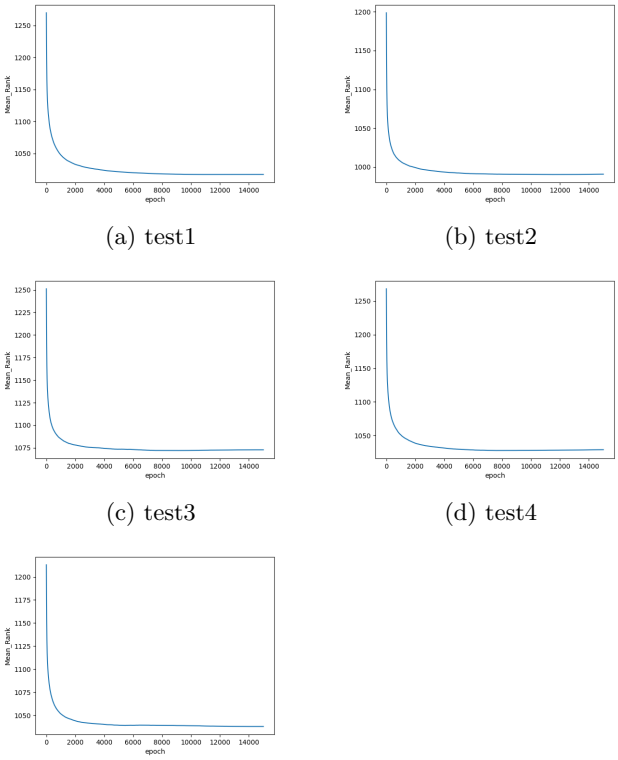
5　まとめと今後の課題

本実験では, 推理小説を題材としたナレッジグラフに対して TransE を用いて tail を推定し, その精度を検証した. 実験の結果, 小説内に多く登場する ”Holmes” などの人物名を head, tail にもつ triple の推定の精度が良くなることがわかった. 今後の課題として, 小説に登場する回数が比較的少ない単語が head や tail となる triple を考慮して, そのような単語に triple として情報を付加することでデータ数を増やすことで, より良い精度での推定を目指すことが挙げられる.

参考文献

[1] 川村隆浩, 江上周作, 田村光太郎, 外園康智, 鵜飼孝典, 小柳佑介, 西野文人, 岡嶋成司, 村上勝彦, 高松邦彦, 杉浦あおい, 白松俊, 張翔宇, 古崎晃司. 第 1 回ナレッジグラフ推論チャレンジ 2018 開催報告 ―説明性のある人工知能システムを目指して―. 人工知能 34 巻 3 号, 2019.

[2] Bordes A., Usunier N., Garcia-Duran A., Weston J., and Yakhnenko O. Translating embeddings for modeling multirelational data. *Advances in Neural Information Processing Systems, pp. 2787-2795*, 2013.

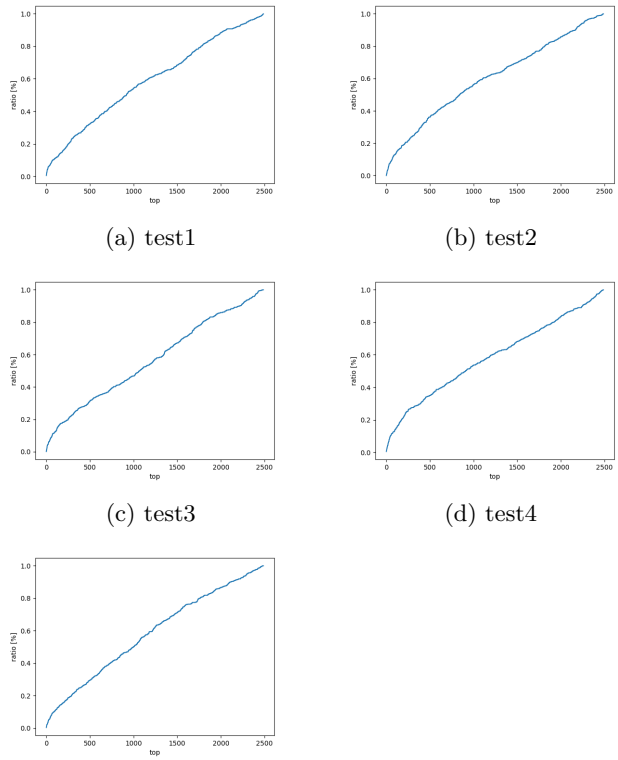


(a) test1　(b) test2

(c) test3　(d) test4

(e) test5

図 4: テストデータにおける loss の変動



(a) test1　(b) test2

(c) test3　(d) test4

(e) test5

図 5: 累積 top 数に対するデータ数の割合の変動