

# Ontologically Grounded Multi-sense Representation Learning for Semantic Vector Space Models

Sujay Kumar Jauhar, Chris Dyer and Eduard Hovy

宮澤 彬

総合研究大学院大学 博士前期 1 年

`miyazawa-a@nii.ac.jp`

August 31, 2015

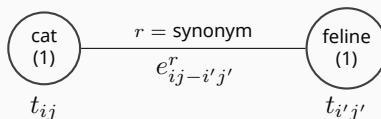
# 問題意識

ベクトル空間モデルは 1 つの語に 1 つのベクトルが対応していることが多い。→ 多義性が表現されていない。

bank /'bænk/

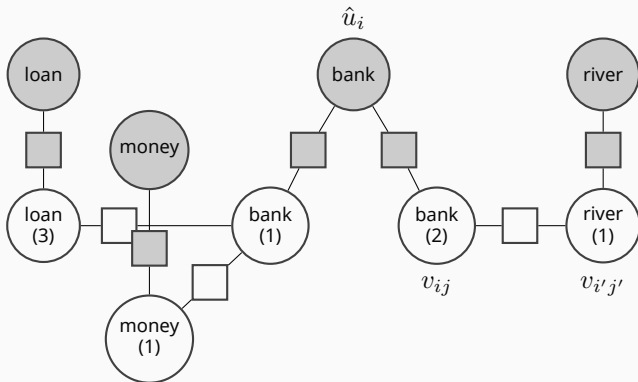
1. 銀行
2. 土手

オントロジーのグラフ表現を使い、これらに別々のベクトルを割り当てる。



# オントロジーと語のベクトル表現をどう結びつけるか

語彙のベクトル空間埋め込み  $\hat{U} = \{\hat{u}_i \mid w_i \in W\}$  とオントロジー  $\Omega$  を与えられたときに、それらと整合性のとれた語義集合  $V = \{v_{ij} \mid s_{ij} \in W_s\}$  を推測する。



灰色で塗られている頂点が語（観測される変数）に対応し、白い頂点が語義（観測されない潜在変数）に対応している。

# アルゴリズム

以下を求めたい.

$$C(V) = \arg \min_V \left\{ \sum_{i-ij} \alpha \|\hat{u}_i - v_{ij}\|^2 + \sum_{ij-i'j'} \beta_r \|v_{ij} - v_{i'j'}\|^2 \right\}$$

$\{\cdot\}$  の中を  $v_{ij}$  について微分して 0 とおくと

$$v_{ij} = \frac{\alpha \hat{u}_i + \sum_{i'j' \in \mathcal{N}_{ij}} \beta_r v_{i'j'}}{\alpha + \sum_{i'j' \in \mathcal{N}_{ij}} \beta_r}$$

となる.

# アルゴリズム

したがって以下のアルゴリズムを得る.

1: **function** RETROFIT( $\hat{U}, \Omega$ )

2:  $V^{(0)} \leftarrow \{v_{ij}^{(0)} = \hat{u}_i; \forall s_{ij} \in W_s\}$

3: **while**  $\|v_{ij}^{(t)} - v_{ij}^{(t-1)}\| \geq \varepsilon$  for all  $i, j$

4:     **for**  $t_{ij} \in T_\Omega$

5:         
$$v_{ij}^{(t+1)} \leftarrow \frac{\alpha \hat{u}_i + \sum_{i'j' \in \mathcal{N}_{ij}} \beta_r v_{i'j'}}{\alpha + \sum_{i'j' \in \mathcal{N}_{ij}} \beta_r}$$

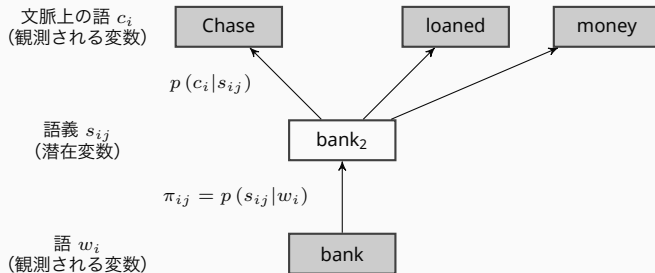
6:     **return**  $V^{(t)}$

# モデルの拡張

語  $w_i$  と文脈上の語  $c_i$  の組を集めたコーパス  $D = \{(w_i, c_i)\}_{i=1}^N$  と、オントロジー  $\Omega$  が与えられているとする。単語の意味情報を含んだベクトル表現の集合  $V = \{v_{ij} \mid \forall s_{ij} \in W_s\}$  を求めたい。

文脈上の語に対応するベクトルの集合を  $U = \{u_i \mid c_i \in W\}$ ，語義に対応するベクトルの集合を  $V = \{v_{ij} \mid s_{ij} \in W_s\}$  とする。さらに文脈を考慮しない意味の比率を  $\pi_{ij} = p(s_{ij} \mid w_i)$  と置き， $\Pi = \{\pi_{ij}\}_{ij}$  とする。

# 拡張されたモデルの枠組み



パラメータ  $\theta = (U, V, \Pi)$  をうまく選び以下を求める.

$$C(\theta) = \arg \max_{\theta} \left\{ \sum_{(w_i, c_i) \in D} \log \left( \sum_{s_{ij}} p(c_i|s_{ij}; \theta) p(s_{ij}|w_i; \theta) \right) - \gamma \sum_{ij-i'j'} \beta_r \|v_{ij} - v_{i'j'}\|^2 \right\} \quad (4)$$

# アルゴリズム

E ステップでは以下を求める.

$$s_{ij} = \arg \max_{s_{ij}} p \left( c_i \mid s_{ij}; \theta^{(t)} \right) \pi_{ij}^{(t)} \quad (5)$$

M ステップでは仮定  $w_i \sim \text{Multi}(w_i | \pi_i)$ ,  $\pi_i \sim \text{Dir} \left( \pi_i \mid \left\{ \lambda \pi_{ij}^{(0)} \right\}_j \right)$  の下で変分ベイズ法を使うことにより,  $\Pi$  を以下で更新すればよいことがわかる.

$$\pi_{ij}^{(t+1)} \propto \frac{\exp \left( \psi \left( \tilde{c}(w_i, s_{ij}) + \lambda \pi_{ij}^{(0)} \right) \right)}{\exp \psi \left( \tilde{c}(w_i) + \lambda \right)} \quad (6)$$

ここで  $\tilde{c}$  は期待される出現回数を表し,  $\psi$  はディガンマ関数を表している.



# アルゴリズム

$U$  と  $V$  については, negative sampling を使ったスキップグラムモデルを使って更新する. 具体的には以下の式に確率的勾配降下法を用いる.

$$\begin{aligned}\mathcal{L} = & \log \sigma(u_i \cdot v_{ij}) + \sum_{\substack{j' \\ j' \neq j}} (-u_i \cdot v_{ij'}) \\ & + \sum_m \mathbb{E}_{c'_i \sim P_n(c)} [\log \sigma(-u_{i'} \cdot v_{ij})]\end{aligned}\tag{7}$$

# アルゴリズム

よって以下のアルゴリズムを得る.

```
1: function SENSEEM( $D, \Omega$ )
2:    $\theta^{(0)} \leftarrow \text{initialize}$ 
3:   for  $(w_i, c_i) \in D$ 
4:     if period  $> k$ 
5:       RETROFIT( $\theta^{(t)}, \Omega$ )
6:       /* E-Step */
7:        $s_{ij} \leftarrow \text{update using equation 5}$ 
8:       /* M-Step */
9:        $\Pi^{(t+1)} \leftarrow \text{update using equation 6}$ 
10:       $U^{(t+1)}, V^{(t+1)} \leftarrow \text{update using 7}$ 
11:   return  $\theta^{(t)}$ 
```

オントロジーとしては WordNet を，ベクトル表現の作成には Word2Vec を使用する．

3 つのタスクで他の手法と比較実験する．

## 1. Similarity Scoring

2 つの単語に対し，あらかじめ人手で付与された類似度と，提案手法で得たベクトルのコサイン類似度を比較する．

## 2. Synonym Selection

与えられた語に対して，リスト中の最も意味が近い語を見つけてくるタスク．提案手法では 2 語の類似度を次で求める．

$$\text{maxSim}(w_i, w_{i'}) = \max_{j, j'} \cos(v_{ij}, v_{i'j'}) \quad (9)$$

# Similarity Scoring と Synonym Selection の結果

		Word Similarity ( $\rho$ )				Synonym Selection (%)		
		WS-353	RG-65	MC-30	MEN-3k	ESL-50	RD-300	TOEFL-80
GC	SINGLE	<b>0.623</b>	0.629	0.657	0.314	47.73	45.07	60.87
	MULTI	0.535	0.510	0.309	0.359	27.27	47.89	52.17
	<b>RETRO</b>	0.543	<b>0.661</b>	<b>0.714</b>	<b>0.528</b>	<b>63.64</b>	<b>66.20</b>	<b>71.01</b>
SG	SINGLE	<b>0.639</b>	0.546	0.627	<b>0.646</b>	52.08	55.66	66.67
	EM	0.194	0.278	0.167	0.228	27.08	33.96	40.00
	WSD	0.481	0.298	0.396	0.175	16.67	49.06	42.67
	IMS	0.549	0.579	0.606	0.591	41.67	53.77	66.67
	<b>RETRO</b>	0.552	0.673	0.705	0.560	56.25	65.09	<b>73.33</b>
	<b>EM+RETRO</b>	0.321	<b>0.734</b>	<b>0.758</b>	0.428	<b>62.22</b>	<b>66.67</b>	68.63

# Similarity Scoring in Context

## 3. Similarity Scoring in Context

文脈上の意味が考慮された Similarity Scoring. 例えば「土手」の意味の“bank”と「銀行」の意味の“bank”は区別される.

$$\begin{aligned} \text{avgSimC}(w_i, c_i, w_{i'}, c_{i'}) \\ = \sum_{j, j'} p(s_{ij} | c_i, w_i) p(s_{i'j'} | c_{i'}, w_{i'}) \cos(v_{ij}, v_{i'j'}) \end{aligned} \quad (10)$$

Vectors	SCWS ( $\rho$ )
SG-WSD	0.343
SG-IMS	0.528
SG-RETRO	0.417
GC-RETRO	0.420
SG-EM	0.613
SG-EM+RETRO	0.587
GC-MULTI	<b>0.657</b>

# Similarity Scoring in Context

3 つ目のタスクでは GC (Global Context), すなわち単語ごとにベクトルを作る方法のほうが提案手法よりも良い結果を出している. これは実験に使った SCWS データセットが異なる品詞の語の組を含んでいる一方で, WordNet が同一の品詞の語の組にしか類義語関係や上位語・下位語の関係を付与していないことに起因していると考えられる.

提案手法の 1 つ目 (RETRO) と 2 つ目 (EM+RETRO) の違いとしては, 前者が訓練にかかる時間が短く, また他のモデルと組み合わせやすいという特徴があるのに対し, 後者は上記のタスクでより高い精度を出しているという特徴がある.

Method	CPU Time
RETRO	~20 secs
EM+RETRO	~4 hours
IMS	~3 days
WSD	~1 year

# 定性的評価

意味を考慮しない場合は、最も頻度が高い語が選ばれやすいのに対し、意味を考慮した場合は、比較的頻度が少ないものも選ばれるようになっている。

Word or Sense	Top 3 Most Similar
hanging hanging (suspending) hanging (decoration)	hung dangled hangs shoring support suspension tapestry braid smock
climber climber (sportsman) climber (vine)	climbers skier Loretan lifter swinger sharpshooter woodbine brier kiwi

語義ごとのベクトル表現を用いることにより， Similarity Scoring in Context を除く WSD のタスクにおいて， 単語ごとにベクトルを割り当てる手法と比較して大幅な改善が見られた．

ここでは固定的なオントロジーに基づいてベクトルを作ったが， 将来の展望としてオントロジーとベクトル空間モデルの間で双方向的に拡大・改善していくような拡張が考えられる． 他には， 複数言語の WordNet を用いて， 多言語に一般化された意味ベクトルを学習するような拡張も考えられる．



Jauhar, S. K., Dyer, C., and Hovy, E. (2015). Ontologically grounded multi-sense representation learning for semantic vector space models.