文の意味構成に伴う高次元空間の最適化と単語表現学習

椿 真史，**Kevin Duh**，**新保 仁**，**松本 裕治**
奈良先端科学技術大学院大学
情報科学研究科　自然言語処理学講座　松本研究室
{masashi-t,kevinduh,shimbo,matsu}@is.naist.jp

1　はじめに

文が持つ意味を計算機上に表現することができれば，自然言語処理の研究において大きな進歩となる．例えば，情報検索から言語生成に至るまでの様々な応用を，文の意味を基準として考えることが可能となる．

これまでの自然言語処理研究においては，単語が持つ意味的な情報をベクトルによって表現する単語ベクトル空間モデル (Turney and Pantel, 2010) が広く用いられ，大きな成功を収めてきた．このモデルは，似た文脈に出現する単語は似た意味を持つという仮説である Distributional Semantic Model (以下 DSM) に基づいている．近年では，ニューラルネットワークによる学習モデルが新たに提案され，低次元かつ密で高効率な単語ベクトル表現が得られるようになった (Collobert and Weston, 2008).

また一方で Mitchell and Lapata (2008) 以降，単語ベクトルを用いて意味の構成性をモデル化する研究が盛んに行われている．この研究では，句や文の意味を適切に表現するベクトルを，それを構成する個々の単語ベクトルの合成演算によって得ることが目標となる．

本論文で我々は，単語から文を構成する際に生じる幾つかの問題に焦点を当て，文の意味に基づいた単語表現学習モデルと類似度計算手法を新たに提案する．

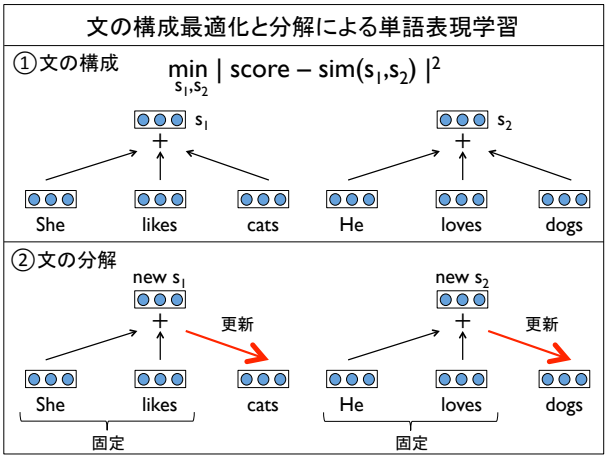

図 1: 構成と分解による最もシンプルな単語表現学習モデル．まず，個々の単語ベクトルの和として文ベクトルを構成する．次に，文ベクトルの類似度と学習データの類似度スコアとの二乗誤差を最小化することで，文ベクトルを学習する．最後に，最適化された文ベクトルを分解して，個々の単語ベクトルを更新する．

2　単語の意味から文の意味へ

2.1　句の構成モデル

これまで，句の意味ベクトルを構成するモデルが様々に提案されてきた．Baroni and Zamparelli (2010) や Socher ら (2012) は，形容詞や副詞をベクトルではなく行列によって表現することで，それらが持つ意味的な作用を単語ベクトル空間内の線形変換として捉えるモデルを提案した．また Grefenstette and Sadrzadeh (2011) や Tsubaki ら (2013) は，主語-動詞-目的語の構成性において，テンソル演算や射影行列によって動詞に対する文脈付けを行い，動詞の語義曖昧性解消に有効なモデルを提案した．しかし，これらはあくまで構成性の部分的なモデル化であり，最終的な目標は単語の意味から文の意味全体を構成するモデルを構築することである．

2.2　文の意味的類似度評価タスク (Semantic Textual Similarity)

SemEval2012 から始まった Semantic Textual Similarity (以下 STS) (Agirre et al., 2012) では，2 つの文の意味的な類似度を人手でスコアリングしたもの (表 1) がデータセットとして公開されている．現在の SemEval2014 では STS と同様の形式で，より大規模なデータセット[1] が公開されている．これらは，単語の意味から文の意味を構成するモデルを評価するための，最適なタスクであると言える．

しかし，ここで幾つかの問題が存在する．例えば，

[1]Evaluation of Compositional Distibutional Semantic Models on Full Sentences through Semantic Relatedness and Entailment http://alt.qcri.org/semeval2014/task1/

文 1	文 2	人手による類似度スコア
The man is playing the piano.	The man is playing the guitar.	1.6
A woman is playing the guitar.	A man is playing guitar.	2.4
A plane is taking off.	An air plane is taking off.	5.0

表 1: Semantic Textual Similarity (STS) のデータセットの一例．2 つの文が与えられた時の類似度スコア (0.0〜5.0) が人手で付けられたものとなっている．スコアが高いほど 2 つの文の意味的類似度が高い．

以下の 2 つ文の意味的類似度を考える．

- The man is playing the piano.
- The man is playing the guitar.

これら 2 つの文では，男性が弾いている楽器が”piano”と”guitar”で異なるため，人手による類似度スコアは低くなっている (表 1)．しかし，”piano”と”guitar”は DSM において同一クラスに属する単語となるため，ベクトルの類似度は常に高くなってしまう．他の例としては，”woman”と”man”や”big”と”small”などについても同様の問題が生じる．また文は主に,「誰が，何を，いつ，どこで，どうした」等の表現を構成するため，単語の意味とはまったく異なった，より多くの意味を持ち得る．このように，DSM に基づいた単語ベクトル空間では捉えられない意味が，文の意味を考える際には決定的に重要となる．

我々はこれらの問題を意識し，単語の意味から文の意味を構成する際の重要な点として，以下の 2 つを考える．

1. 似た文脈に出現する単語は似た意味を持つという DSM の観点ではなく，文に対する意味的寄与として個々の単語が意味を持つ (飯田隆, 1987) という観点から，ベクトル空間における単語表現を捉え直す．
2. 文の意味を表現する空間は，単語の意味を表現する空間とは異なり，またより多くの表現を構成するため高次元であるとした上で，その空間で文の意味的類似度を考える必要がある．

次章の提案手法で，これら 2 つの点を踏まえた新たなモデルを提案する．

3 提案手法

3.1 文ベクトルの構成最適化と分解による単語表現学習

この節では，最もシンプルな文ベクトルの構成最適化と分解による単語表現学習モデルについて概説する (図 1)．これは，3.2 節で提案する文の意味的類似度計算手法と合わせて 3.3 節で拡張される，最も基礎的なモデルとなる．

まず，単語 w のベクトルを $\mathbf{d}(w)$ とすると，文 S を表現するベクトル $\mathbf{s}$ を，それを構成する個々の単語ベクトルの総和 $\mathbf{s} = \sum_{w \in S} \mathbf{d}(w)$ とする．2 つの文 S_1，S_2 の意味的類似度は，コサイン類似度 $\cos(\mathbf{s_1}, \mathbf{s_2}) = \frac{\mathbf{s_1} \cdot \mathbf{s_2}}{|\mathbf{s_1}||\mathbf{s_2}|}$ を用いて計算する．

ここで，学習データの類似度スコアを $score$ とすると，コスト関数 J を

$$J(\mathbf{s_1}, \mathbf{s_2}) = \frac{1}{2}||score - \cos(\mathbf{s_1}, \mathbf{s_2})||^2 \quad (1)$$

として，以下の 2 ステップで単語ベクトルを学習する．

1. $\mathbf{s_1}$ と $\mathbf{s_2}$ を確率的急降下法 (SGD) によって最適化する．
2. 学習後の文ベクトルを $\mathbf{s}_{new}$，更新する単語ベクトルを $\mathbf{d}(w)$ とすると，以下の式で新たな単語ベクトル $\mathbf{d}(w)_{new}$ を得る．

$$\mathbf{d}(w)_{new} = \mathbf{s}_{new} - (\mathbf{s} - \mathbf{d}(w)) \quad (2)$$

つまり，構成後の文ベクトルを学習データから最適化し，それを構成とは逆の演算によって分解することで，どの単語がどの程度文の意味に寄与するのかを単語ベクトルが学習することになる (図 1)．

3.2 カーネル和による文の意味的類似度計算

3.1 節では単純に，文ベクトルを単語ベクトルの和として計算した．この時，文は単語と同じ空間で表現されることになる．しかしここで我々は，単語から文を構成する際には，文をより適切に表現する別の意味空間の構成が伴うと考える．また，文は単語よりも多くの意味を持ち得るとすると，文を表現する空間は高次元とした上で類似度を計算する必要があると考える．そこで新たに，カーネルを用いた文の意味的類似度計算手法を提案し，以下に具体的な例を通して概説する．

まずはじめに，文 S_1 を”She runs a big company.”，文 S_2 を”He runs a small company.”とすると，それぞれの文が持つ述語項の種類 (例 : (NN, VB, NN)) と，それに含まれる単語 (例 : {she, run, company}) は，表 2 のようになる (文 S が持つ，述語項の種類-それに含まれる単語を $P(S)$ で表す)．ここで，{she, run, company} を p とすると，これらの単語の合成ベクトル (以下，述語項ベクトルと呼ぶ) $\mathbf{p}$ を以下のように計算する．

$$\mathbf{p} = \sum_{w \in p} \mathbf{d}(w) \quad (3)$$

文 S	述語項の種類-それに含まれる単語 $P(S)$
She runs a big company.	(NN,VB,NN) : { she, run, company } (ADJ,NN) : { big, company } (SENTE) : { she, runs, a, big, company }
He runs a big company.	(NN,VB,NN) : { he, run, company } (ADJ,NN) : { small, company } (SENTE) : { he, runs, a, small, company }

表 2: 文 S と，述語項の種類-それに含まれる単語 $P(S)$ の例．SENTE は文全体を意味する．

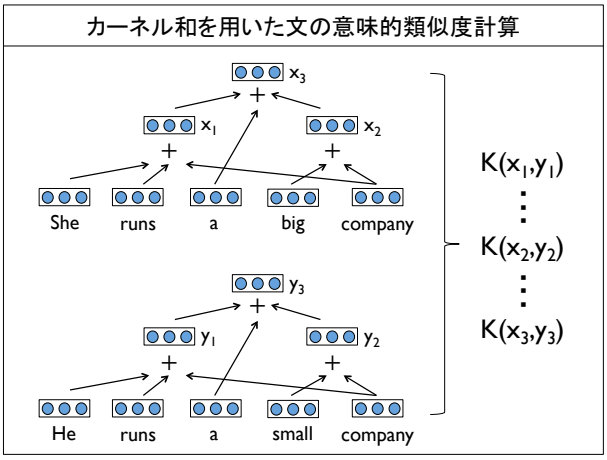

図 2: カーネル和を用いた文の意味的類似度計算手法．まず，各々の文の述語項ベクトル (単語ベクトルの和) を計算する．次にそれらのカーネルを計算し，最終的にその和を文の意味的類似度とする．

そして，2 つの文の類似度 $sim_K(S_1, S_2)$ を，2 つの文に現れる述語項ベクトルのカーネル K の総和とする．

$$sim_K(S_1, S_2) = \sum_{p_1 \in P(S_1)} \sum_{p_2 \in P(S_2)} K(\mathbf{p_1}, \mathbf{p_2}) \qquad (4)$$

つまり，2 つの文に含まれるの述語項ベクトルのカーネル和を計算することで，元の単語ベクトル空間とは異なる高次元空間での類似度を計算することになる (図 2) .

3.3 文の意味的類似度の最適化による単語表現学習

この節では，3.2 節の文の意味的類似度計算手法を用いて，3.1 節の単語表現学習モデルを新たに拡張する．

まず，(1) 式中のコサイン類似度 $\cos(\mathbf{s_1}, \mathbf{s_2})$ を，(4) 式のカーネル和を用いた文の意味的類似度 $sim_K(S_1, S_2)$ に置き換えて次式を得る．

$$J(S_1, S_2) = \frac{1}{2} ||score - sim_K(S_1, S_2)||^2 \qquad (5)$$

このコスト関数を最小化し，すべての述語項ベクトルについて (2) 式と同様の更新手順を用いて，単語ベクトルを学習する．つまり，3.1 節のように単語ベクトル空間において文ベクトルを最適化するのではなく，より高次元空間での文の意味的類似度を最適化する中で，個々の単語ベクトルを学習することになる．

4 評価実験

4.1 データセットと評価方法

SemEval2012 の STS データセット [2] の中から，今回は MSR video corpus のドメインのみを対象に実験を行う (このドメインに含まれる文は比較的短く，述語項構造解析の失敗が少ない) ．トレーニングデータとテストデータはそれぞれ，750 の文対と人手による類似度スコアから構成されており (表 1) ，述語項構造解析には Enju[3] を使用する．単語ベクトル (50 次元) はニューラルネットワークによって学習された SENNA[4] を初期値とし，文の意味的類似度計算には以下の 3 つのカーネルを用いて実験結果の違いを比較する．

$$K_{poly}(\mathbf{x}, \mathbf{y}, c) = (c + \cos(\mathbf{x}, \mathbf{y}))^2 \qquad (6)$$

$$K_{\tanh}(\mathbf{x}, \mathbf{y}, a, b) = \tanh(a \cos(\mathbf{x}, \mathbf{y})) + b) \qquad (7)$$

$$K_{gaus}(\mathbf{x}, \mathbf{y}, \sigma) = \exp\left(-\frac{1 - \cos(\mathbf{x}, \mathbf{y})}{\sigma^2}\right) \qquad (8)$$

多項式，シグモイド，ガウシアンカーネルにおける内積はすべてコサイン類似度に統一し，カーネル内のパラメータは述語項の種類別に学習する．単語ベクトルの学習率は 0.1，カーネル内のパラメータの学習率は 0.001 とする．最後に，学習された単語ベクトルを用いて計算した文の類似度と，データセットの類似度スコアとのピアソン相関係数 r で評価する．比較する既存研究のモデルについては，我々のモデルとの関連を踏まえ 5 章で概説する．

4.2 結果と考察

表 3 に，我々のモデルと幾つかの既存モデルの相関係数を示す．以下がその結果と考察である．

[2] http://www.cs.york.ac.uk/semeval-2012/task6/index.php?id=data

[3] http://www.nactem.ac.uk/enju/index.ja.html

[4] ronan.collobert.com/senna/

モデル	r (学習前)	r (学習後)
加算モデル	0.472	0.706
多項式カーネル	0.598	0.814
シグモイドカーネル	0.536	0.815
ガウシアンカーネル	0.596	**0.826**
Jimenez et al. (2012)	−	0.858
Bär et al. (2012)	−	0.873
Šarić et al. (2012)	−	0.880
Pilehvar et al. (2013)	−	**0.887**

表 3: 我々のモデルと幾つかの既存モデルの相関係数．我々のモデルの中では，ガウシアンカーネルが最も高い相関係数 (0.826) を示している．

1. カーネルを用いた我々のモデルはすべて，単純な加算モデルの相関係数を上回っている．これは，単語ベクトル空間とは異なる高次元空間において，文の意味的類似度を最適化するためと考えられる．

2. 我々のモデルの中では，ガウシアンカーネルを用いた場合に最も高い相関係数 (0.826) を示したが，既存のモデルと比較するとその性能は低い．これは，我々のモデルが主に単語ベクトルを学習するアプローチであり，テストデータのみに現れる単語が学習されないためだと考えらえる．

3. 単語ベクトルの学習は，デスクトップマシン (Intel Core i7 2.93Ghz CPU，8GB RAM) を用いて 10 分程度で完了する．本論文のモデルは単純なベクトルの和とその分解に基づいているため，学習が高速であると考えられる．

5 関連研究

STS に対する主なアプローチは，本論文とはまったく異なる．例えば，2 つの文に含まれる共通の単語や句の数，n グラムや木構造のアライメントなどの様々な素性を考え，サポートベクター回帰で学習するものがほとんどである (Jimenez et al., 2012; Bär et al., 2012; Šarić et al., 2012)．また Pilehvar ら (2013) は，WordNet の情報や語義曖昧性解消のアルゴリズムを用い，STS において最も良い性能を持つモデルを提案している．これらはある程度成功しているが，単語ベクトル空間と文の構成性に基づいた意味的なアプローチとはなっていない．我々の知る限りそのような研究はまだ少なく，本研究がその基礎的なモデルになると考えている．

6 結論と今後の課題

本論文で我々は，文の構成最適化と分解による単語表現学習モデルと，カーネル和を用いた文の意味的類似度計算手法を新たに提案した．今後の課題として，以下の 2 点を挙げる．

1. 本論文では，述語項ベクトルの計算に単純な単語ベクトルの和を用いたが，既存研究の様々な合成演算を組み合わせて詳細にモデル化する．

2. 文の持つ木構造や階層構造を考慮したカーネルを用いて文の意味的類似度を計算，最適化し，単語ベクトル表現を学習する．

特に 2 点目は，再帰的ニューラルネットワークや深層カーネル (Deep Kernel) の研究とも密接に関係しており，今後それらと比較し合わせて議論していきたい．

References

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *SemEval*.

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. In *SemEval*.

Marco Baroni and Roberto Zamparelli. 2010. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space. In *EMNLP*.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*.

Edward Grefenstette and Mehrnoosh Sadrzadeh. 2011. Experimental support for a categorical compositional distributional model of meaning. In *EMNLP*.

Sergio Jimenez, Claudia Becerra, and Alexander Gelbukh. 2012. Soft cardinality: A parameterized similarity function for text comparison. In *SemEval*.

Jeff Mitchell and Mirella Lapata. 2008. Vector-based models of semantic composition. In *ACl*.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *ACL*.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity. In *SemEval*.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *EMNLP-CoNLL*.

Masashi Tsubaki, Kevin Duh, Masashi Shimbo, and Yuji Matsumoto. 2013. Modeling and learning semantic co-compositionality through prototype projections and neural networks. In *ENNLP*.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *JAIR*, 37(1):141–188.

飯田隆. 1987. 言語哲学大全 論理と言語.