文献紹介:

Specializing Word Embeddings for Similarity or Relatedness

発表者: 長岡技術科学大学 竹野 峻輔

※ 発表資料中の図・式・表などは論文中のものから引用しています。

- Kiela, D., Hill, F., & Clark, S. (2015). Specializing Word Embeddings for Similarity or Relatedness. EMNLP 2015, pp.2044–2048.

論文概要

- “We demonstrate the advantage of special- izing semanticword embeddings for either similarity or relatedness. We compare two variants of retrofitting and a joint-learning approach, and find that all three yield specialized semantic spaces that capture human intuitions regarding similarity and relatedness better than unspecialized spaces. We also showthat using specialized spaces in NLP tasks and applications leads to clear improvements, for document classification and synonym selection, which rely on either similarity or relatedness but not both.” (本文概要)

- 分散表現では Relatedness(associative similarity)[e.g. dog, cat …] v.s. Similarity(genuine similarity) [e.g. dog, canine, …] を区別することができない.
  どちらが臨まれるべき性質かはタスクによって異なる.これらを区別できるように分散表現を改良することで, タスクに応じて性能が向上するようになる.

導入

- 分散表現(単語埋め込み) とは？

  単語をN次元上のベクトルに写像する手法. 単語と単語の関係を定める

  one-hot encoding ([0,…1,...,0]) :全ての単語間の距離が同じ,

  Skip-gram, **GloVe**(分布仮説に基づき教師なしで学習できる. トレンド)

    加法構成性を持つ i.e. v(king) - v(man) + v(woman) = v(queen)

- “似ている” にも種類がある. (dog, puppy, canine, …) v.s. (dog, cat, …)

  タスクに応じて, これらの類似度はむしろ区別できた方が良い

    文書のトピック分類 : 連想する類似度が良い (dog, cat …)

    機械翻訳 : 連想する類似度は翻訳に適さない (table ≠ chair)

導入

- “似ている” にも種類がある. (dog, puppy, canine, …) v.s. (dog, cat, …)
  タスクに応じて, これらの類似度は区別できた方が良い
    文書のトピック分類 : 関連度が高いものがあると良い (dog, cat …)
    機械翻訳 : 類似度は適すが, 関連度は適さない (table ≠ chair)

- 論文の目的 :
  1. 関連度(Relatedness)と類似度(Similarity), 各々特化する表現学習手法の検証
  2. 応用タスクにおいての, これらの表現の有効性の確認.

理論 - 関連度・類似度に特化した学習方法

- 従来の分散表現の学習に一工夫を加える:
  生テキスト + シソーラス(MyThes)[~類似度] or 連想辞書(USF free association norms)[~関連度])
  で学習する

$$\frac{1}{T}\sum_{t=1}^{T} J_\theta(w_t) = \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c} \log p(w_{t+j}|w_t)$$

- **Joint Learning Approach**
  目的関数に関連度・類似度の項を追加.

  Sampling: $\frac{1}{T}\sum_{t=1}^{T}\left(J_\theta(w_t) + [w^a \sim \mathcal{U}_{A_{w_t}}] \log p(w^a|w_t)\right)$

  All: $\frac{1}{T}\sum_{t=1}^{T}\left(J_\theta(w_t) + \sum_{w^a\in A_{w_t}} \log p(w^a|w_t)\right)$

- **Retrofitting Approach**
  コーパスを交互に学習:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c} \log p(w_{t+j}|w_t)$$

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{w^a\in A_{w_t}} \log p(w^a|w_t)$$

実験1: 関連度 v.s. 類似度 の学習の確認

- 実験1: 関連度 v.s. 類似度 の学習の確認

  SimLex (Hill et al. 2014b) - 999 ペアの単語セット → 類似度評価

  MEN (Bruni et al., 2014) - 3000個の比較セット → 関連度評価

- 実験2: 応用タスクにおける分散表現の性能の影響評価

  TOEFL synonym selection task (Landauer and Dumais, 1997) → 類似度評価

  Reuters Corpus Volume1(Lewis et al., 2004)の文書分類タスク →関連度評価

実験結果1

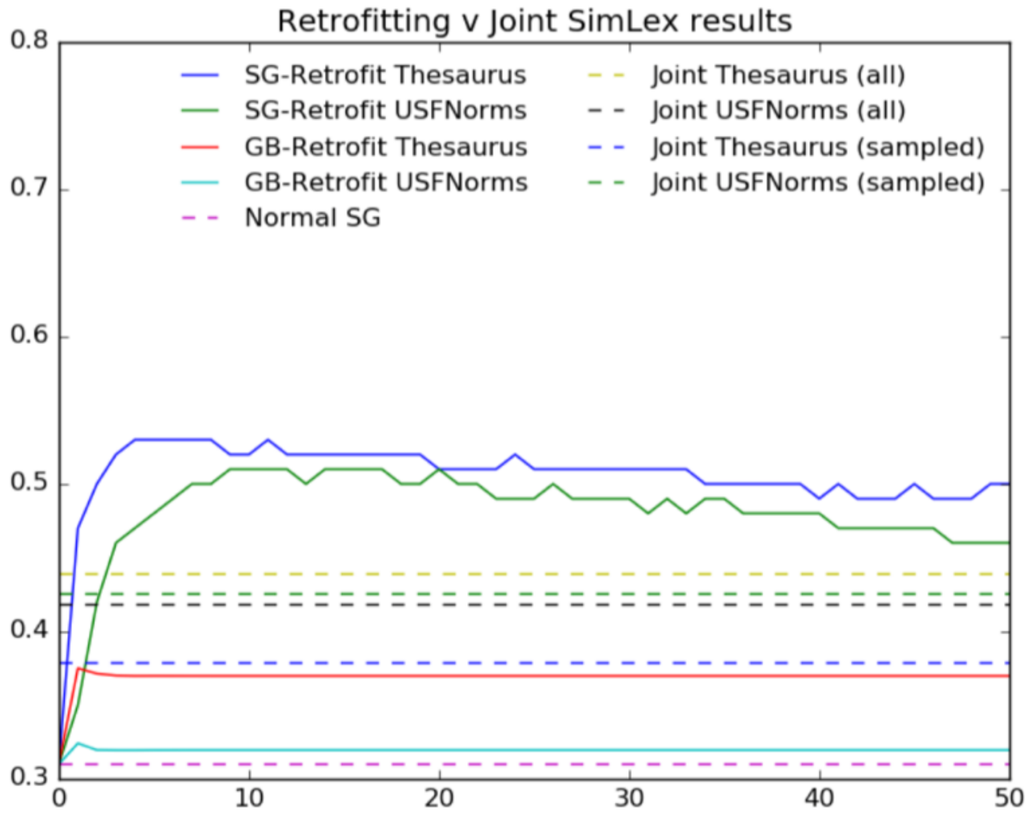
- SG(Skip-gram)より性能が向上
- 類似度タスクでは, 外部資源の挿入により性能が大幅に向上.
  関連度タスクでは 微増
- 類似度タスクでは
  SG - Retrofit - Thesaurusが有効
- 関連度タスクでは
  Joint-norms-sampledが有効

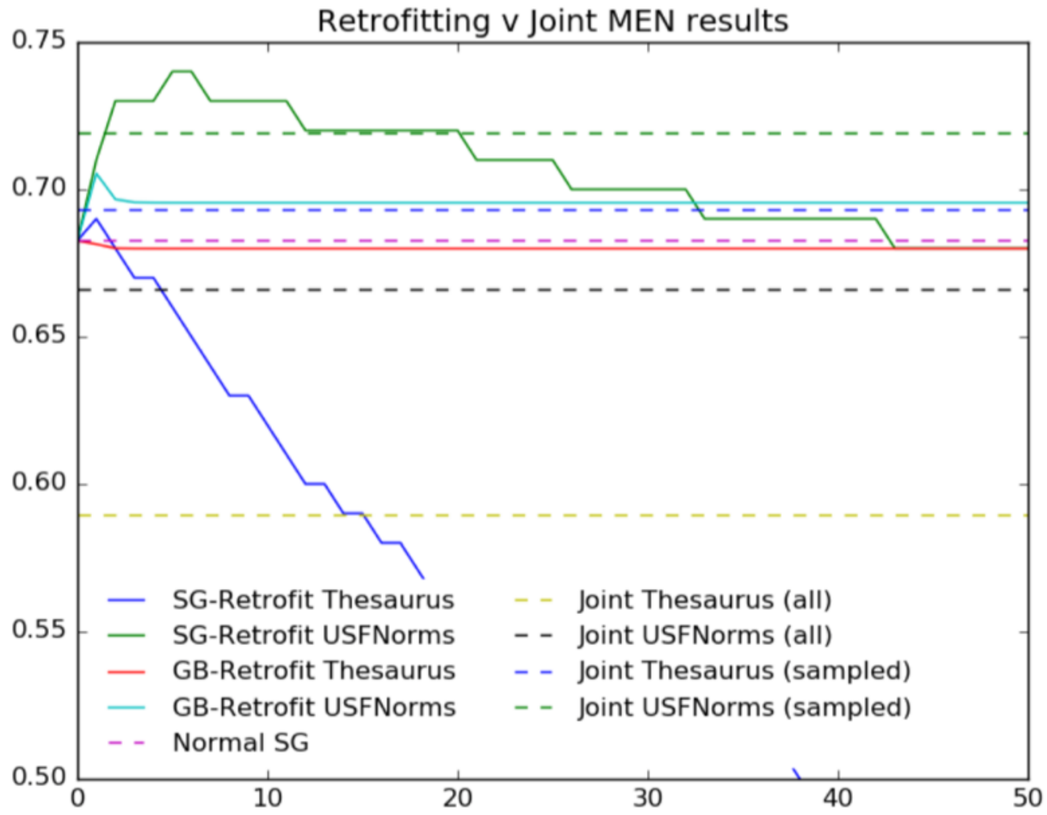
Method	SimLex-999	MEN
Skip-gram	0.31	0.68
Fit-Norms	0.08	0.14
Fit-Thesaurus	0.26	0.14
Joint-Norms-Sampled	0.43	**0.72**
Joint-Norms-All	0.42	0.67
Joint-Thesaurus-Sampled	0.38	0.69
Joint-Thesaurus-All	0.44	0.60
GB-Retrofit-Norms	0.32	0.71
GB-Retrofit-Thesaurus	0.38	0.68
SG-Retrofit-Norms	0.35	0.71
SG-Retrofit-Thesaurus	**0.47**	0.69

Table 1: Spearman ρ on a genuine similarity (SimLex-999) and relatedness (MEN) dataset.

実験結果1

- Retrofittingでは 外部資源を利用し, 分散表現を再最適化を繰り返すことで分散表現の関連度の特性を変更することができる.





実験結果2

- Joint-Norms-All(関連度特化)が
  文書分類のタスクで良い性能;
  SG-Retrofit-Thesaurus(類似度特化)が
  類義語判定タスクで良い性能.

  → 問題に応じて 関連度と類似度をわけることは有効.

Method	TOEFL		Doc
Skip-gram	77.50		83.96
Joint-Norms-Sampled	78.75		84.46
Joint-Norms-All	66.25		**84.82**
Joint-Thesaurus-Sampled	81.25		83.90
Joint-Thesaurus-All	80.00		83.56
GB-Retrofit-Norms	80.00		80.58
GB-Retrofit-Thesaurus	83.75		80.24
SG-Retrofit-Norms	80.00		84.56
SG-Retrofit-Thesaurus	**88.75**		84.55

Table 2: TOEFL synonym selection and document classification accuracy (percentage of correctly answered questions/correctly classified documents).

所感

- 提案手法は, 複数の語彙資源が統合できる手法.
  分野適応やpositive-negative といった性質の反映も簡単にできそう:

- 類義語判定タスクのような直接的なタスクでは効果は大きい しれないが
  類義語言語資源 v.s. 関連度言語資源 でみると性能差は微妙 :
    外部資源によるつかる情報が増えただけに見える

まとめ

- 教師無しで得られた分散表現を関連度、類似度に分けた最適化手法の提案

  Joint-learning model : 目的関数に言語資源の 項を加える(sampling/all)

  Refortting model : 生コーパス と 言語資源 で 交互に最適化

- 実験によりこれら手法の有効性を確認

  - 外部資源により関連度特化，類似度特化の分散表現の最適化が可能

  - 応用タスク(文書分類, 同義語判定)では, これらのベクトルが有効