Learning Word Vectors for Sentiment Analysis

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 142-150). Association for Computational Linguistics.

김미숙

Contents

- Introduction
- Our Model
- Experiments
- Discussion

1. Introduction

- Word representation
  - Vector-based model: 단어 사이의 유사도를 distance, angle로 표현하는 방법
    - Capture the rich relational structure of the lexicon(semantic…)
    - AI, Cognitive Science에 활발하게 사용되고 있음
    - 적용 분야: word sense disambiguation, named entity recognition(NER), part of speech tagging(POS tagging), document retrieval
- In this paper,
  - Capture both semantic and sentiment similarities among words
  - The vector representation of words to predict the sentiment annotations on contexts in which the words appear → words expressing similar sentiment to have similar vector representations
  - How the model can leverage document-level sentiment annotations
  - Data: Pang and Lee(2004) – sentiment and subjectivity corpora, Internet Movie Database(IMDB)

2. Our Model

- Capturing Semantic Similarities
  - *Assumption*: each word w_i is conditionally independent of the other words given θ
  - The probability of a document

$$p(d) = \int p(d, \theta) d\theta = \int p(\theta) \prod_{i=1}^{N} p(w_i|\theta) d\theta.$$

    - θ: multi-dimensional random variable
    - N: the number of words in d
  - Each word w in the vocabulary V has a β dimensional vector representation $\emptyset_w = Rw$
    - $R \in \mathbb{R}^{\beta \times |V|}$: word representation matrix
  - Energy:

$$E(w; \theta, \phi_w, b_w) = -\theta^T \phi_w - b_w.$$

    - b_w: each word to capture differences in overall word frequencies
  - $p(w|\theta)$, use a softmax

$$p(w|\theta; R, b) = \frac{\exp(-E(w; \theta, \phi_w, b_w))}{\sum_{w' \in V} \exp(-E(w'; \theta, \phi_{w'}, b_{w'}))} = \frac{\exp(\theta^T \phi_w + b_w)}{\sum_{w' \in V} \exp(\theta^T \phi_{w'} + b_{w'})}$$

2. Our Model

- Capturing Semantic Similarities
  - Derive maximum likelihood learning given a set of unlabeled documents

$$\max_{R,b} p(D; R, b) = \prod_{d_k \in D} \int p(\theta) \prod_{i=1}^{N_k} p(w_i|\theta; R, b) d\theta$$

  - Using maximum a posteriori(MAP) estimates for θ,

$$\max_{R,b} \prod_{d_k \in D} p(\hat{\theta}_k) \prod_{i=1}^{N_k} p(w_i|\hat{\theta}_k; R, b)$$

    - $\hat{\theta}_k$: MAP estimate of θ for d_k
  - By taking the logarithm and simplifying,

$$\max_{R,b} \quad \nu||R||_F^2 + \sum_{d_k \in D} \lambda||\hat{\theta}_k||_2^2 + \sum_{i=1}^{N_k} \log p(w_i|\hat{\theta}_k; R, b)$$

    - $\left|\left|R\right|\right|_F^2$: Frobenious norm regularization term for the word representation matrix R

2. Our Model

- Capturing Word Sentiment
  - Unlabeled data로 sentiment를 예측하는 명확한 방법이 없어서 labeled documents에 적용
  - Sentiment: Complex, multi-dimensional concept
  - Sentiment label s,

$$\hat{s} = f(\phi_w).$$

    - $f(x)$: an appropriate predictor function
    - $\emptyset_w$: a word vector
    - Improve our word vector $\emptyset_w$ to better predict the sentiment labels of contexts in which that word occurs
    - s: scalar continuous value representing sentiment polarity of a document $\in [0, 1]$
    - $f(x)$: logistic regression

2. Our Model

- Capturing Word Sentiment
  - Probability

$$p(s = 1|w; R, \psi) = \sigma(\psi^T \phi_w + b_c)$$

    - $\emptyset_w$: w's vector representation
    - ψ: regression weights
    - $\sigma(x)$: the logistic function

  - Log-objective,

$$\max_{R,\psi,b_c} \sum_{k=1}^{|D|} \sum_{i=1}^{N_k} \log p(s_k|w_i; R, \psi, b_c)$$

    - D: the set of labeled documents
    - s_k: sentiment label for document d_k

2. Our Model

- Learning
  - Full learning objective,

$$\nu||R||_F^2 + \sum_{k=1}^{|D|} \lambda||\hat{\theta}_k||_2^2 + \sum_{i=1}^{N_k} \log p(w_i|\hat{\theta}_k; R, b) \; + \sum_{k=1}^{|D|} \frac{1}{|S_k|} \sum_{i=1}^{N_k} \log p(s_k|w_i; R, \psi, b_c)$$

    - $|S_k|$: the number of documents in the dataset with the same rounded value of s_k
    - $\frac{1}{|S_k|}$: the weighting to combat the well-known imbalance in ratings
    - Weighting: prevents the overall distribution of document ratings from affecting the estimate of document rating in which a particular word occurs

  - Maximizing the objective function,
    - Non-convex problem, thus use alternating maximization
    - Step1. MAP estimates($\hat{\theta}_k$) fixed, optimizes the word representations (R, b, ψ, and b_c)
    - Step2. Find new MAP estimate for each document, (R, b, ψ, and b_c) fixed
    - Step3. continue this process until convergence
    - The optimization algorithm quickly finds a global solution for each $\hat{\theta}_k$ because we have a low dimensional, convex problems in each $\hat{\theta}_k$

3. Experiments

- Word Representation Learning
  - Data: 25,000 movie reviews form IMDB
    - At most 30 reviews from any movie
    - Dictionary: 5,000 most frequent tokens, but ignore 50 most frequent terms
    - Not use traditional stop word removal(e.g. negating words)
    - No Stemming: similar representations for words of the same stem
    - Use non-word token(e.g. “!” and “:-)”)
    - Star values($\in \{1,2,\dots,10\}$) → [0, 1]
  - Semantic component of our model
    - Does not require document labels, thus use 50,000 unlabeled reviews in addition to the labeled set of 25,000 reviews
    - For all word vector models, use 50-dimensional vectors
  - Assessment of word representations
    - A query word w and an other word w’
    - Obtain vector representations and evaluate their cosine similarity as

$$S(\phi_w, \phi_{w'}) = \frac{\phi_w^T \phi_{w'}}{||\phi_w|| \cdot ||\phi_{w'}||}$$

3. Experiments

- Word Representation Learning
  - Similarity of learned word vectors

	Our model Sentiment + Semantic	Our model Semantic only	LSA
melancholy	bittersweet	thoughtful	poetic
	heartbreaking	warmth	lyrical
	happiness	layer	poetry
	tenderness	gentle	profound
	compassionate	loneliness	vivid
ghastly	embarrassingly	predators	hideous
	trite	hideous	inept
	laughably	tube	severely
	atrocious	baffled	grotesque
	appalling	smack	unsuspecting
lackluster	lame	passable	uninspired
	laughable	unconvincing	flat
	unimaginative	amateurish	bland
	uninspired	clichéd	forgettable
	awful	insipid	mediocre
romantic	romance	romance	romance
	love	charming	screwball
	sweet	delightful	grant
	beautiful	sweet	comedies
	relationship	chemistry	comedy

- Both versions of our model better than LSA in avoiding accidental distributional similarity(e.g., *screwball* and *grant* as similar to *romantic*)
- Adding sentiment better
- However, problem of genre and content effects
- *Ghastly(무시/무시한)*: the sentiment enriched vectors, truly semantic alternatives to that word

3. Experiments

- Document Polarity Classification
  - Classifier must predict whether a given review is positive or negative given the review text
  - v: document's bag of words vector(tf.idf weight)
  - Matrix-vector product Rv → feature vector
  - In preliminary experiments, obtain 'bnn' weighting to work best for v, and use this weighting to get multi-word representation from word vectors

3. Experiments

- Document Polarity Classification
  - Pang and Lee Movie Review Dataset
    - 2,000 movie reviews with binary sentiment polarity label
    - Use Linear support vector machine classifier trained with LIBLINEAR and set the SVM regularization parameter to the same value used by Pang and Lee

Features	PL04	Our Dataset	Subjectivity
Bag of Words (bnc)	85.45	87.80	87.77
Bag of Words (bΔt'c)	85.80	88.23	85.65
LDA	66.70	67.42	66.65
LSA	84.55	83.96	82.82
Our Semantic Only	87.10	87.30	86.65
Our Full	84.65	87.44	86.19
Our Full, Additional Unlabeled	87.05	87.99	87.22
Our Semantic + Bag of Words (bnc)	88.30	88.28	88.58
Our Full + Bag of Words (bnc)	87.85	88.33	88.45
Our Full, Add'l Unlabeled + Bag of Words (bnc)	88.90	88.89	88.13
Bag of Words SVM (Pang and Lee, 2004)	87.15	N/A	90.00
Contextual Valence Shifters (Kennedy and Inkpen, 2006)	86.20	N/A	N/A
tf.Δidf Weighting (Martineau and Finin, 2009)	88.10	N/A	N/A
Appraisal Taxonomy (Whitelaw et al., 2005)	90.20	N/A	N/A

3. Experiments

- Document Polarity Classification
  - IMDB Review Dataset
    - 50,000 reviews from IMDB, no more than 30 reviews per movie(training 25,000 reviews)
    - Constructed dataset contains an even number of positive and negative reviews
    - Randomly guessing 50% accuracy
    - Use only highly polarized reviews

Features	PL04	Our Dataset	Subjectivity
Bag of Words (bnc)	85.45	87.80	87.77
Bag of Words (bΔt'c)	85.80	88.23	85.65
LDA	66.70	67.42	66.65
LSA	84.55	83.96	82.82
Our Semantic Only	87.10	87.30	86.65
Our Full	84.65	87.44	86.19
Our Full, Additional Unlabeled	87.05	87.99	87.22
Our Semantic + Bag of Words (bnc)	88.30	88.28	88.58
Our Full + Bag of Words (bnc)	87.85	88.33	88.45
Our Full, Add'l Unlabeled + Bag of Words (bnc)	88.90	88.89	88.13
Bag of Words SVM (Pang and Lee, 2004)	87.15	N/A	90.00
Contextual Valence Shifters (Kennedy and Inkpen, 2006)	86.20	N/A	N/A
tf.Δidf Weighting (Martineau and Finin, 2009)	88.10	N/A	N/A
Appraisal Taxonomy (Whitelaw et al., 2005)	90.20	N/A	N/A

    - Our model superior performance to other approaches

3. Experiments

- Subjectivity Detection
  - Performed sentence-level subjectivity classification
  - Decide whether a given sentence is subjective or objective
  - Subjective data: movie review summaries
  - Objective data: movie plot summaries

Features	PL04	Our Dataset	Subjectivity
Bag of Words (bnc)	85.45	87.80	87.77
Bag of Words (bΔt'c)	85.80	88.23	85.65
LDA	66.70	67.42	66.65
LSA	84.55	83.96	82.82
Our Semantic Only	87.10	87.30	86.65
Our Full	84.65	87.44	86.19
Our Full, Additional Unlabeled	87.05	87.99	87.22
Our Semantic + Bag of Words (bnc)	88.30	88.28	88.58
Our Full + Bag of Words (bnc)	87.85	88.33	88.45
Our Full, Add'l Unlabeled + Bag of Words (bnc)	88.90	88.89	88.13
Bag of Words SVM (Pang and Lee, 2004)	87.15	N/A	90.00
Contextual Valence Shifters (Kennedy and Inkpen, 2006)	86.20	N/A	N/A
tf.Δidf Weighting (Martineau and Finin, 2009)	88.10	N/A	N/A
Appraisal Taxonomy (Whitelaw et al., 2005)	90.20	N/A	N/A

  - Our model superior compared against other VSMs

4. Discussion

- Vector space model that learns word representations **capturing semantic and sentiment information**
- Our model is parametrized as **a log-bilinear model** following recent success in using similar techniques for language models(e.g., Bengio)
- We parametrize the topical component of our model in a manner that **aims to capture word representations** instead of latent topics
- Our method performed better than LDA
- Unsupervised model leverage the abundance of sentiment-labeled texts available online to yield word representations that capture both sentiment and semantic relations
- Existing datasets as well as a larger one
- These tasks involve relatively simple sentiment information, thus is broadly applicable in the growing areas of sentiment analysis and retrieval

Reference

- B. Pang and L. Lee. 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the ACL, pages 271–278.
- B. Pang and L. Lee. 2005. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of ACL, pages 115– 124.
- B. Pang, L. Lee, and S. Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In Proceedings of EMNLP, pages 79–86.
- C. Lin and Y. He. 2009. Joint sentiment/topic model for sentiment analysis. In Proceeding of the 18th ACM Conference on Information and Knowledge Management, pages 375–384.
- P. D. Turney and P. Pantel. 2010. From frequency to meaning: vector space models of semantics. Journal of Artificial Intelligence Research, 37:141–188.
- H. Wallach, D. Mimno, and A. McCallum. 2009. Rethinking LDA: why priors matter. In Proceedings of NIPS, pages 1973–1981.
- C. Whitelaw, N. Garg, and S. Argamon. 2005. Using appraisal groups for sentiment analysis. In Proceedings of CIKM, pages 625–631.