Document context language models(2016), Document embedding with Paragraph vectors(2015)

Y.C, Choi

2018.04.26

1. Document context language models

RNNLM

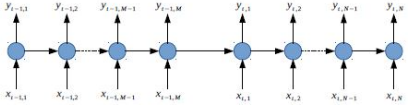


Figure 1: A fragment of document-level recurrent neural network language model (DRNNLM). It is also an extension of sentence-level RNNLM to the document level by ignoring sentence boundaries.

- Extension of sentence-level RNNLM.
- ignore sentence boundaries.

Two problems of document-level RNNLM

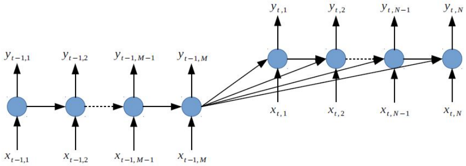
- Information decay
  - Meaningful document-level information is unlikely to survive
- learning
  - Since document-level RNNLM ignores sentence boundary, there are too many steps.

Models

The author of this paper suggested 3 models.

- Context-To-Context DCLM (ccDCLM)
- Context-To-Output DCLM (coDCLM)
- Attentional DCLM

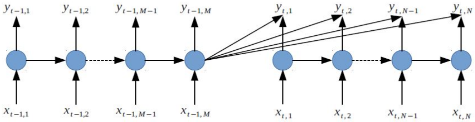
ccDCLM



(a) CCDCLM

- $c_{t-1} = h_{t-1,M}$ where M : the number of words in t-1 th sentence.
- $h_{t,n} = g_\theta(h_{t,n-1}, s(x_{t,n}, c_{t-1}))$

coDCLM



(b) coDCLM

- $h_{t,n} = g_\theta(h_{t,n-1}, x_{t,n})$
- $y_{t,n}$: $softmax(\mathbf{W_h h_{t,n}} + \mathbf{W_c c_{t-1}} + \mathbf{b})$

Difference between ccDCLM and coDCLM

1. The number of parameters
   (H : dim of hidden vector, K : dim of word representation,
   V : vocabulary size)
   - ccDCLM : H(16H+3K+6) + V(H+K+1)
   - coDCLM : H(13H+3K+6) + V(2H+K+1)
   - The difference of the parameter numbers is $VH - 3H^2$
   - In general, $V \gg H$
2. Computational advantage
   - In coDCLM, hidden vectors h_t and $h_{t'}$ are decoupled.

Attensional DCLM

- $c_{t-1,n} = \sum_{m=1}^{M} \alpha_{n,m} h_{t-1,m}$
- $\alpha_n = softmax(a_n)$
- $a_{n,m} = w_a^T tanh(\mathbf{W_{a1} h_{t,n}} + \mathbf{W_{a2} h_{t-1,m}})$
- $h_{t,n} = g_\theta(h_{t,n-1}, [c_{t-1,n}^T, x_{t,n}^T]^T)$
- $y_{t,n}\ softmax(\mathbf{W_o tanh(W_h h_{t,n}} + \mathbf{W_c c_{t-1,n}} + \mathbf{b}))$

2. Document Embedding with Paragraph Vectors

Paragraph Vectors

- The model inserts a memory vector to the standard language model
- To capturing the topics of the document.
- Two type of VP : The distributed memory model, The distributed bag of word model

Structure of the models

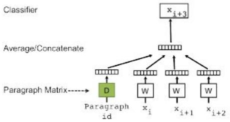


Figure 1: The distributed memory model of Paragraph Vector for an input sentence.

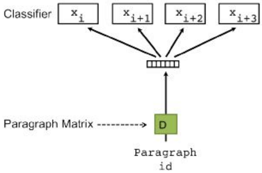


Figure 2: The distributed bag of words model of Paragraph Vector.

Accuracy measure

- For the quantitative evaluation, the author of this paper suggesed triplet measure.
- Given a article a_i, Construct a triplet$(a_i, b(i), c(i)$ i=1,...,n
- where $b(i)$ are closed to a_i but $c(i)$ is unrelated.
- After learning Paragraph vector model, check distance $d(a_i, b(i)), d(a_1, c(i))$
- accuracy $= \frac{\sum_{i=1}^{n}(I(d(a_i,b(i))>d(a_i,c(i))}{n}$

P.V using Wikipeda data

- use the distributed memory model of Paragraph Vector
- compare with LDA($\alpha = 0.1, \beta : between 0.01 and 1e - 6$)
- 4,4990,000 articles, 915,715 words.

Result of the model using wikipedia data

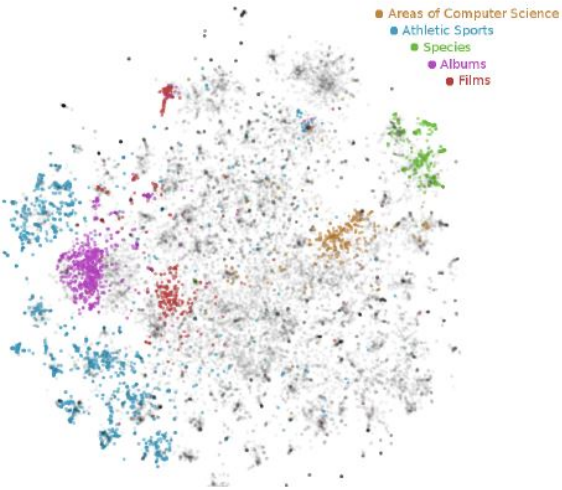


Figure 3: Visualization of Wikipedia paragraph vectors using t-SNE.

Result of the model using wikipedia data

Table 1: Nearest neighbours to "Machine learning." Bold face texts are articles we found unrelated to "Machine learning." We use Hellinger distance for LDA and cosine distance for Paragraph Vectors as they work the best for each model.

LDA	Paragraph Vectors
Artificial neural network	Artificial neural network
Predictive analytics	Types of artificial neural networks
Structured prediction	Unsupervised learning
Mathematical geophysics	Feature learning
Supervised learning	Predictive analytics
Constrained conditional model	Pattern recognition
Sensitivity analysis	Statistical classification
SXML	Structured prediction
Feature scaling	Training set
Boosting (machine learning)	Meta learning (computer science)
Prior probability	Kernel method
Curse of dimensionality	Supervised learning
Scientific evidence	Generalization error
Online machine learning	Overfitting
N-gram	Multi-task learning
Cluster analysis	Generative model
Dimensionality reduction	Computational learning theory
Functional decomposition	Inductive bias
Bayesian network	Semi-supervised learning

Result of the model using wikipedia data

Table 2: Wikipedia nearest neighbours

(a) Wikipedia nearest neighbours to "Lady Gaga" using Paragraph Vectors. All articles are relevant.

Article	Cosine Similarity
Christina Aguilera	0.674
Beyonce	0.645
Madonna (entertainer)	0.643
Artpop	0.640
Britney Spears	0.640
Cyndi Lauper	0.632
Rihanna	0.631
Pink (singer)	0.628
Born This Way	0.627
The Monster Ball Tour	0.620

(b) Wikipedia nearest neighbours to "Lady Gaga" - "American" + "Japanese" using Paragraph Vectors. Note that Ayumi Hamasaki is one of the most famous singers, and one of the best selling artists in Japan. She also has an album called "Poker Face" in 1998.

Article	Cosine Similarity
Ayumi Hamasaki	0.539
Shoko Nakagawa	0.531
Izumi Sakai	0.512
Urbangarde	0.505
Ringo Sheena	0.503
Toshiaki Kasuga	0.492
Chihiro Onitsuka	0.487
Namie Amuro	0.485
Yakuza (video game)	0.485
Nozomi Sasaki (model)	0.485

Result of the model using wikipedia data

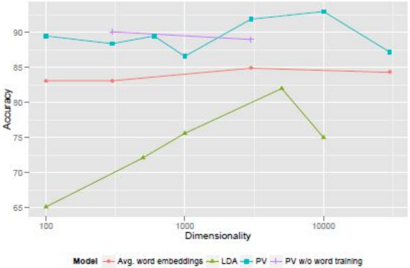


Figure 4: Results of experiments on the hand-built Wikipedia triplet dataset.

Table 3: Performances of different methods on hand-built triplets of Wikipedia articles on the best performing dimensionality.

Model	Embedding dimensions/topics	Accuracy
Paragraph vectors	10000	93.0%
LDA	5000	82%
Averaged word embeddings	3000	84.9%
Bag of words		86.0%

Result of the model using arXiv data

Table 7: arXiv nearest neighbours to "Distributed Representations of Sentences and Documents" - "neural" + "Bayesian". I.e., the Bayesian equivalence of the Paragraph Vector paper.

Title	Cosine Similarity
Content Modeling Using Latent Permutations	0.629
SimLex-999: Evaluating Semantic Models With (Genuine) Similarity Estimation	0.611
Probabilistic Topic and Syntax Modeling with Part-of-Speech LDA	0.579
Evaluating Neural Word Representations in Tensor-Based Compositional Settings	0.572
Syntactic Topic Models	0.548
Training Restricted Boltzmann Machines on Word Observations	0.548
Discrete Component Analysis	0.547
Resolving Lexical Ambiguity in Tensor Regression Models of Meaning	0.546
Measuring political sentiment on Twitter: factor-optimal design for multinomial inverse regression	0.544
Scalable Probabilistic Entity-Topic Modeling	0.541

Result of the model using arXiv data

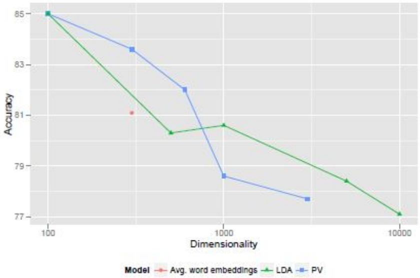


Figure 6: Results of experiments on the arXiv triplet dataset.

Table 8: Performances of different methods at the best dimensionality on the arXiv article triplets.

Model	Embedding dimensions/topics	Accuracy
Paragraph vectors	100	85.0%
LDA	100	85.0%
Averaged word embeddings	300	81.1%
Bag of words		80.4%