Linguistic Regularities in Continuous Space Word Representations

Tomas Mikolov, Wen-tau Yih, Geoffrey Zweig

Microsoft Research, Redmond

NAACL 2013

Abstract

- **Neural network language model**
  and **distributed representation** for words (Vector representation)
- Capture syntactic and remantic regularities in language
- Outperform state-of-the-art

Word Representation

- **One-hot Representation:**
  Example:
  "nervous" : [0 1 0 0 0 0 0 0 0 0 0 0...]
  "tense " : [0 0 0 0 0 1 0 0 0 0 0 0...]

- **Distributed Representation:**
  Example:
  "nervous" : [0.792,-0.177,-0.107,0.109,...]

  How to learn it?
  - Neural Network Language Model
  - Yoshua Bengio, 2003, **A neural probabilistic language model**

Language Model

- **The essence of language model**
  - input: $w_1, w_2, ..., w_{t-1}$
  - output: $P(w_t | w_1, w_2, ..., w_{t-1})$

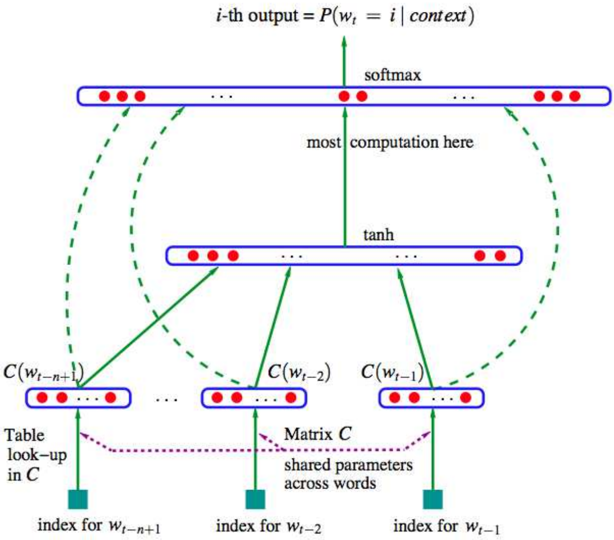
- **Traditional language model: N-gram**
  - Let $p(w_t | w_{t-1}, w_{t-2}, ..., w_{t-n+1}) = P(w_t | w_1, w_2, ..., w_{t-1})$
  - Weakness: curse of dimensionality
  - Weakness: not considering the "similarity" between words

  Example:
  "*The cat is walking in the bedroom*" in the trainning corpus contributes little to
  "*The dog was running in a room*"

Neural Network Language Model

- Associate with each word in the vocabulary a distributed *word feature vector*(a real-valued vector in $\mathbb{R}^m$)
  $C(\text{"}nervous\text{"}) = [0.792, -0.177, -0.107, 0.109, ...]$
- Express the joint *probability function* of word sequences in terms of the feature vectors
  $p(w_t|w_{t-1}, ..., w_{t-n+1}) = f(C(w_t), ..., C(w_{t-n+1}))$
- Learn simultaneously the *word feature vectors* and the parameters of that *probability function*

Neural Network Language Model



Neural Network Language Model

- **Layer1:(Input Layer)**
  - Input vector x: $m(n-1)$
  - Matrix C: $|V| \times m$
- **Layer2:(Hidden Layer)**
  - Hidden vector $\vec{h} = d + Hx$
  - Matrix H: $h \times m(n-1)$
  - Vector d: h
- **Layer3:(Output Layer)**
  - Output vector y: $|V|$
  - $y = b + Wx + Utanh(d + Hx)$
- **Softmax output layer:**
  - $P(w_t|w_{t-1}, ..., w_{t-n+1}) = \frac{exp(y_{w_t})}{\sum_i exp(y_i)}$

Neural Network Language Model Extention

- **CW**
  Collobert & Weston, 2011, JMLR
  Natural Language Processing(Almost) from Scratch
- **HLBL**
  Mnih & Hinton, 2007, ICML
  Three new graphical models for statistical language model
- **RNN**
  Mikolov(RNN), 2012, PhD thesis
  Statistical Language Models based on Neural Networks

Measuring Linguistic Regularity

- **Syntactic test**
  - *"a is to b as c is to ?"*
  - base/comparative/superlative forms of adjectives
  - singular/plural forms of common nouns

  ...

Category	Relation	Patterns Tested	# Questions	Example
Adjectives	Base/Comparative	JJ/JJR, JJR/JJ	1000	good:better rough:___
Adjectives	Base/Superlative	JJ/JJS, JJS/JJ	1000	good:best rough:___
Adjectives	Comparative/ Superlative	JJS/JJR, JJR/JJS	1000	better:best rougher:___
Nouns	Singular/Plural	NN/NNS, NNS/NN	1000	year:years law:___
Nouns	Non-possessive/ Possessive	NN/NN_POS, NN_POS/NN	1000	city:city's bank:___
Verbs	Base/Past	VB/VBD, VBD/VB	1000	see:saw return:___
Verbs	Base/3rd Person Singular Present	VB/VBZ, VBZ/VB	1000	see:sees return:___
Verbs	Past/3rd Person Singular Present	VBD/VBZ, VBZ/VBD	1000	saw:sees returned:___

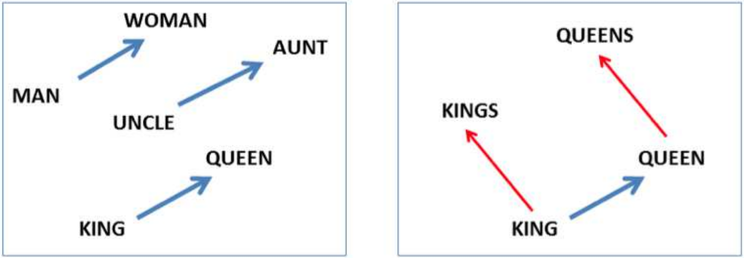
Measuring Linguistic Regularity

- **Semantic test**
  - SemEval-2012 Task 2, *Measuring Relation Similarity*
  - Given a group of word pairs, order the target pairs according to the degree to which this relation holds.

Subcategory	Relation name	Relation schema	Paradigms	Responses
8(e)	AGENT:GOAL	"Y is the goal of X"	pilgrim:shrine assassin:death climber:peak	patient:health runner:finish astronaut:space
5(e)	OBJECT:TYPICAL ACTION	"an X will typically Y"	glass:break soldier:fight juggernaut:crush	ice:melt lion:roar knife:stab
4(h)	DEFECTIVE	"an X is is a defect in Y"	fallacy:logic astigmatism:sight limp:walk	pimple:skin ignorance:learning tumor:body

Vector Offset Method

- To answer question $a : b \ c : d$ where d is unknown
  compute $y = x_b - x_a + x_c$
  $\omega^* = argmax_\omega \frac{x_\omega y}{\|x_\omega\|\|y\|}$
- "King - Man + Woman = Queue"



Experimental Results

- **Syntactic regularities identifying test**

Method	Adjectives	Nouns	Verbs	All
LSA-80	9.2	11.1	17.4	12.8
LSA-320	11.3	18.1	20.7	16.5
LSA-640	9.6	10.1	13.8	11.3
RNN-80	9.3	5.2	30.4	16.2
RNN-320	18.2	19.0	45.0	28.5
RNN-640	21.0	25.2	54.8	34.7
RNN-1600	**23.9**	**29.2**	**62.2**	**39.6**

Experimental Results

- **Syntactic regularities identifying test**
  Compared with other Neural Network Language Model

Method	Adjectives	Nouns	Verbs	All
RNN-80	**10.1**	8.1	**30.4**	**19.0**
CW-50	1.1	2.4	8.1	4.5
CW-100	1.3	4.1	8.6	5.0
HLBL-50	4.4	5.4	23.1	13.0
HLBL-100	7.6	**13.2**	30.2	18.7

Experimental Results

- **Semantic similarity test**

Method	Spearman's ρ	MaxDiff Acc.
LSA-640	0.149	0.364
RNN-80	0.211	0.389
RNN-320	0.259	0.408
RNN-640	0.270	0.416
RNN-1600	**0.275**	**0.418**
CW-50	0.159	0.363
CW-100	0.154	0.363
HLBL-50	0.149	0.363
HLBL-100	0.146	0.362
UTD-NB	0.230	0.395