Efficient Estimation of Word Representations in Vector Space

Google Inc., Mountain View, CA

Tomas Mikolov Kai Chen Greg Corrado Jeffrey Dean

Junmyeong Lee,
Junior student of GIST College

Raised problem

How to represent information well?

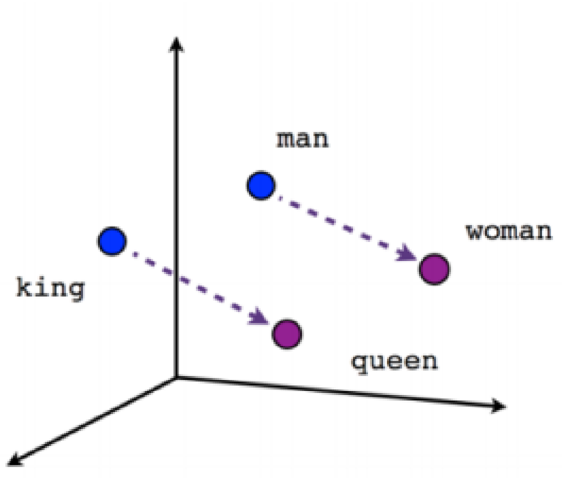
↓

How to encoding various words/sentences/documents?

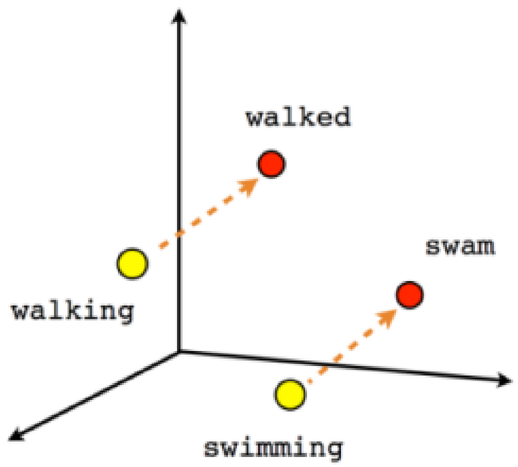
↓

Word embedding

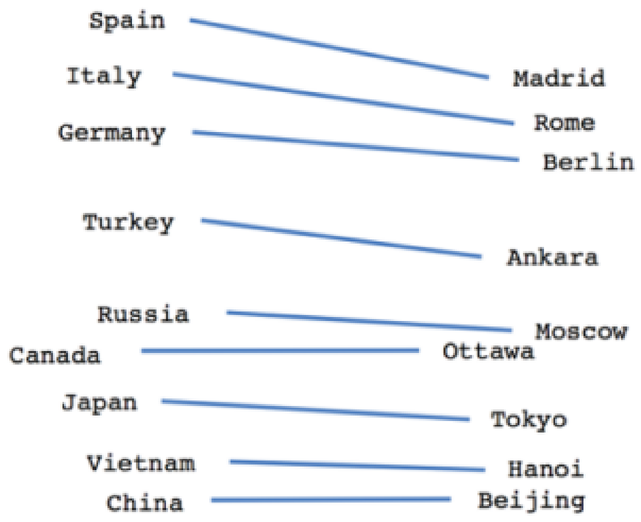
Word Embedding



Male-Female



Verb tense



Country-Capital

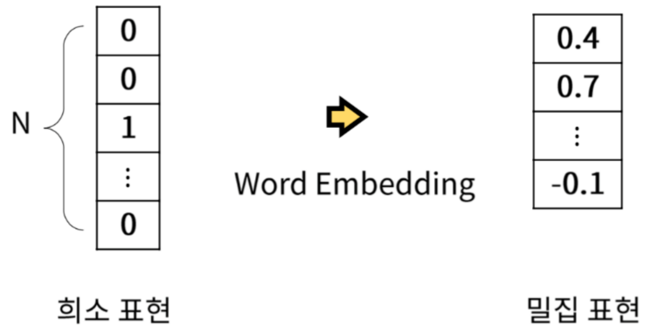
- Convert word to high-dimensional vector
- Represent one-hot vector to dense vector

Word Embedding

One-hot encoding
(Sparse representation)

Dense vector
(Dense representation)

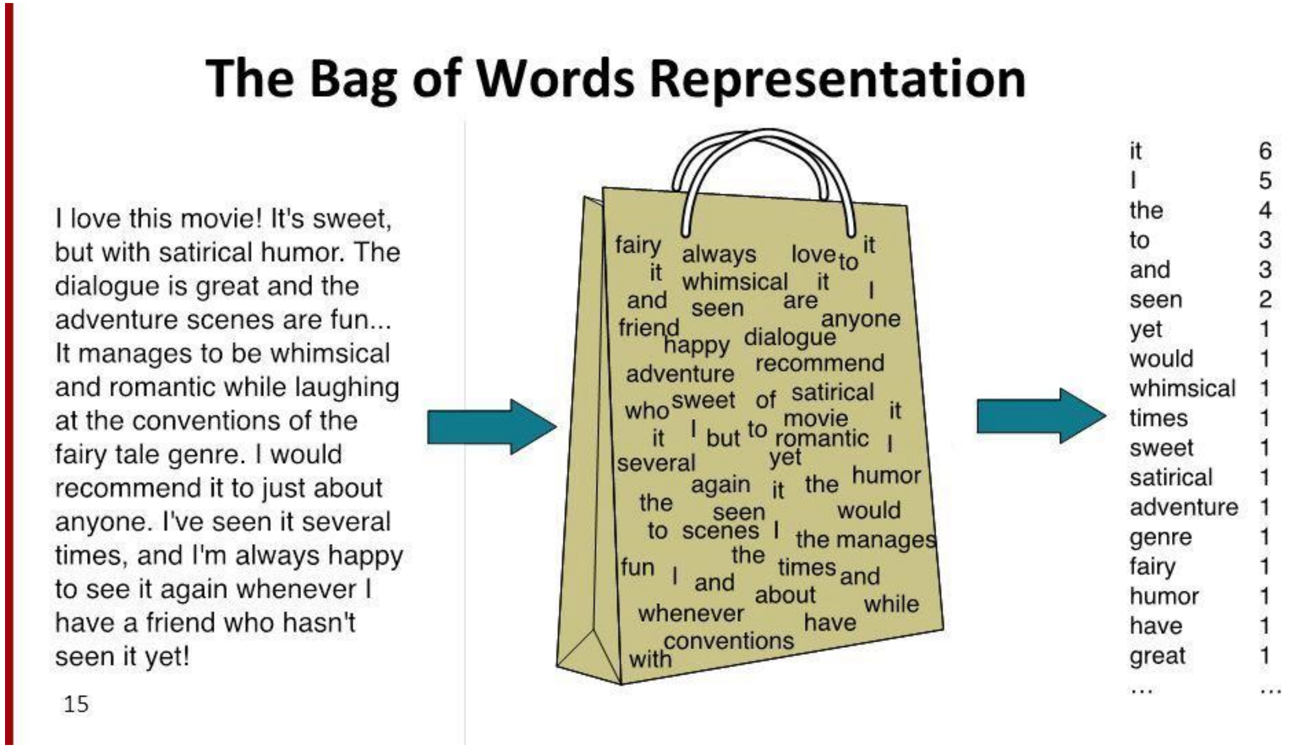
밀집 표현 Dense Representation



희소 표현된 단어를 임의의 길이의 실수 벡터로 표현할 경우, 이를 밀집 표현(Dense Representation)이라고 한다.
이 과정을 Word Embedding이라고 하며, 밀집 표현된 결과를 임베딩 벡터(Embedding Vector)라고 부른다.

Statistical Language model: Bag-of-Words model

- Procedure
  1. Assign index to each word
  2. Count # of words in document
  3. Make frequency table(or histogram)
- This model shows strong performance!
- But, we can’t consider relations between words in same sentence(or document)
- Applied to document classification/measuring similiarity



Statistical Language model: Bag-of-Words model

Overfitting

- About unseen word, model can’t represent word/document
- For sparse data, model can’t represent word well

Poor generalization performance

	가지	감자	고구마	당근	무	미역	양파	피망
문서0	12	10	3	8	6	3	4	12
문서1	13	1	4	10	1	6	3	1
문서2	1	4	8	8	13	4	2	12
문서3	3	15	9	11	11	3	11	2
문서4	10	11	7	14	5	12	0	8
문서5	1	2	1	15	3	3	9	3
문서6	15	10	12	11	5	2	3	10
문서7	7	8	13	7	9	6	13	3
문서8	2	12	10	10	0	1	5	8
문서9	14	14	0	5	11	6	0	3

Document-Term matrix

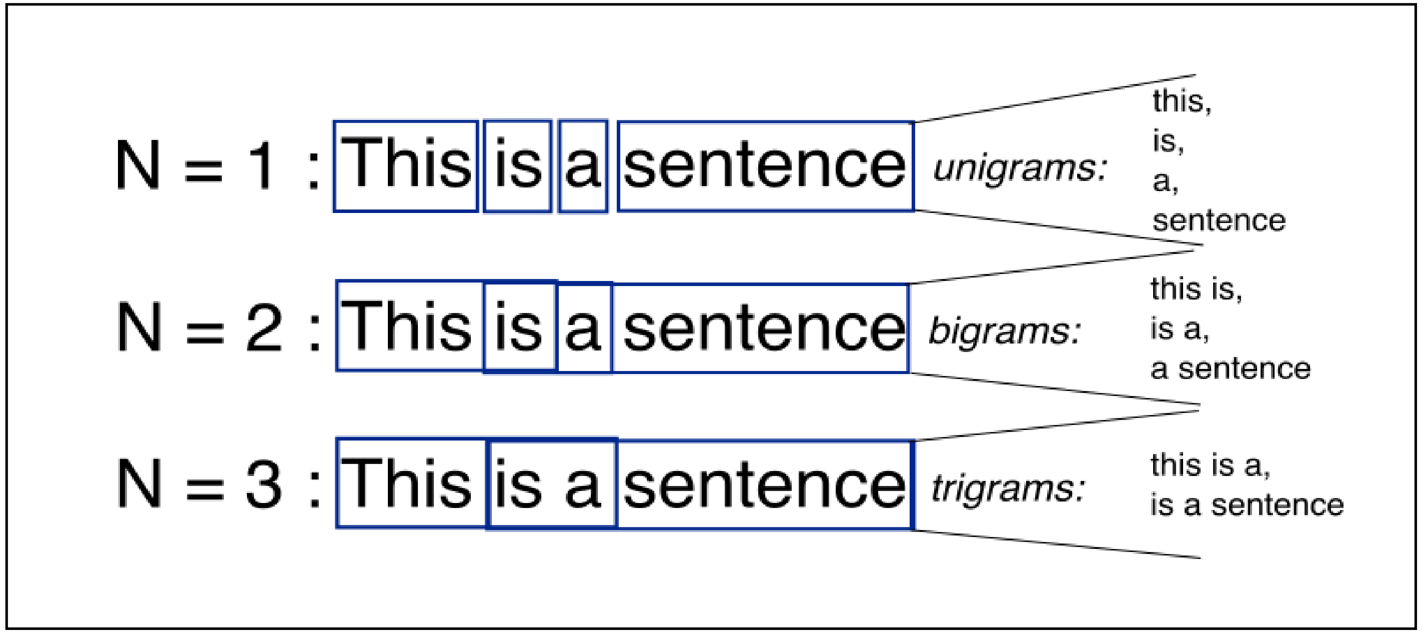
$$P(\text{is}|\text{An adorable little boy}) = \frac{\text{count}(\text{An adorable little boy is})}{\text{count}(\text{An adorable little boy })}$$

Statistical Language model: N-gram model

- Consider neighbor N-words (token)
- Richer representation than BoW model
- Richer representation

<Tokens>

- N=1 : unigram
- N=2 : bigram
- N=3 : trigram → N-gram



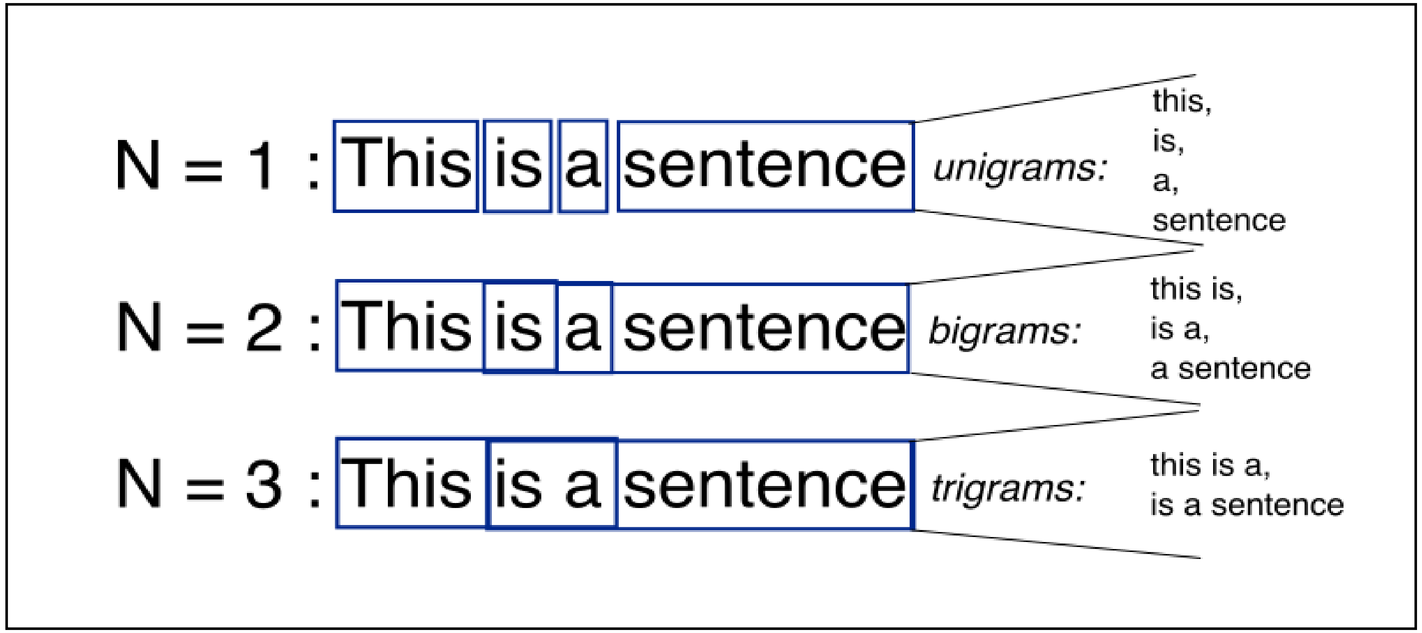
Statistical Language model: N-gram model

Overfitting

- Poor generalization performance

Tradeoff about N

- Sparsity problem
- Performance-complexity tradeoff



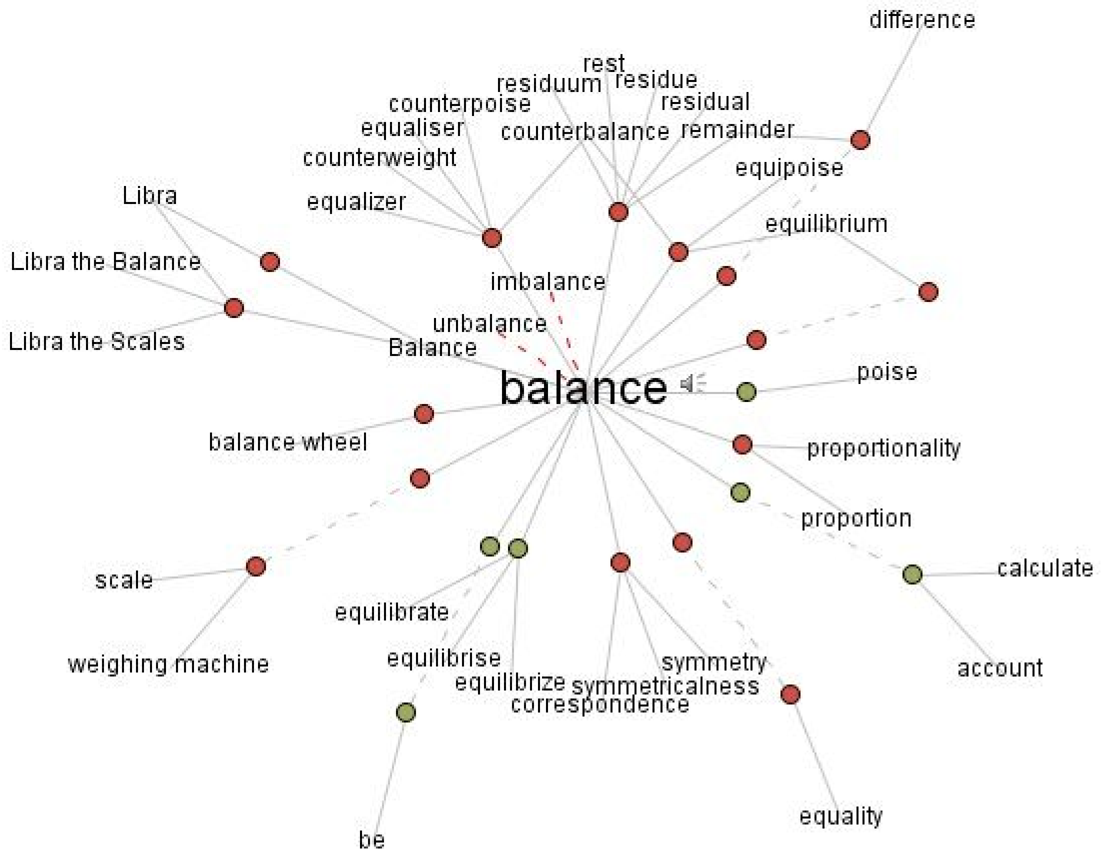
Neural Network based Language model

Distributed Hypothesis(분산 가설)

- We can think word vectors on the similar region have similar meanings
- NN based language model adopt distribution hypothesis as inductive bias

Distributed Representation(분산 표현)

- Dense vector representation of word
- Under "Distribution hypothesis"



Neural Network based Language model : NNLM series

NNLM(Neural Network Language Model)

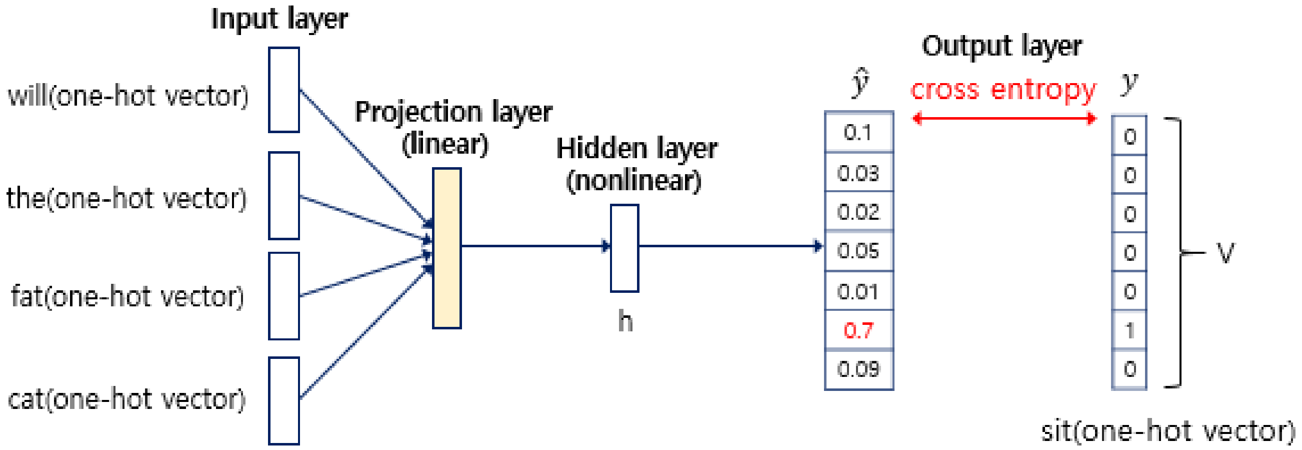
1. Projection Layer
2. Hidden Layer
3. Output layer

- Projection Layer

  - “Projection” each words to vector
  - No activation function

- Embedding Vector

  - Embedding Vector is row of projection matrix
  - By applying inner product, lookup one row of the projection matrix, which represents correspond word



$$x_{fat} \times W_{V\times M} = e_{fat}$$

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0.5 & 2.1 & 1.9 & 1.5 & 0.8 \\ 0.8 & 1.2 & 2.8 & 1.8 & 2.1 \\ 0.1 & 0.8 & 1.2 & 0.9 & 0.7 \\ 2.1 & 1.8 & 1.5 & 1.7 & 2.7 \\ & & & & \\ & & & & \\ & & & & \end{bmatrix} = \begin{bmatrix} 2.1 & 1.8 & 1.5 & 1.7 & 2.7 \end{bmatrix}$$

lookup table

Neural Network based Language model : Word2Vec

CBOW(Continuous BoW)

- Predict center word by using neighbor words

Skip-gram

- Predict neighbors by using center word

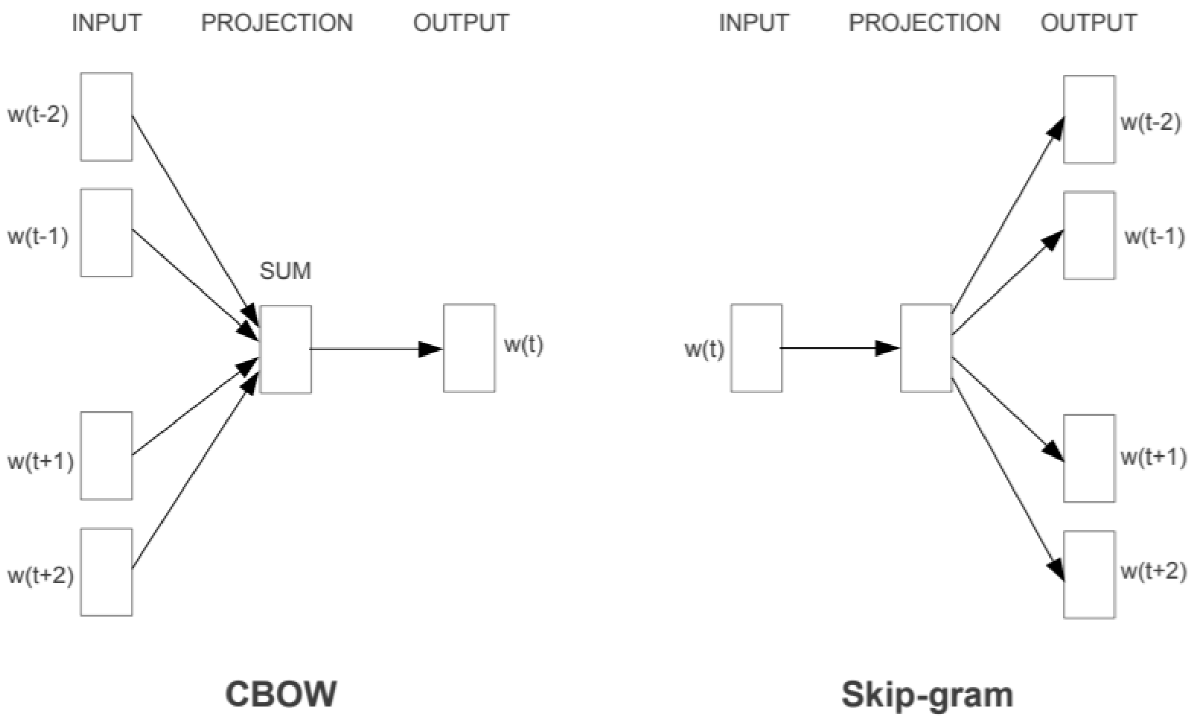


Figure 1: New model architectures. The CBOW architecture predicts the current word based on the context, and the Skip-gram predicts surrounding words given the current word.

Neural Network based Language model - Word2Vec : CBOW

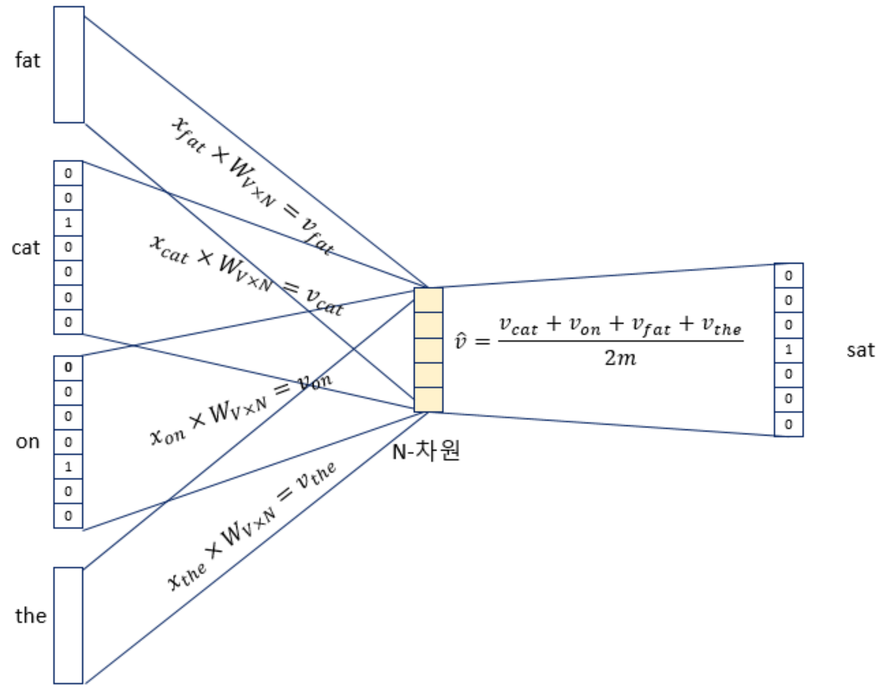
- Window size : m
- Consider all the words before and after (2m)

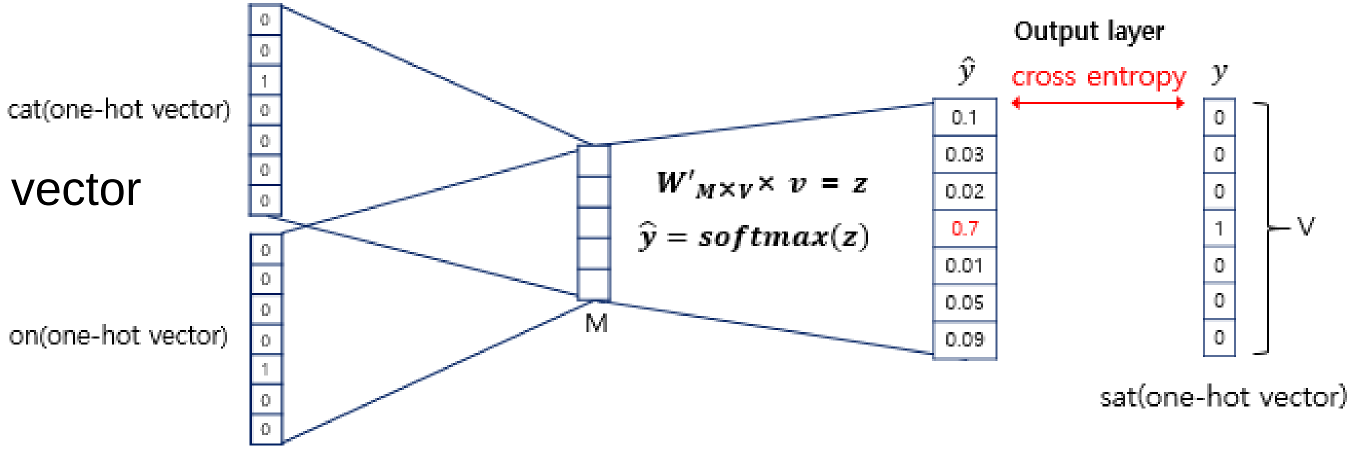
Projection layer

- Averaging all projected word vectors
- No activation function

Output layer

- Averaged vector → Probabilistic vector
- Get loss by comparing with target one-hot vector





Neural Network based Language model - Word2Vec : Skip-gram

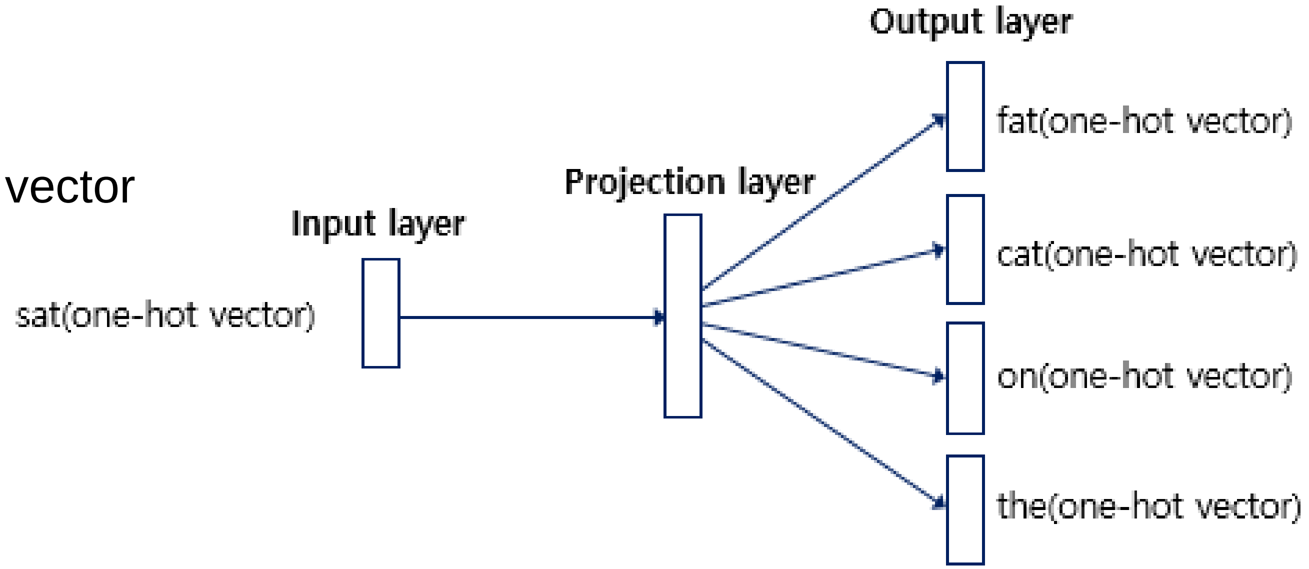
- Window size : m
- Predict all the words before and after (2m)

Projection layer

- Project input word(one-hot vector) to dense vector
- No activation function

Output layer

- Projected vector → 2m one-hot vectors
- Get loss by comparing with target one-hot vector



Neural Network based Language model - Word2Vec : Skip-gram

중심 단어
주변 단어

The fat cat sat on the mat

The fat cat sat on the mat

The fat cat sat on the mat

The fat cat sat on the mat

The fat cat sat on the mat

The fat cat sat on the mat

The fat cat sat on the mat

중심 단어	주변 단어
[1, 0, 0, 0, 0, 0, 0]	[0, 1, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0, 0]	[1, 0, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0, 0], [0, 0, 0, 1, 0, 0, 0]
[0, 0, 1, 0, 0, 0, 0]	[1, 0, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0, 0], [0, 0, 0, 1, 0, 0, 0], [0, 0, 0, 0, 1, 0, 0]
[0, 0, 0, 1, 0, 0, 0]	[0, 1, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0, 0], [0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 1, 0]
[0, 0, 0, 0, 1, 0, 0]	[0, 0, 1, 0, 0, 0, 0], [0, 0, 0, 1, 0, 0, 0], [0, 0, 0, 0, 0, 1, 0], [0, 0, 0, 0, 0, 0, 1]
[0, 0, 0, 0, 0, 1, 0]	[0, 0, 0, 1, 0, 0, 0], [0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 0, 1]
[0, 0, 0, 0, 0, 0, 1]	[0, 0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 1, 0]

CBOW

중심 단어
주변 단어

The fat cat sat on the mat

The fat cat sat on the mat

중심 단어	주변 단어
cat	The
cat	Fat
cat	sat
cat	on
sat	fat
sat	cat
sat	on
sat	the

Skip-gram

Word2Vec : Test

Semantic question

Syntatic question

Table 1: *Examples of five types of semantic and nine types of syntactic questions in the Semantic-Syntactic Word Relationship test set.*

Type of relationship	Word Pair 1		Word Pair 2	
Common capital city	Athens	Greece	Oslo	Norway
All capital cities	Astana	Kazakhstan	Harare	Zimbabwe
Currency	Angola	kwanza	Iran	rial
City-in-state	Chicago	Illinois	Stockton	California
Man-Woman	brother	sister	grandson	granddaughter
Adjective to adverb	apparent	apparently	rapid	rapidly
Opposite	possibly	impossibly	ethical	unethical
Comparative	great	greater	tough	tougher
Superlative	easy	easiest	lucky	luckiest
Present Participle	think	thinking	read	reading
Nationality adjective	Switzerland	Swiss	Cambodia	Cambodian
Past tense	walking	walked	swimming	swam
Plural nouns	mouse	mice	dollar	dollars
Plural verbs	work	works	speak	speaks

Word2Vec : Test

Table 3: *Comparison of architectures using models trained on the same data, with 640-dimensional word vectors. The accuracies are reported on our Semantic-Syntactic Word Relationship test set, and on the syntactic relationship test set of [20]*

Model Architecture	Semantic-Syntactic Word Relationship test set		MSR Word Relatedness Test Set [20]
	Semantic Accuracy [%]	Syntactic Accuracy [%]	
RNNLM	9	36	35
NNLM	23	53	47
CBOW	24	64	61
Skip-gram	55	59	56

Word2Vec : Test

Table 5: *Comparison of models trained for three epochs on the same data and models trained for one epoch. Accuracy is reported on the full Semantic-Syntactic data set.*

Model	Vector Dimensionality	Training words	Accuracy [%]			Training time [days]
			Semantic	Syntactic	Total	
3 epoch CBOW	300	783M	15.5	53.1	36.1	1
3 epoch Skip-gram	300	783M	50.0	55.9	53.3	3
1 epoch CBOW	300	783M	13.8	49.9	33.6	0.3
1 epoch CBOW	300	1.6B	16.1	52.6	36.1	0.6
1 epoch CBOW	600	783M	15.4	53.3	36.2	0.7
1 epoch Skip-gram	300	783M	45.6	52.2	49.2	1
1 epoch Skip-gram	300	1.6B	52.2	55.1	53.8	2
1 epoch Skip-gram	600	783M	56.7	54.5	55.5	2.5

Word2Vec : Result

- The relationship is defined by subtracting two word vectors!
- We can apply linear operation to words!

Table 8: *Examples of the word pair relationships, using the best word vectors from Table 4 (Skip-gram model trained on 783M words with 300 dimensionality).*

Relationship	Example 1	Example 2	Example 3
France - Paris	Italy: Rome	Japan: Tokyo	Florida: Tallahassee
big - bigger	small: larger	cold: colder	quick: quicker
Miami - Florida	Baltimore: Maryland	Dallas: Texas	Kona: Hawaii
Einstein - scientist	Messi: midfielder	Mozart: violinist	Picasso: painter
Sarkozy - France	Berlusconi: Italy	Merkel: Germany	Koizumi: Japan
copper - Cu	zinc: Zn	gold: Au	uranium: plutonium
Berlusconi - Silvio	Sarkozy: Nicolas	Putin: Medvedev	Obama: Barack
Microsoft - Windows	Google: Android	IBM: Linux	Apple: iPhone
Microsoft - Ballmer	Google: Yahoo	IBM: McNealy	Apple: Jobs
Japan - sushi	Germany: bratwurst	France: tapas	USA: pizza

Word2Vec : Conclusion

- Proposed simple and high-performance architecture for word embedding
- Has much lower complexity
- Two architecture : CBOW & Skip-gram
- Can be utilized to various NLP task, like machine translation or question answering

https://word2vec.kr/search/