



Data X

Introduction to NLP-II (Word2vec)

By Sana Iqbal
26th October, 2017

Review

In the last lecture we discussed:

- ❑ Goal of NLP:
Understand the meaning of the text - that is get to semantic level analysis
- ❑ We looked at **bag of words** model, which just keeps the count of words in a document.
 - ❑ They lose the ordering of the words.
 - ❑ They also ignore semantics of the words.
 - ❑ We know order and context of words is important to understand the document.

Discrete Representation Of Words

- ❑ We represent words in our corpus as atomic symbols i.e each word is independant.
- ❑ Eg: 1. 'I love knitting' 2. 'I love dogs'
 - ❑ Vocabulary, $V = [i, \text{love}, \text{knitting}, \text{dogs}]$
 - ❑ If we want to represent them as numbers in our machine we either assign them an id say $i=1, \text{love}=2, \text{knitting}=3, \text{dogs}=4$

Discrete Representation Of Text cntd...

❑ One-hot-Encode

- ❑ We can represent these words as vectors we can say in the vocabulary space $V = [i, \text{love}, \text{knitting}, \text{dogs}]$

- ❑ $i = [1, 0, 0, 0]$

- ❑ $\text{love} = [0, 1, 0, 0]$

- ❑ $\text{knitting} = [0, 0, 1, 0]$

- ❑ $\text{dogs} = [0, 0, 0, 1]$

- ❑ So size of One-hot-Encoded word vectors will depend on the corpus vocabulary size.

Discrete Representation Of Text cntd...

- But we want you use NLP on large corpuses eg the Imdb reviews, the yelp reviews, wikipedia articles etc
- **Google corpus:** a vocabulary of 3 million words Google News dataset

Problems:

- We will end up having very very large sized sparse vectors.
 - good = [1,0.....]
 - nice = [0,1,0.....]
 - We are not able to capture any semantic relations between words.
- Eg: To capture similarity between **good** & **nice** vectors:
- Cosine similarity= Dot(good,nice) = 0

Distributional Representation of Words

- ❑ We want to represent words as vectors that encodes its meaning.
- ❑ To do that we rely on **Distributional similarity** concept.
 - ❑ The idea is that if we look at the different contexts in which a word appears or is used in a language, we will be able to infer its meaning.



Distributional Representation of Words cntd ...

Example:

Eating healthy is a key to fitness.

Junk **eating** causes obesity.

If you stop **eating**, you will die.

Too much **eating** will make you obese.

Not all cultures use spoons for **eating** food.

Eating seen in context of healthy, junk, food, fitness, spoons, die etc. gives the idea of its meaning.

Distributional Representation of Words cntd ...

JR Firth, a British linguist: "You shall know a word by the company it keeps."

- ❑ So in **Distributional Representation**, we want to represent words as vectors that capture the *context* of these words in the corpus.
- ❑ The most intuitive way would be to construct **co-occurrence matrix** of the corpus vocabulary.

Distributional Representation of Words cntd ...

Co-Occurrence Matrix

Corpus:

I like deep learning. I like NLP. I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

Issue:

- High dimensional vectors for large corpus

Solution:

- Use SVD decomposition on Co-occurrence matrix

Issue with SVD:

- High computational cost for large matrices.

Ref: <http://mysite.science.uottawa.ca/phofstra/MAT2342/SVDproblems.pdf>



WORD2VEC

By Mikolov et al.

- ❑ WORD2VEC is a method of creating distributional representations of words called **word embeddings**, using backpropagation.
- ❑ Eg: an = [0.2, 0.35, 0.1, -0.2, 0.15, 0.4]^T

WORD2VEC

- ❑ Models that aim to predict between a center word or words that appear in it's context.

Like any other model:

1. Model is parametrized. Here parameters are '**word vector representations**' of words.
2. Trained using a loss/ objective function
3. Word Vectors readjusted to minimize loss.



WORD2VEC cntd ...

❑ Two algorithms of word2vec:

1. Skip-gram
2. Continuous Bag of words (Cbow)



Skip-gram:

the task is "***predicting the context given a word***".

❑ $p(\text{context} | w_t)$

CBOW:

the task as "***predicting the word given its context***".

❑ $p(w_t | \text{context})$

WORD2VEC cntd ...

If we have:

n = Vocabulary in the corpus

d = Word vector dimension

w = Window size on each side



WORD2VEC - Skip Gram Model

- ❑ **Input:** One hot encoded word, c
- ❑ Weight Matrix V and W which is a matrix of word vectors.
- ❑ Objective Function= maximize $p(\text{context} | c)$
- ❑ $\text{probability}(x_i | c) = \text{softmax}(x_i \cdot c)$

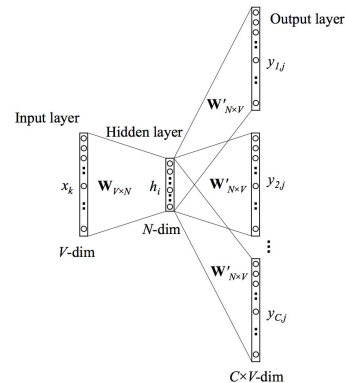
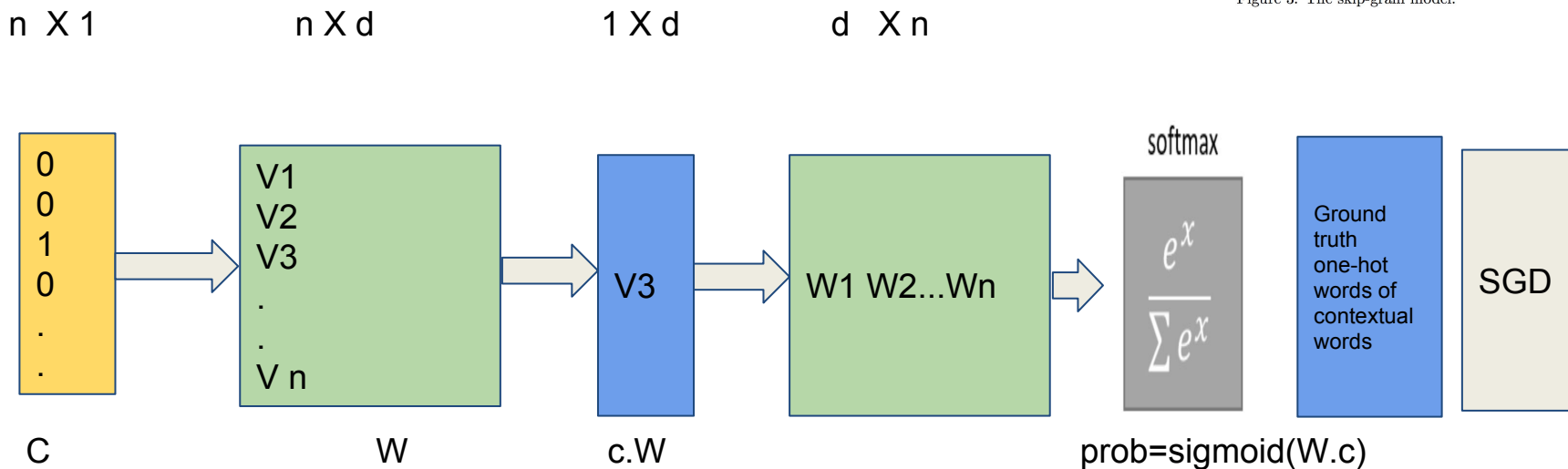


Figure 3: The skip-gram model.



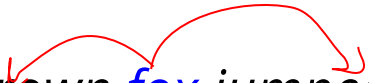
Skip gram example data:

$n=11$

$d=3$

$w=2$, 1 on each side

Corpus: *the quick brown fox jumped over the lazy dog and killed it*

A diagram illustrating the skip-gram model. It shows the sentence "the quick brown fox jumped over the lazy dog and killed it". The word "fox" is highlighted in blue. Two red curved arrows originate from "fox": one points to the word "brown" (two words to the left) and the other points to the word "jumped" (one word to the right), representing the context window of size 2.

Output

[the, brown]

[quick, fox]

[brown, jumped]

...

input

quick

brown

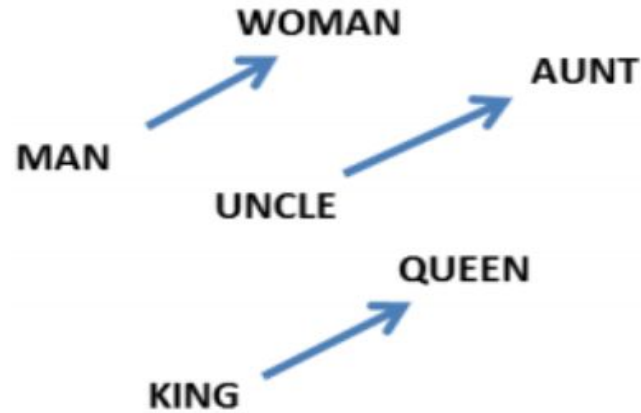
fox

Results with word2vec in the original paper trained on Google news dataset

FRANCE	JESUS	XBOX	REDDISH	SCRATCHED	MEGABITS
AUSTRIA	GOD	AMIGA	GREENISH	NAILED	OCTETS
BELGIUM	SATI	PLAYSTATION	BLUISH	SMASHED	MB/S
GERMANY	CHRIST	MSX	PINKISH	PUNCHED	BIT/S
ITALY	SATAN	IPOD	PURPLISH	POPPED	BAUD
GREECE	KALI	SEGA	BROWNISH	CRIMPED	CARATS
SWEDEN	INDRA	PSNUMBER	GREYISH	SCRAPED	KBIT/S
NORWAY	VISHNU	HD	GRAYISH	SCREWED	MEGAHERTZ
EUROPE	ANANDA	DREAMCAST	WHITISH	SECTIONED	MEGAPIXELS
HUNGARY	PARVATI	GEFORCE	SILVERY	SLASHED	GBIT/S
SWITZERLAND	GRACE	CAPCOM	YELLOWISH	RIPPED	AMPERES



Results with word2vec in the original paper trained on Google news dataset



From Mikolov *et al.*
(2013a)

Results with word2vec in the original paper trained on Google news dataset

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

