Word2Vec

Outline

Semantics

- Matrix factorization
- Word2vec
- Skip-Thought vectors

Text Semantics

- In Natural Language Processing (NLP), ***semantics*** is concerned with the meanings of texts.
- There are two main approaches:
  - ***Propositional or formal semantics:*** A block of text is to converted into a formula in a logical language, e.g. predicate calculus.
  - ***Vector representation.*** Texts are ***embedded*** into a high-dimensional space.

Semantic Approaches

Propositional:

- “dog bites man” □ bites(dog, man)
- bites(*,*) is a binary relation. man, dog are objects.
- Probabilities can be attached.

Vector representation:

- vec(“dog bites man”) = (0.2, -0.3, 1.5,…) $\in \Re^n$
- Sentences similar in meaning should be close to this embedding (e.g. use human judgments)

Vector Embedding of Words

Word embeddings depend on a notion of ***word similarity***.

A very useful definition is paradigmatic similarity:

Similar words occur in ***similar contexts***. They are ***exchangable***.

Yesterday { POTUS / The President / Obama } called a press conference

Vector Embedding of Words

Much of the work on text embedding has used word embeddings and bag-of-words representation:

vec(“dog”) = (0.2, -0.3, 1.5,…)

vec(“bites”) = (0.5, 1.0, -0.4,…)

vec(“man”) = (-0.1, 2.3, -1.5,…)

vec(“dog bites man”) = (0.6, 3.0, -0.4,…)

Vector Embedding: Word Similarity

Word embeddings depend on a notion of ***word similarity***.

A very useful definition is paradigmatic similarity:

Similar words occur in ***similar contexts***. They are ***exchangable***.

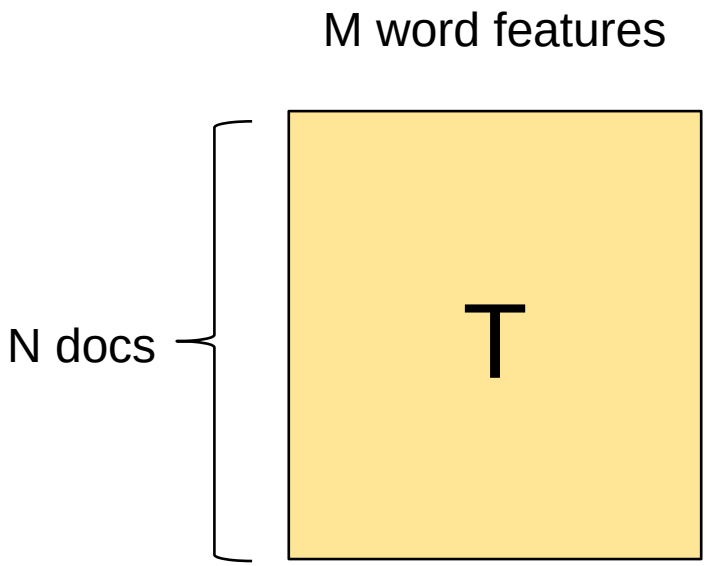
This definition supports unsupervised learning: cluster or embed words according to their contexts.

Embedding: Latent Semantic Analysis

Latent semantic analysis studies documents in **Bag-Of-Words format** (1988).

i.e. given a matrix T encoding some documents:

T_{ij} is the count* of word j in document i. Most entries are 0.



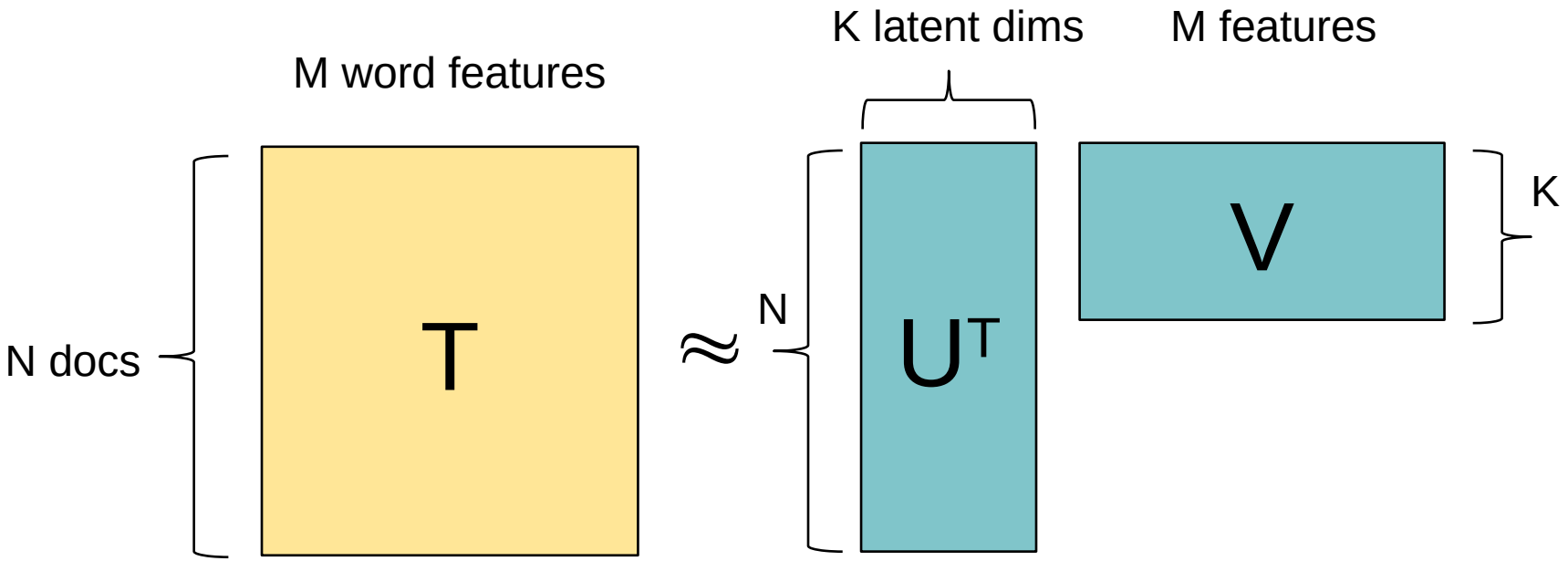
* Often tfidf or other “squashing” functions of the count are used.

Embedding: Latent Semantic Analysis

Given a bag-of-words matrix T, compute a factorization $T \approx U^T * V$ (e.g. a best L_2 approximation to T)

Factors encode similar ***whole document contexts***.

Factors are rows of V.

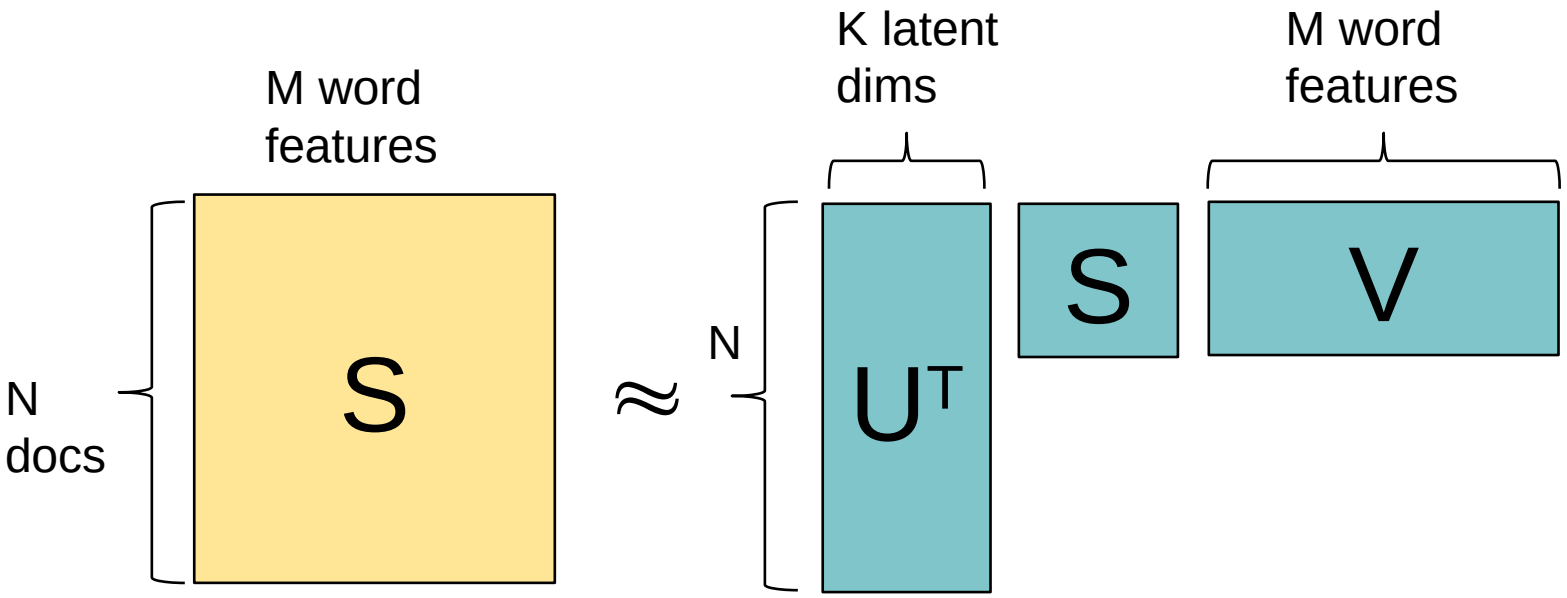


Embedding: Latent Semantic Analysis

If in addition U and V are orthogonal, S a diagonal matrix of singular values. Then if is a document (row of):

 is an embedding of the document in the latent space.

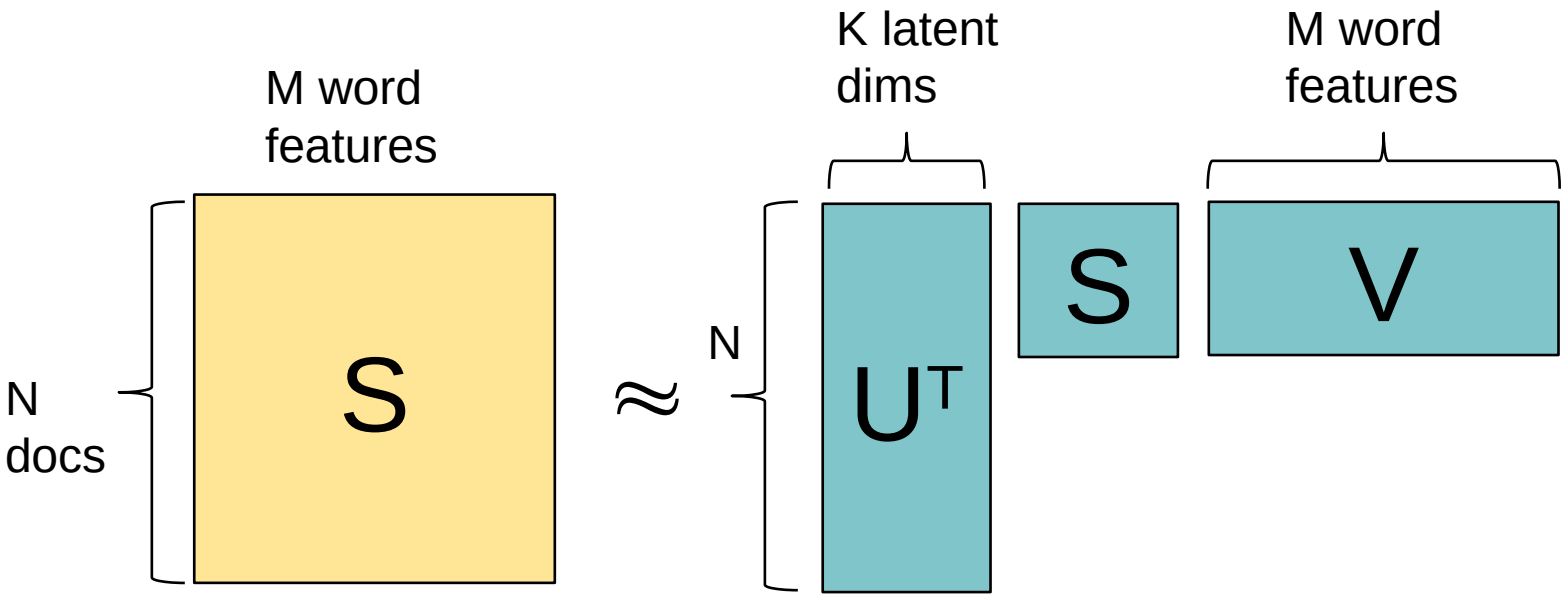
 is the decoding of the sentence from its embedding.



Embedding: Latent Semantic Analysis

is the decoding of the sentence from its embedding.

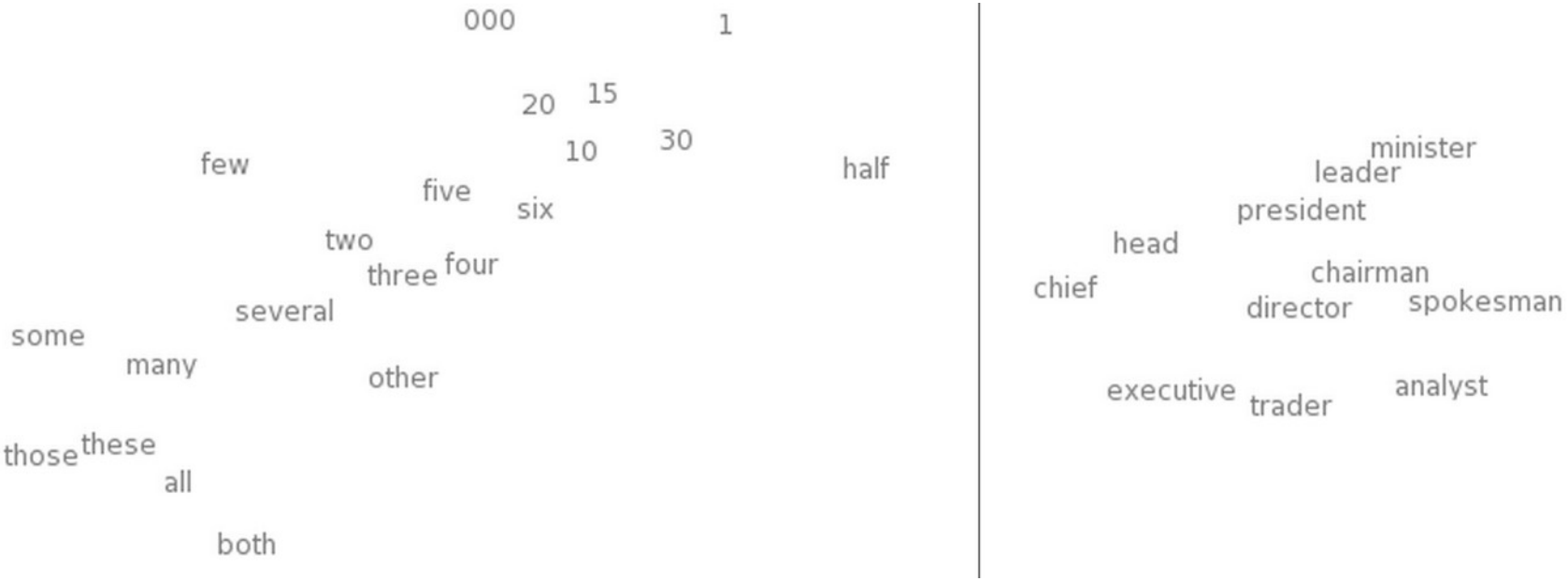
An SVD factorization gives the ***best possible reconstructions*** of the documents from their embeddings.



t-SNE of Word Embeddings

From: “Word representations: A simple and general method for semi-supervised learning” Joseph Turian, Lev Ratinov, Yoshua Bengio, ACL 2010.

t-SNE of Word Embeddings



Left: Number Region; Right: Jobs Region

from “Deep Learning, NLP, and Representations” by Chris Olah. See also

http://colah.github.io/posts/2015-01-Visualizing-Representations/

Word2vec: Local contexts

Instead of entire documents, Word2vec uses words a few positions away from each center word. The pairs of center word/context word are called **“skip-grams.”**

“It was **a bright cold day in April, and** the clocks were striking”

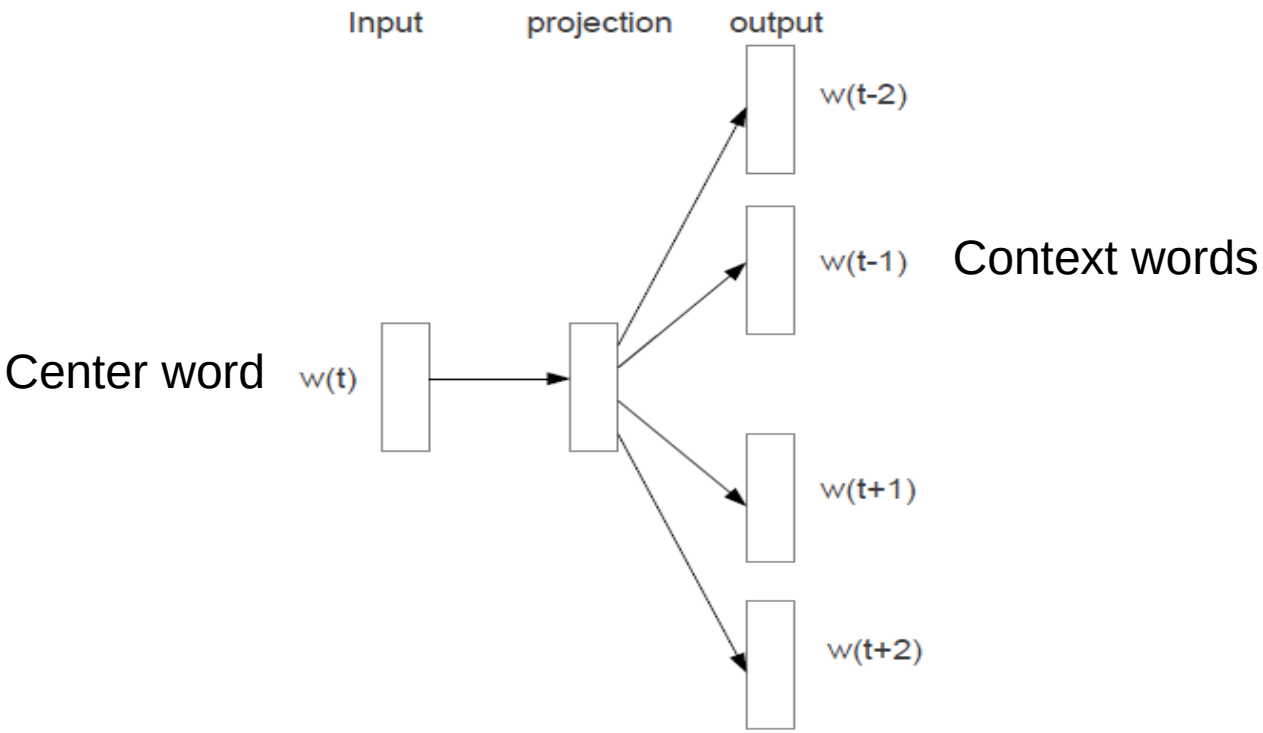
Center word: red

Context words: blue

Word2vec considers all words as center words, and all their context words.

Word2vec: Local contexts

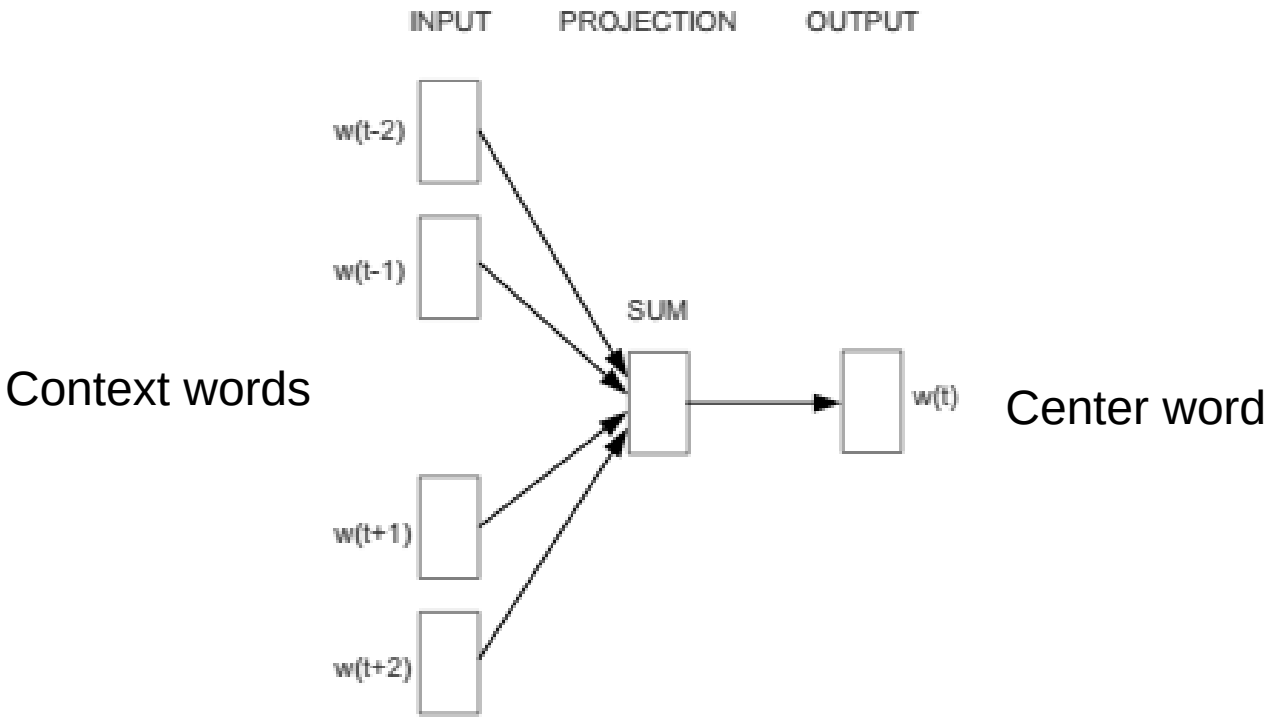
The pairs of center word/context word are called **“skip-grams.”** Typical distances are 3-5 word positions. Skip-gram model:



Distributed Representations of Words and Phrases and their Compositionality
Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, NIPS 2013

Word2vec: Local contexts

Models can also predict center word from context, CBOW model. Generally, skip-gram performs better.



Distributed Representations of Words and Phrases and their Compositionality
Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, NIPS 2013

Word2vec: Local contexts

Word2vec optimizes a softmax loss for each output word:

Where is the output word, is the input word. ranges over a context of $\pm$ 3-5 positions around the input word.

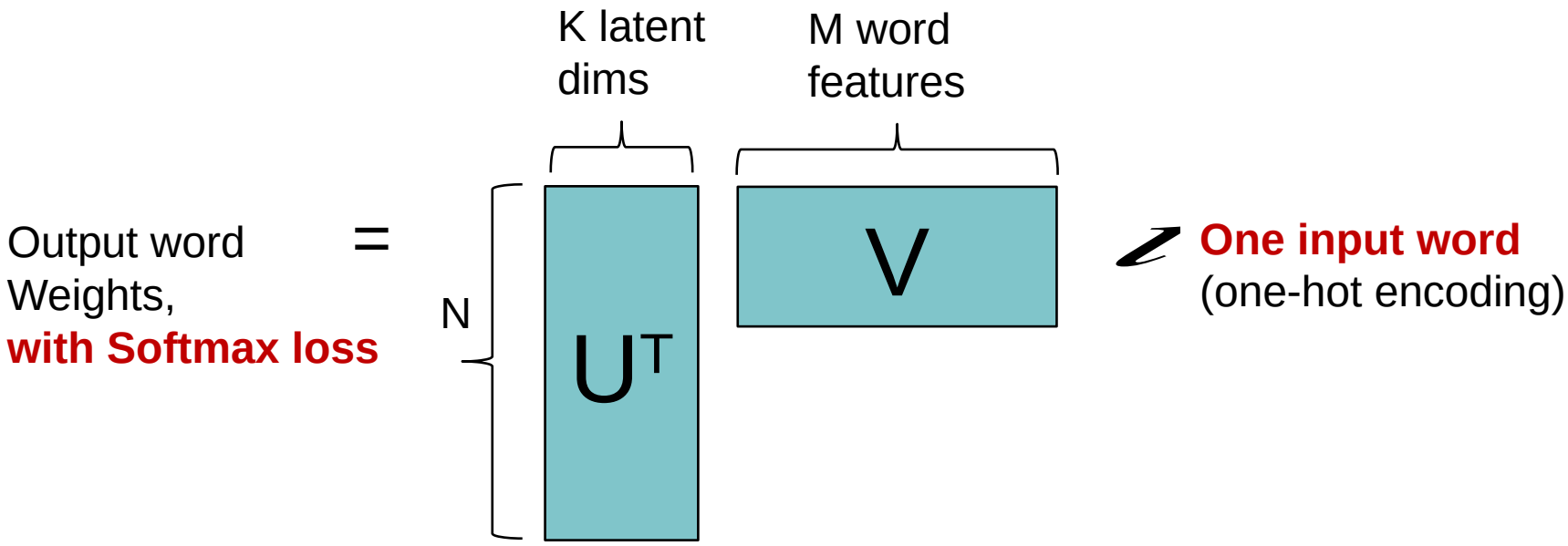
 is an output embedding vector.

 is an input embedding vector.

Word2vec can be implemented with standard DNN toolkits, by backpropagating to optimize and .

Matrix perspective

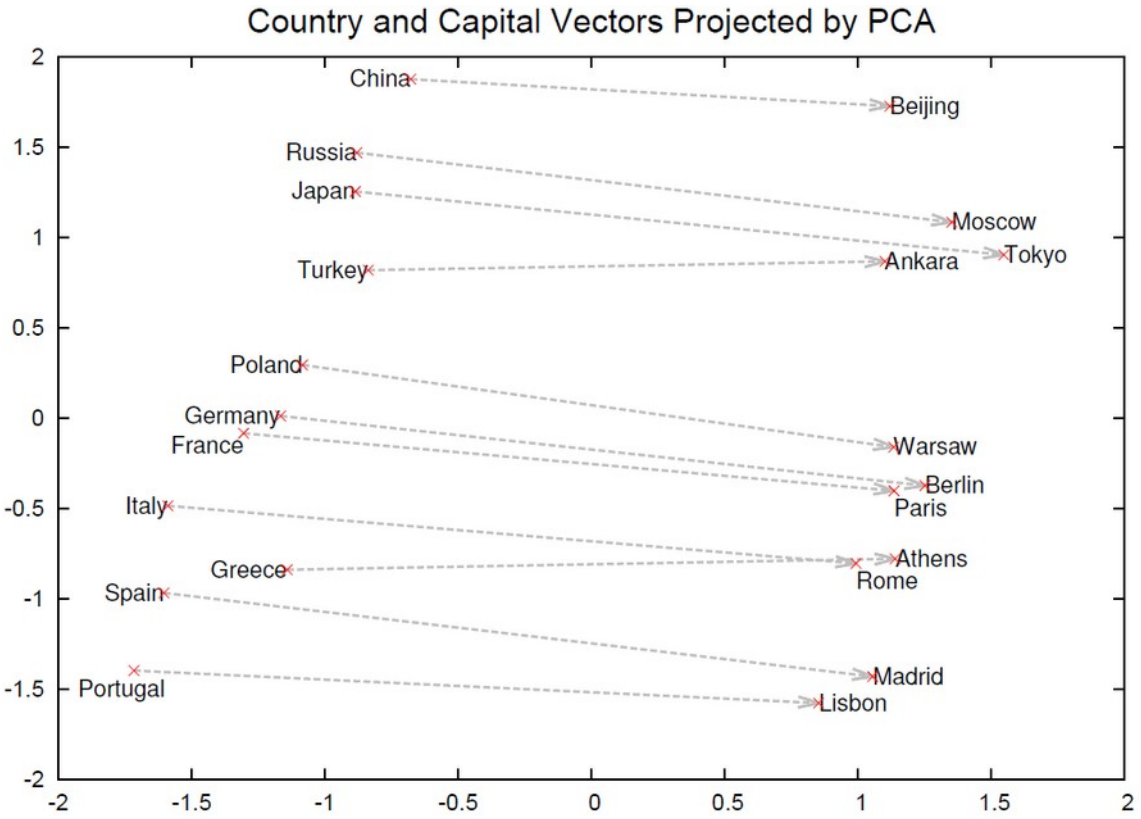
Using matrix representation:



See: “GloVe: Global Vectors for Word Representation” Jeffrey Pennington, Richard Socher, Christopher D. Manning, 2014

Word2vec: Local contexts

Local contexts capture much more information about relations and properties than LSA:

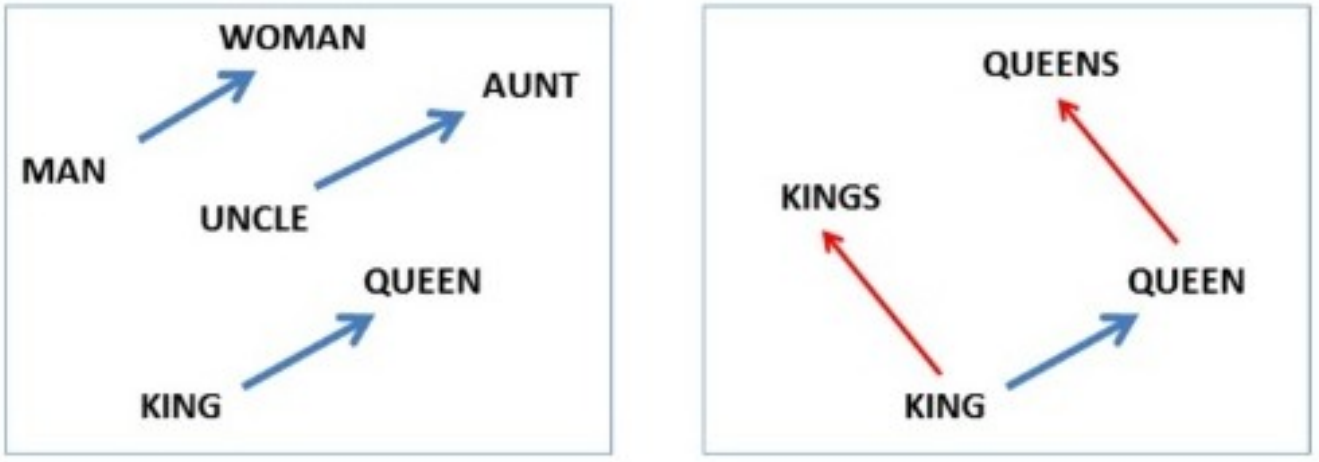


Composition

Algebraic relations:

vec(“woman")−vec(“man") ≃ vec(“aunt")−vec(“uncle")

vec(“woman")−vec(“man") ≃ vec(“queen")−vec(“king")



From “Linguistic Regularities in Continuous Space Word Representations”
Tomas Mikolov , Wen-tau Yih, Geoffrey Zweig, NAACL-HLT 2013

Relations Learned by Word2vec

Word2vec model computed from 6 billion word corpus of news articles

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

“Efficient Estimation of Word Representations in Vector Space” Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

“Efficient Estimation of Word Representations in Vector Space” Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

“Efficient Estimation of Word Representations in Vector Space” Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

"Efficient Estimation of Word Representations in Vector Space" Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

"Efficient Estimation of Word Representations in Vector Space" Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

"Efficient Estimation of Word Representations in Vector Space" Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

"Efficient Estimation of Word Representations in Vector Space" Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013

Relations Learned by Word2vec

A relation is defined by the vector displacement in the first column. For each start word in the other column, the closest displaced word is shown.

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

"Efficient Estimation of Word Representations in Vector Space" Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Arxiv 2013