

Natural Language Processing

Fundamentals of Artificial Intelligence

Fabien Cromieres

fabien@nlp.ist.i.kyoto-u.ac.jp

Kyoto University

Natural Language Processing

- **Natural Language Processing** is a subfield of Computer Science
- It is about processing texts in **Human Languages** (English, Japanese, etc.)
- It covers many tasks:
 - Machine Translation

Machine Translation

  Traduction   

📄 Texte

 Documents

JAPONAIS - DÉTECTÉ

FRANÇAIS

ANGLAIS

A ▼

↔

FRANÇAIS

ANGLAIS

ARABE ▼

人間広告塔は、広告を身につけている人物。最も一般的なのは、何らかの広告看板を持っているか、身にまとっている形態であるが、衣服として広告を着用している場合や、極端な例では身体に直接、入れ墨として広告を入れている場合なども含まれることがある

Ningen kōkokutō wa, kōkoku o mi ni tsukete iru jinbutsu. Mottomo ippantekina no wa, nanrakano kōkoku kanban o motte iru ka, mi ni matotte iru keitaidearuga, ifuku to shite kōkoku o chakuyō shite iru

[Afficher plus](#)



→

×

A human advertisement tower is a person wearing advertisements. The most common is the case where you have some advertisement signboard or wearing yourself but when you are wearing advertisements as clothing or in extreme cases you put ads as tattoos directly on your body May also be included

Natural Language Processing

- Natural Language Processing is a subfield of Computer Science
- It is about processing texts in Human Languages (English, Japanese, etc.)
- It covers many tasks:
 - Machine Translation
 - Automatic summarization

Automatic summarization

Russian defense minister Ivanov called Sunday for the creation of a joint front for combating global terrorism



Russia calls for a joint front against terrorism

Natural Language Processing

- Natural Language Processing is a subfield of Computer Science
- It is about processing texts in Human Languages (English, Japanese, etc.)
- It covers many tasks:
 - Machine Translation
 - Automatic summarization
 - Human-Machine Dialog (chatbots, etc.)

Question Answering and Dialog



Computer, who was the Prime Minister of Japan in 2002?



It was Jun'Ichiro Koizumi



How long was he Prime Minister?



About 5 years



Is he still alive?



Yes

Some interactive demonstration

- <https://demo.allennlp.org/reading-comprehension>

Natural Language Processing

- Natural Language Processing is a subfield of Computer Science
- It is about processing texts in Human Languages (English, Japanese, etc.)
- It covers many tasks:
 - Machine Translation
 - Automatic summarization
 - Human-Machine Dialog (chatbots, etc.)
 - Text understanding (Sentiment Analysis, etc.)

Sentiment Analysis

Imdb comment

Since Disney is incompetent of coming up with new ideas, and must resort to using older stories they did years ago, they certainly better live up to what they're doing. This does not happen with Mary Poppins Returns.



This user thinks the movie is bad or good?

Bad

amazon.com comment

I just got these today and they were exactly what I was looking for. They are lightweight without being too thin. Almost my entire house is hardwood or tile, and I use these on my swiffer mops in place of the expensive wet cloths



This user liked the product or not?

Liked

Natural Language Processing

- Natural Language Processing is a subfield of Computer Science
- It is about processing texts in Human Languages (English, Japanese, etc.)
- It covers many tasks:
 - Machine Translation
 - Automatic summarization
 - Text understanding (Sentiment Analysis, etc.)
 - Human-Machine Dialog (chatbots, etc.)

Natural Language Processing

- **Natural Language Processing** is a subfield of Computer Science
- It is about processing texts in **Human Languages** (English, Japanese, etc.)
- Like Computer Vision, it is now dominated by approaches based on **Neural Networks**
- Slightly more recent trend:
 - Neural Networks beat other methods for Image recognition since 2012
 - Neural Networks beat other methods for Machine Translation since 2015~2016

Sequence Processing

- A **text** is a **sequence of words**
- The Neural Network architectures we are going to see can also be applied to **any sequence of symbols**
- For example:
 - Music sheets (sequences of notes)
 - DNA sequences (sequence of genes)
 - Etc.

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
- Mainly for 2 reasons:

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
- Mainly for 2 reasons:
 - Text is not “naturally” represented by numbers
 - Text can be of unlimited length
 - And it cannot be “naturally” reduced to a fixed size

Text as numbers

- How to see this sentence as a set of numbers?

Let us go to the beach!

- First, we look at it as a sequence of symbols. This is called **“Tokenization”**

Let

us

go

to

the

beach

!

Tokenization

For a language like English, tokenization is easy: we just “cut” along the spaces:

Let us go to the beach!

Let

us

go

to

the

beach

!

For a language like Japanese, not so simple. Two options:

Use a specialized software (called “segmenter”) that can recognize limits of Japanese words:

ビーチ

へ

行こう

!

ビーチへ行こう！

Cut out each character:

ビ

ー

チ

へ

行

こ

う

！

Text as numbers

- How to see this sentence as a set of numbers?

Let us go to the beach!

- First, we look at it as a sequence of symbols. This is called **“Tokenization”**

Let

us

go

to

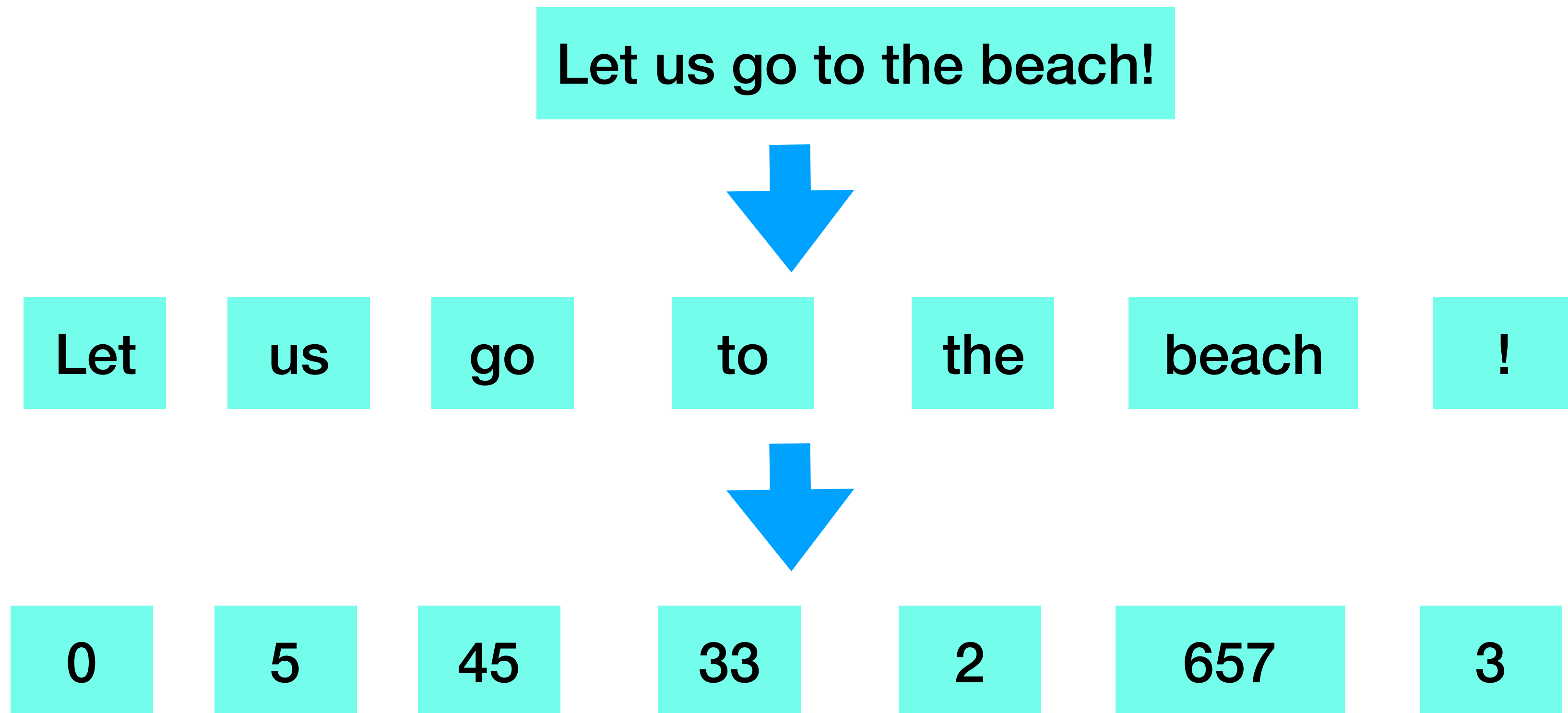
the

beach

!

Text as numbers

- Then maybe we can just assign a number to each word?
 - “Let” = 0
 - “Us” = 5
 - “go” = 45
 - ...



Text as numbers

- Then maybe we can just assign a number to each word?

- “Let” = 0

- “Us” = 5

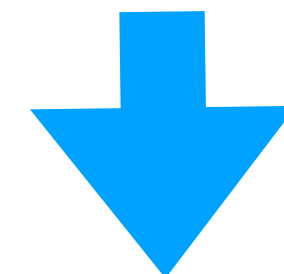
- “go” = 45

- ...

Will not work!

**The relations between words
cannot be represented by a
simple number**

Let us go to the beach!



Let

us

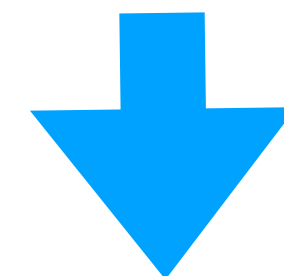
go

to

the

beach

!



0

5

45

33

2

657

3

Words are not independent!

- Intuitively, we have some notion of distance between words

Close in meaning:

Big

Large

Gigantic

Not so close in meaning:

Big

Green

Lawyer

Words are not independent!

- Intuitively, we have some notion of relation between words

Similar relations:

Big ↔ **Bigger**

Small ↔ **Smaller**

Light ↔ **Lighter**

Similar relations

France ↔ **Paris**

Germany ↔ **Berlin**

Japan ↔ **Tokyo**

Text as numbers

- Words are more **expressive** than single numbers
- We will not get good results by using a single number for each word
- What about using more than one number?

Vectors

Addition:

$$\begin{bmatrix} 0.2 & 3.6 & 2.1 & 5.3 & -2.2 \end{bmatrix} + \begin{bmatrix} 1.1 & -2.0 & 1.1 & -5.3 & -1.0 \end{bmatrix} = \begin{bmatrix} 1.3 & 1.6 & 3.2 & 0.0 & -3.2 \end{bmatrix}$$

Norm:

$$\left\| \begin{bmatrix} 1 & -1 & 2 & 5 & -2 \end{bmatrix} \right\| = \sqrt{1^2 + (-1)^2 + 2^2 + 5^2 + (-2)^2} = \sqrt{32} \approx 5.65$$

Note: Norm is always positive

Distance: $d(\vec{x}, \vec{y}) = \left\| \vec{x} - \vec{y} \right\|$

$$d\left(\begin{bmatrix} 1 & -1 & 2 & 5 & -2 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 & 4 & 2 \end{bmatrix} \right) = \sqrt{(1-1)^2 + (0-1)^2 + (2-2)^2 + (5-4)^2 + (-2-2)^2} \approx 4.24$$

Text as numbers

- Words are more expressive than single numbers
- We will not get good results by using a single number for each word
- What about using more than one number?
- We have a **notion of distance on vectors**

Word Embeddings

- A word embedding associate a **fixed-size vector** to *each word* in a **vocabulary**

*100 000 words
in our Vocabulary*

*100 000 vectors
of dimension 200*

Madrid

[2,4,1.2,..., − 1.3,4]

Big

[5,3,1.2,...,5, − 4]

Japan

[1,2.1, − 1,...,2,2.1]

Large

[6,2.9, − 2.2,...,2,2.7]

Small

[−2.2,2.3, − 1,...,2,5.1]

Spain

[1.5,41, − 1,...,2,2.3]



Word Embeddings

- **Word embeddings** are vectors associated to each word in a vocabulary
- The vectors are expected to be “meaningful”:
 - Vectors for words “big” and “large” should be close together

$$d(\overrightarrow{big}, \overrightarrow{large}) < d(\overrightarrow{big}, \overrightarrow{lawyer})$$

- Relation between vectors should reflect semantic relations between words:

$$\overrightarrow{King} - \overrightarrow{Man} + \overrightarrow{Woman} \approx \overrightarrow{Queen}$$

$$\overrightarrow{Paris} - \overrightarrow{France} + \overrightarrow{Japan} \approx \overrightarrow{Tokyo}$$

How to obtain meaningful Word Embeddings?

- We can obtain these Word Embeddings by training a Simple Neural Network on text data
- In practice, we will typically use vectors of dimension 200 to ~4000

Skip-Gram and CBOW (word2vec)

- (Mikolov+, 2013) proposed two very efficient simple models for learning word embeddings
- These models are sometimes referred to as **word2vec** models (word2vec being the name of the software that was released for producing these vectors)
- In the paper, the models are actually called **CBOW** and **Skip-Gram**
- Let us see how “Skip-Gram” works (we will see a slightly modified version)

Skip-Gram (word2vec)

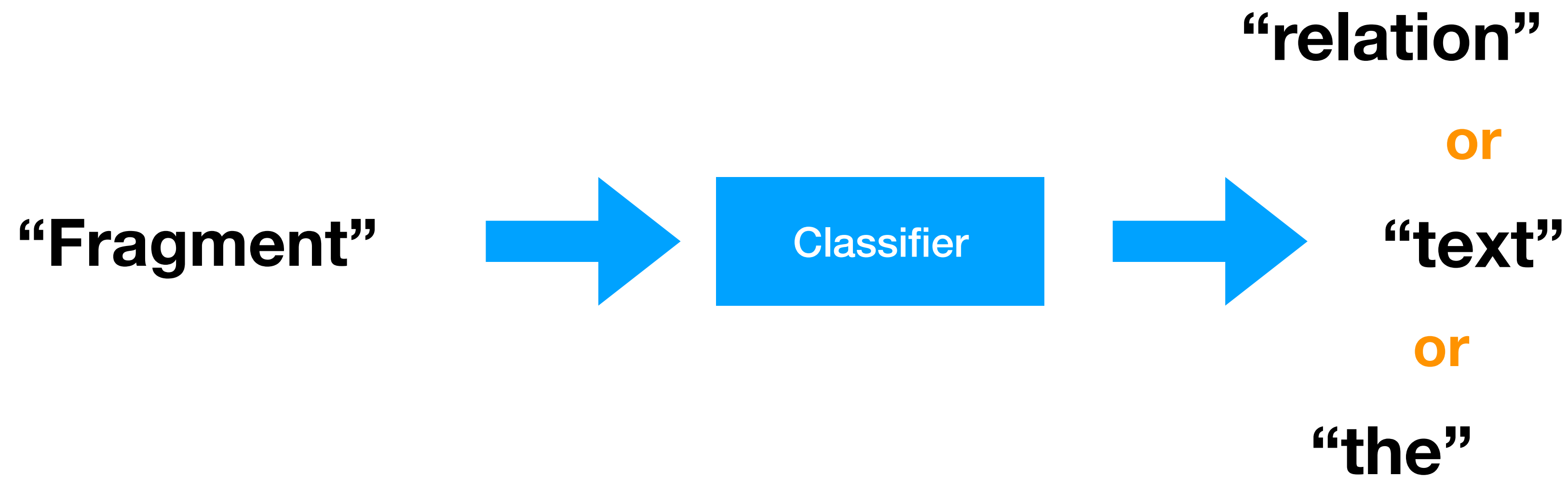
- For each word w in a corpus C , we consider its ***context*** as the c words that come before and the c words that come after

Textual entailment (TE) in natural language processing is a directional relation between text **fragments**. The relation holds whenever the truth of one text fragment follows from another text. In the TE framework, the entailing and entailed texts are termed text (t) and hypothesis (h), respectively. Textual entailment is not the same as pure logical entailment — it has a

- *Context* of this instance of **fragments** (if $c = 3$):
 - relation, between, text, the, relation, holds

Skip-Gram (word2vec)

- We are going to train a simple Neural Network classifier that tries to predict context words given an input word



Multiclass Classification

Inputs:



Multiclass Classifier
(7 classes)

Which animal? (cat, dog, lion, bird, fish, human, other)



Predicted Class:

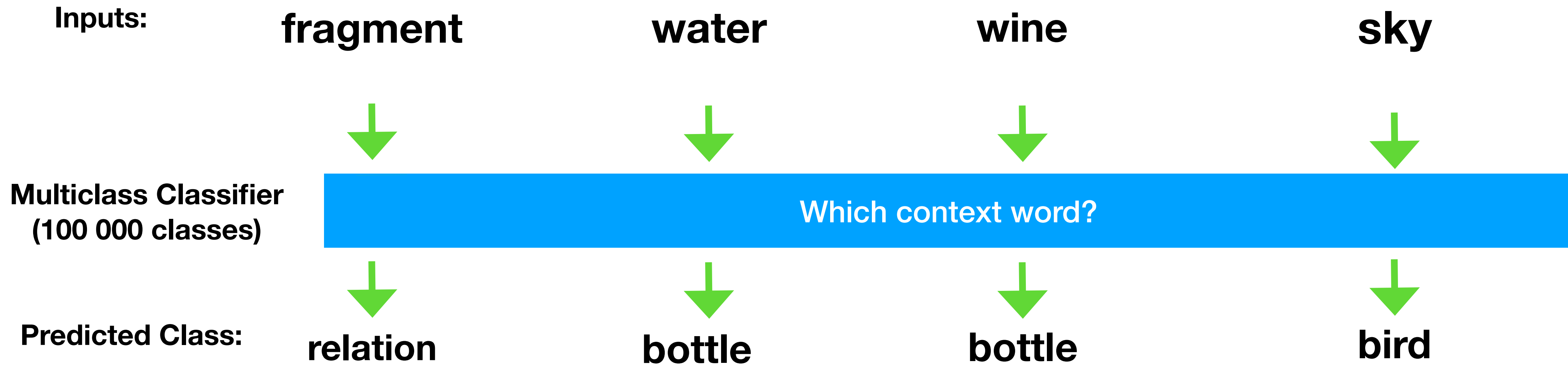
Cat

Dog

Bird

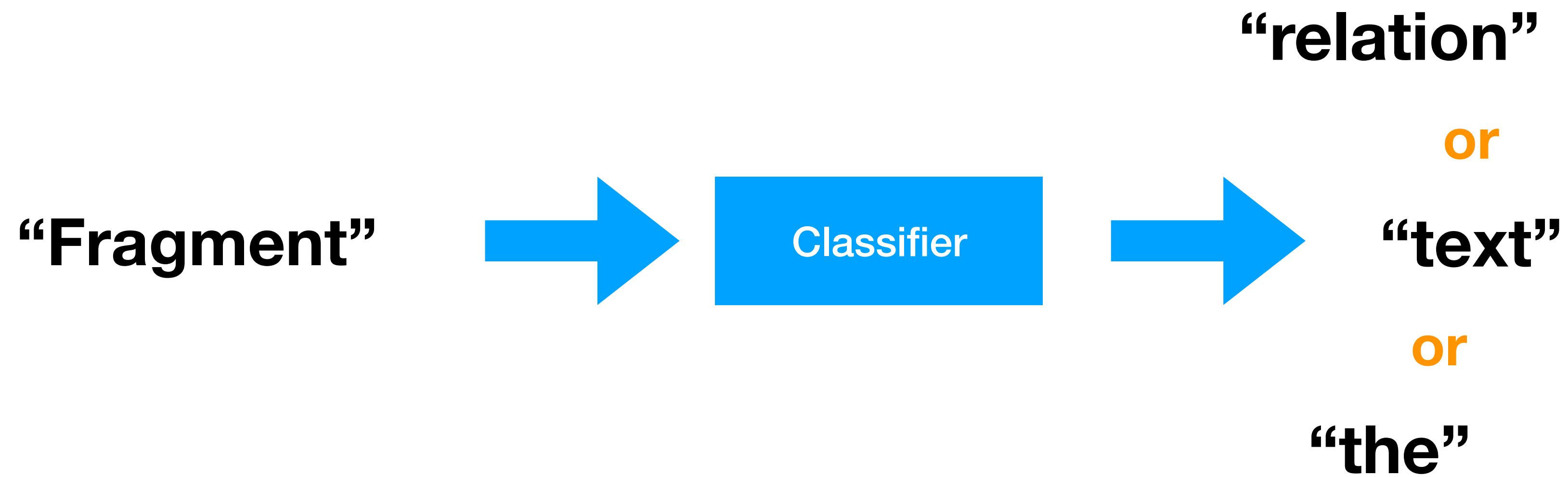
Cat

Multiclass Classification



Skip-Gram (word2vec)

- We are going to train a simple Neural Network classifier that tries to predict context words given an input word



Skip-Gram (word2vec)

- We are going to train a simple Neural Network classifier that tries to predict context words given an input word

“Dictionary” of Embeddings for each word

“Fragment”	[2,4,1.2,..., − 1.3,4]
“Bottle”	[5,3,1.2,...,5, − 4]
“Japan”	[1,2.1, − 1,...,2,2.1]
“Tokyo”	[6,2.9, − 2.2,...,2,2.7]
“ ... ”	...

Simple Neural Network
Classifier with one layer of
100 000 neurons

“Fragment”

[2,4,1.2,..., − 1.3,4]

“relation”

or

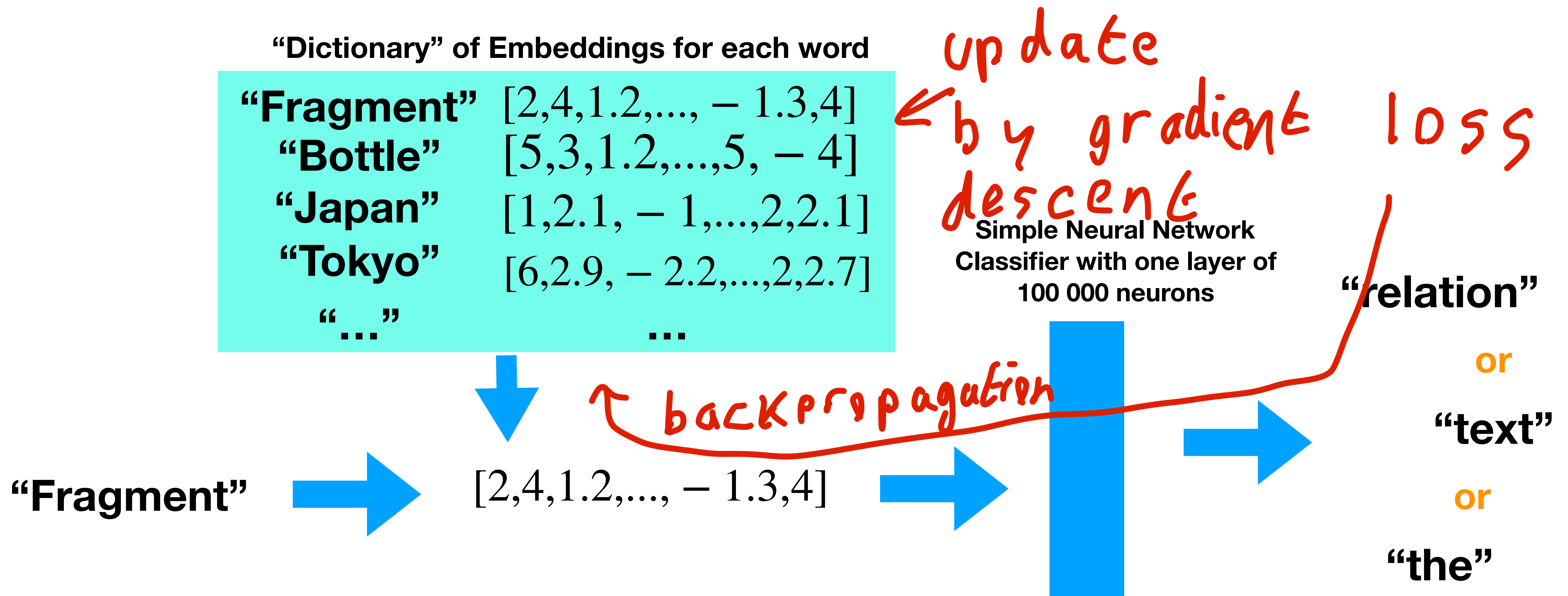
“text”

or

“the”

Skip-Gram (word2vec)

- During training, we use back propagation and gradient descent to update the embeddings in the dictionary



Skip-Gram (word2vec)

- Training data is easy to obtain. Just download Wikipedia, for example. Then extracts all pairs of (input word, context word)

Textual entailment (TE) in natural language processing is a directional relation between text **fragments**. The relation holds whenever the truth of one text fragment follows from another text. In the TE framework, the entailing and entailed texts are termed text (t) and hypothesis (h), respectively. Textual entailment is not the same as pure logical entailment — it has a

- *Context* of this instance of **fragments** (if $c = 3$):
 - relation, between, text, the, relation, holds

Skip-Gram (word2vec)

- The embeddings obtained in this way work very well to express relationship between words

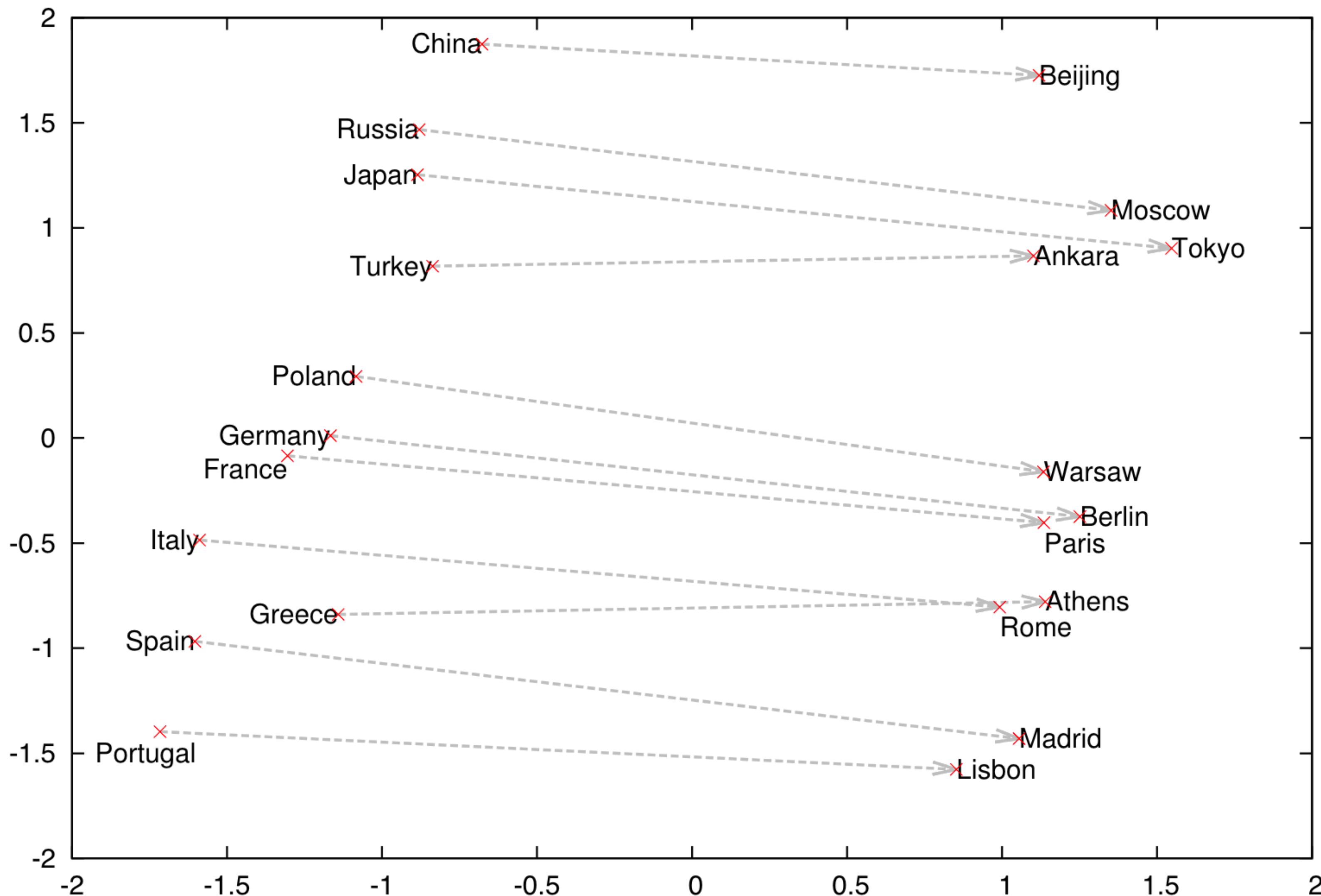
$$d(\overrightarrow{big}, \overrightarrow{large}) < d(\overrightarrow{big}, \overrightarrow{lawyer})$$

$$\overrightarrow{King} - \overrightarrow{Man} + \overrightarrow{Woman} \approx \overrightarrow{Queen}$$

$$\overrightarrow{Paris} - \overrightarrow{France} + \overrightarrow{Japan} \approx \overrightarrow{Tokyo}$$

Skip-gram Embeddings

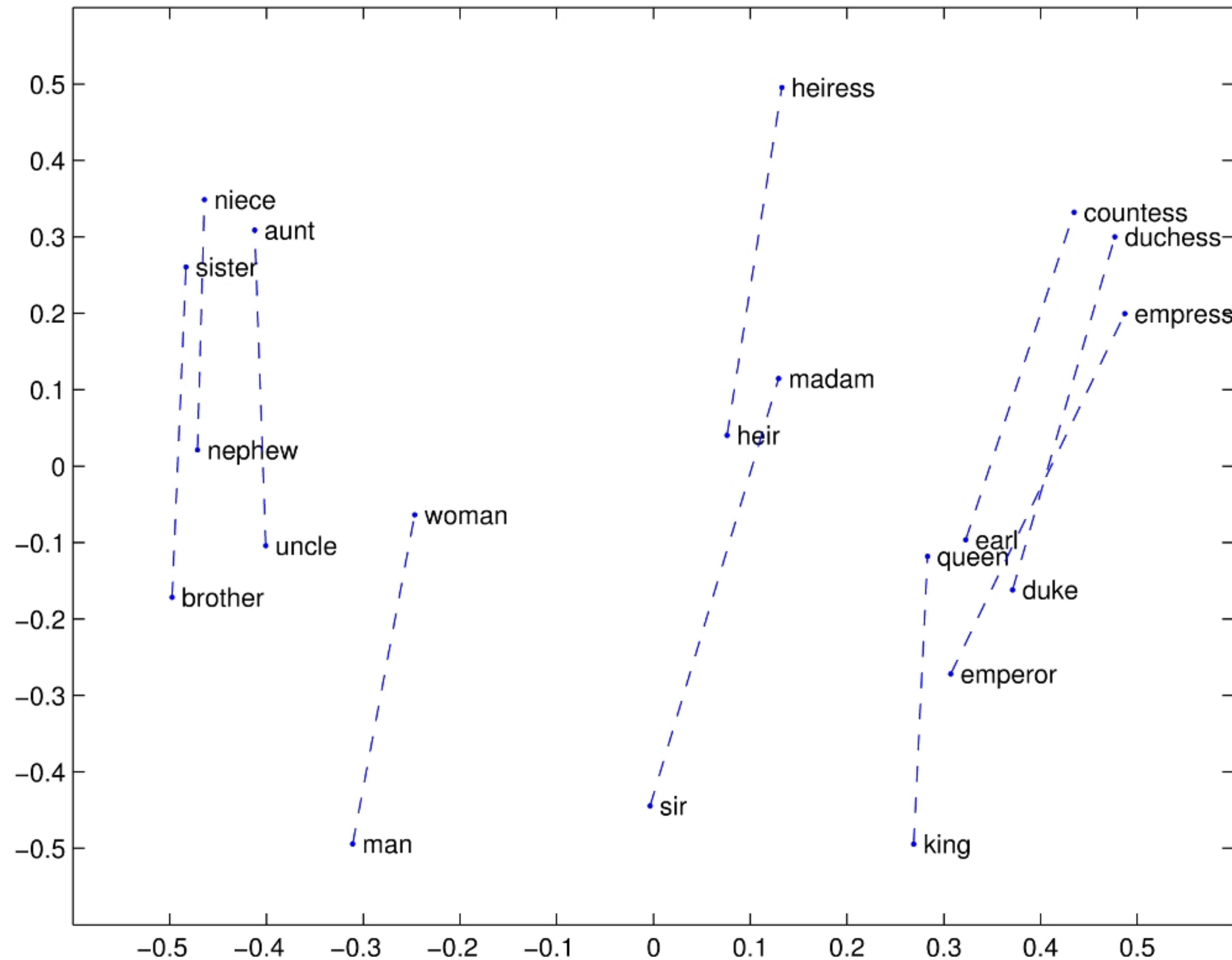
Country and Capital Vectors Projected by PCA



From (Mikolov+, 2013)

- If we map the embeddings into 2 dimension, we can visualize the relationships
- PCA = Principal Component Analysis
- Reduces dimension of vectors to 2

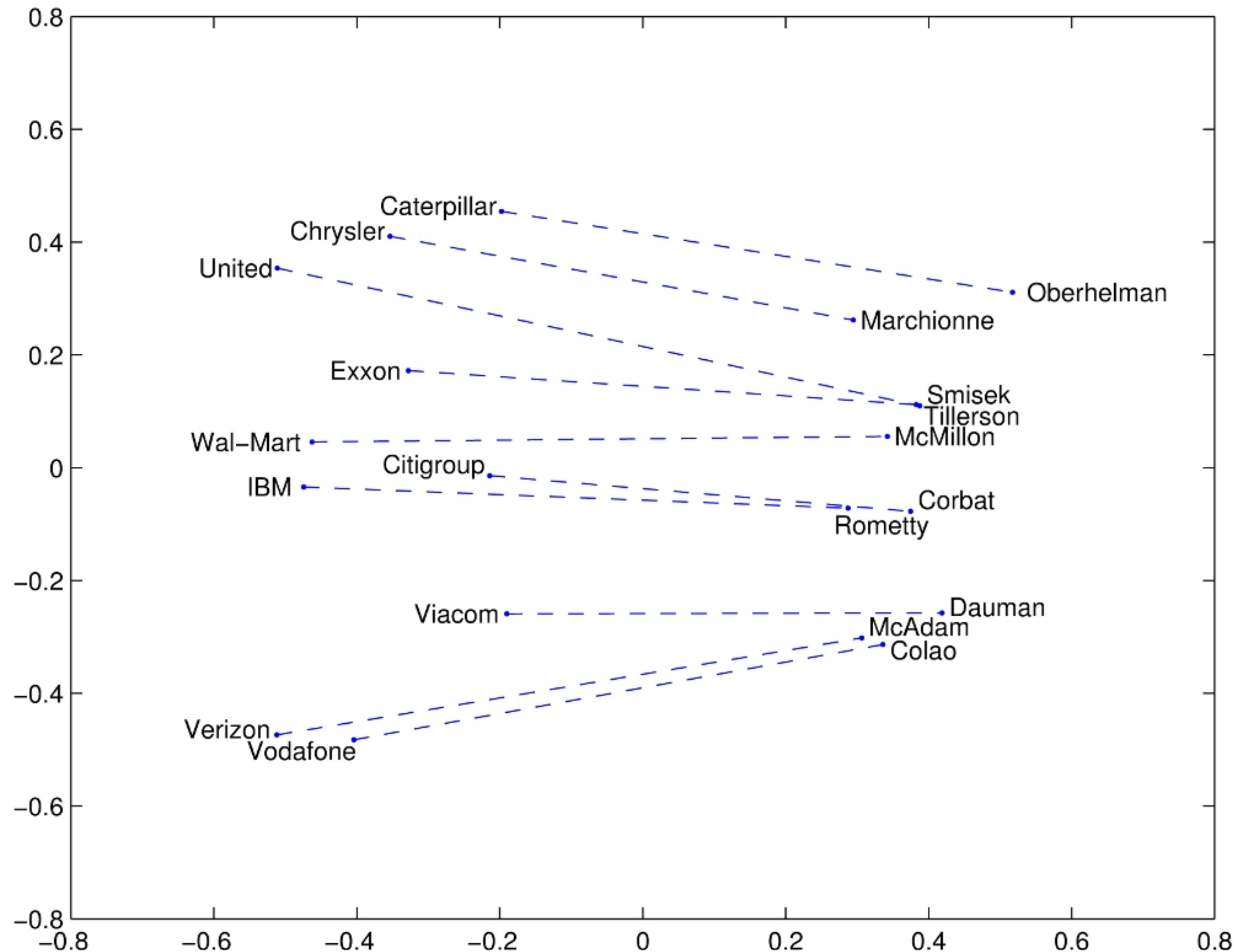
Other examples



Men-Women

**(Not Skip-Gram, but
similar model)**

Other Examples



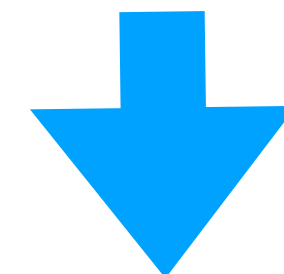
Company-CEO

**(Not Skip-Gram, but
similar model)**

Text as numbers

- We solved our first issue: now we know how to see words as numbers

Let us go to the beach!



Let

us

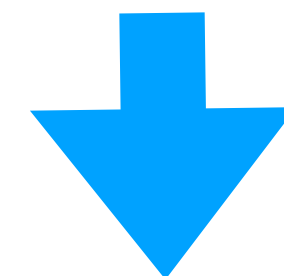
go

to

the

beach

!



Embeddings (in practice 200~4000 dimensions for each vector)

0 3 4 -1

5 3 -1 -1

2 2 3 4

0 0 1 1

0 1 0 1

0.1 1 2 1

0 1 1 1

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
- Mainly for 2 reasons:
 - Text is not “naturally” represented by numbers
 - Text can be of unlimited length
 - And it cannot be “naturally” reduced to a fixed size

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
 - Mainly for 2 reasons:
 - Text is not “naturally” represented by numbers
 - Text can be of unlimited length
 - And it cannot be “naturally” reduced to a fixed size
- Actually, Word Embeddings works well for that

Processing sequences

- We can now transform our text into a sequence of vectors.
- But this sequence could have any length
- For images, having input images of different sizes is not really a big problem, because we it is easy to “resize” the image

Processing sequences

- We can now transform our text into a sequence of vectors.
- But this sequence could have any length
- For images, having input images of different sizes is not really a big problem, because we it is easy to “resize” the image

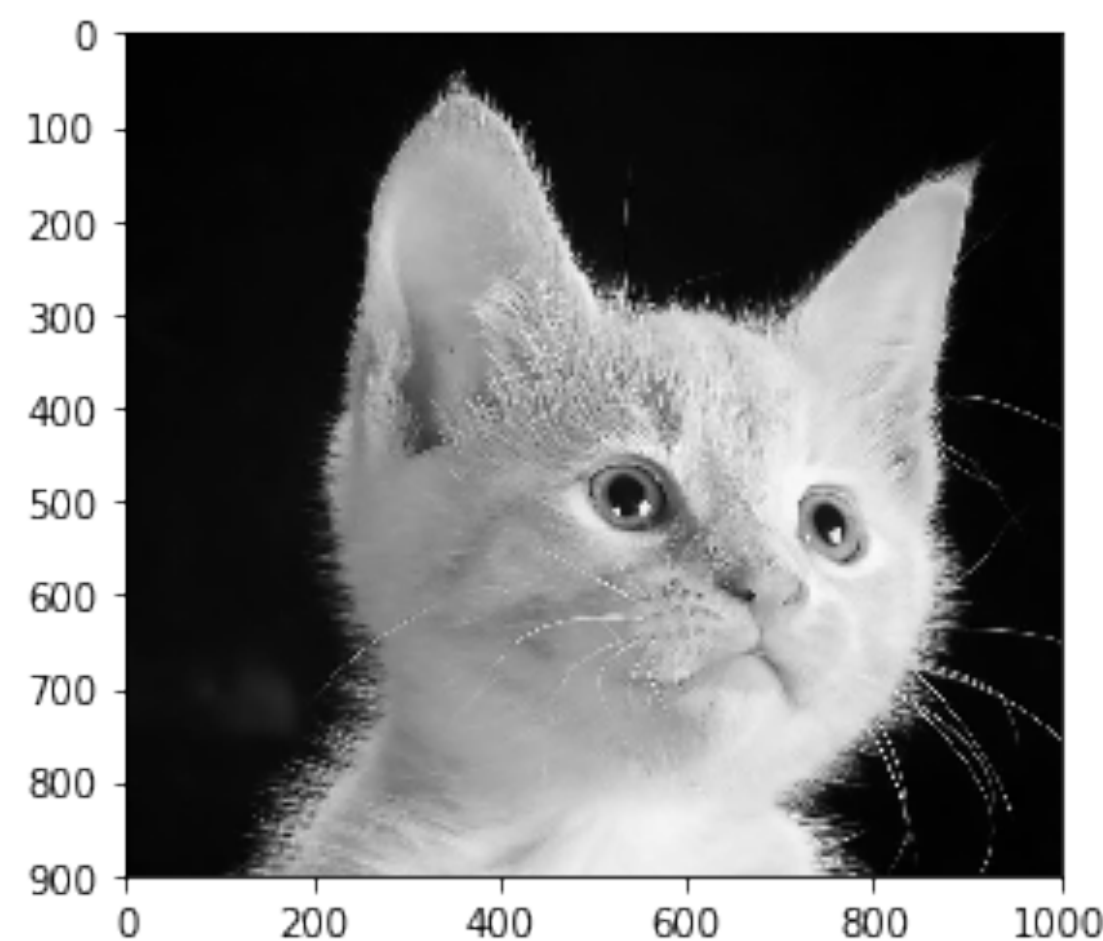
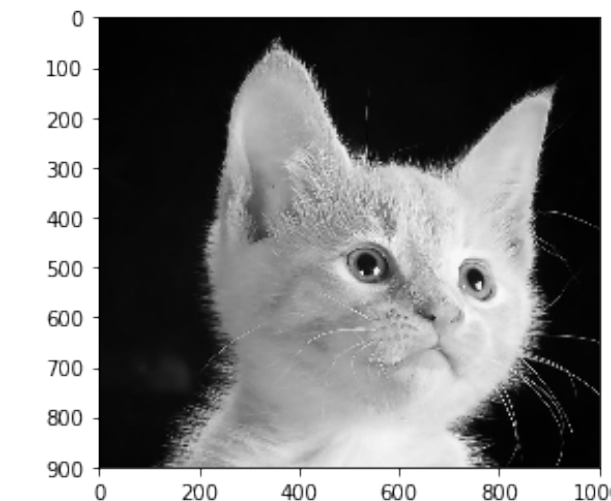


Image of a cat

For each 4 pixels, only
keep one



Smaller Image of a cat

Processing sequences

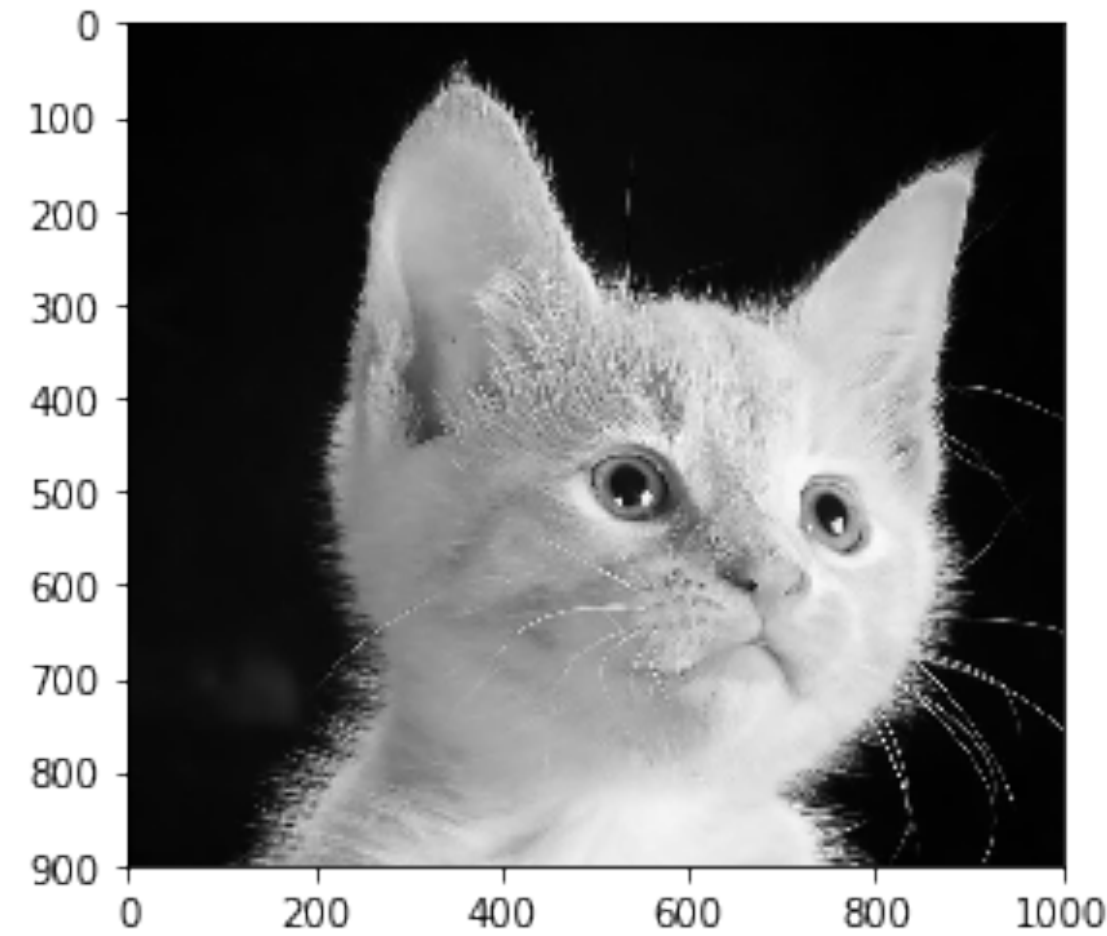
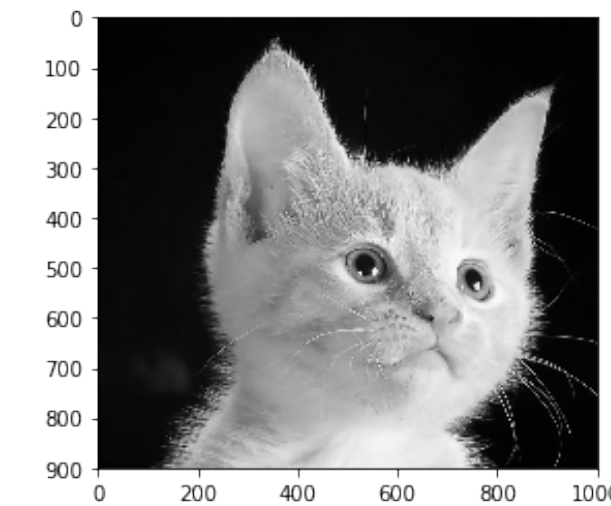


Image of a cat

For each 4 pixels, only
keep one



Smaller Image of a cat

Meaningful text

Since Disney is incompetent of coming up with new ideas, and must resort to using older stories they did years ago, they certainly better live up to what they're doing. This does not happen with Mary Poppins Returns.

For each 4 words, only
keep one

Meaningless text

Since of new resort stories ago, live they're not Poppins

Processing Sequences

- In an image, individual pixels do not have strong meaning by themselves
 - Changing one pixel in an image will not change the meaning of the image
- In a text, each word influence the whole meaning

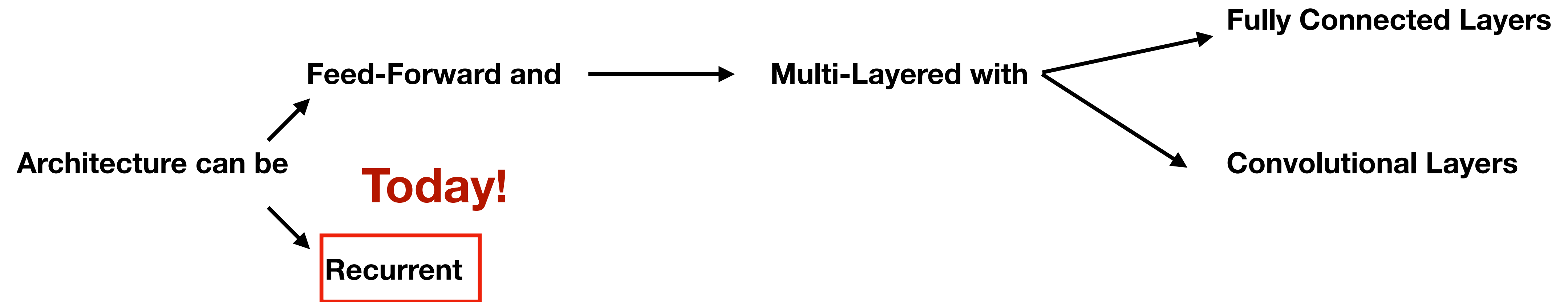
I do like you

I do not like you

- Because of that, we cannot easily manipulate the size of our input text like we can manipulate the size of input images
- Therefore, it is the network that has to be flexible

Neural Network Architectures

- In short:

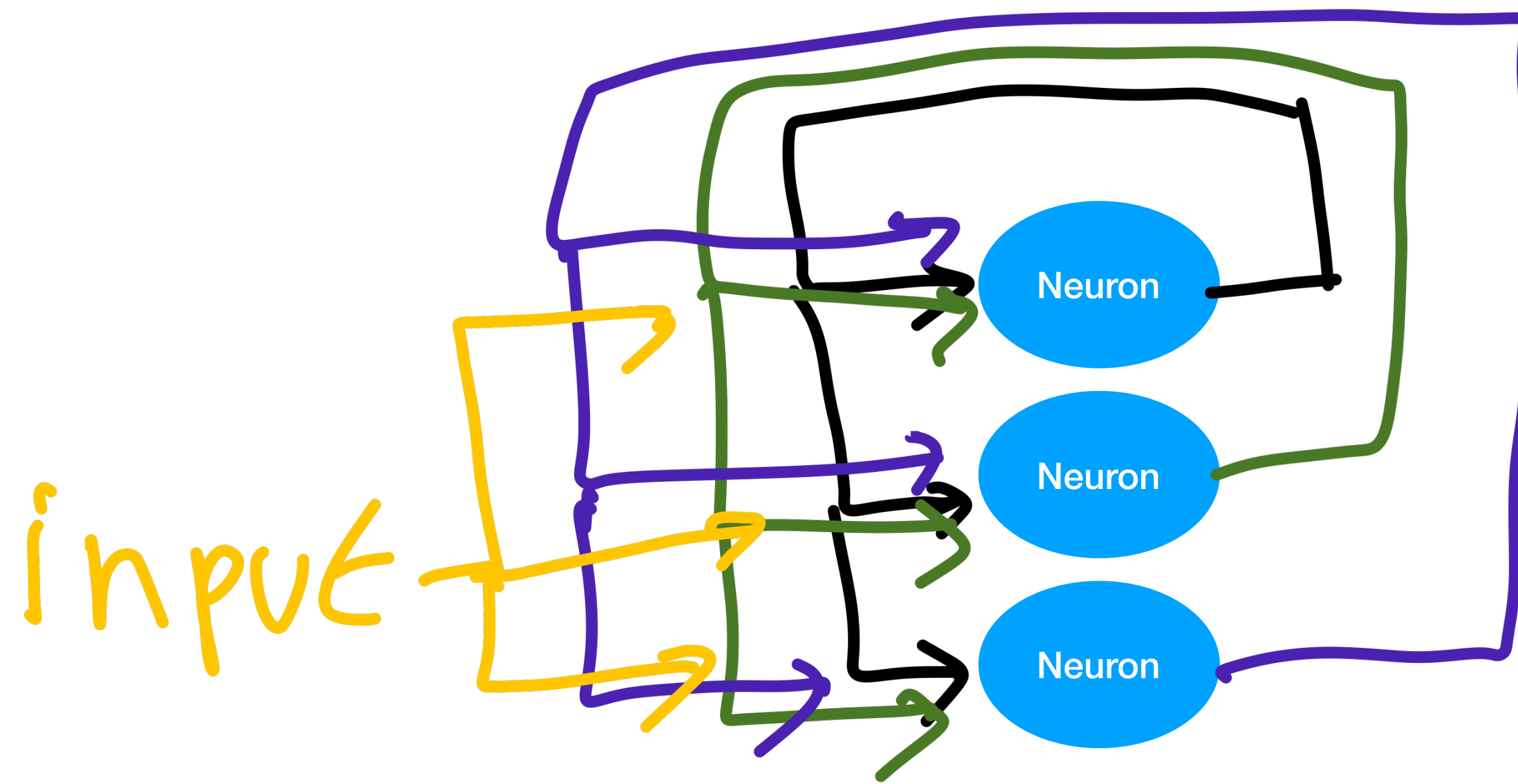


Recurrent Neural Network

- So far, we have only seen “Feed-Forward” Architectures
 - ie. Information is processed from left to right

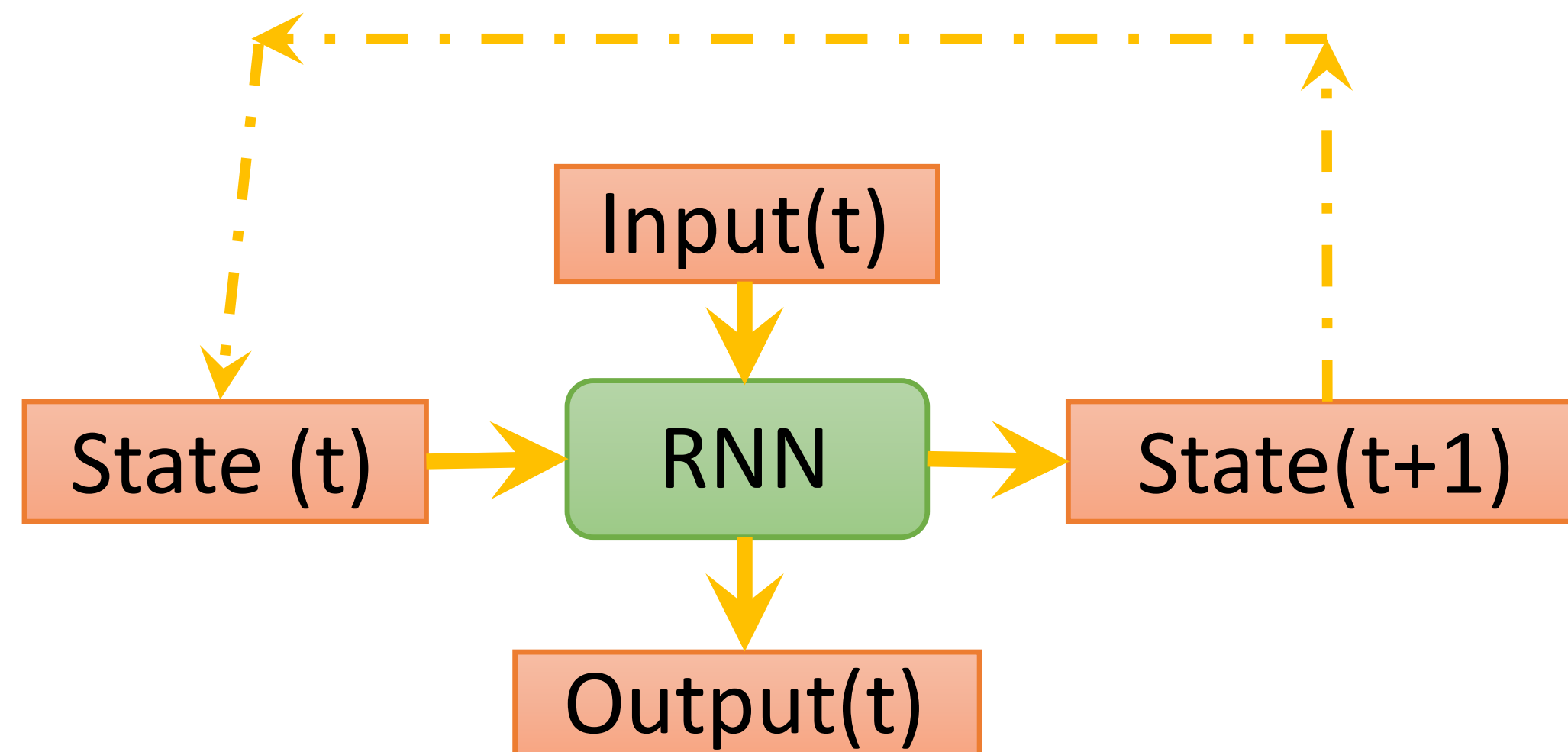
Recurrent Layer

- The idea of a recurrent Neural Network Layer is that, on top of its normal input, neurons in the layer will receive their **output** as **additional input**



Recurrent Layers

- In practice, we use more complicated architecture
- But the general idea is that part of the output of a Recurrent Neural Network (RNN) layer is used as additional input
- The part of the output that is redirected to the input will be seen as a “state” of the RNN

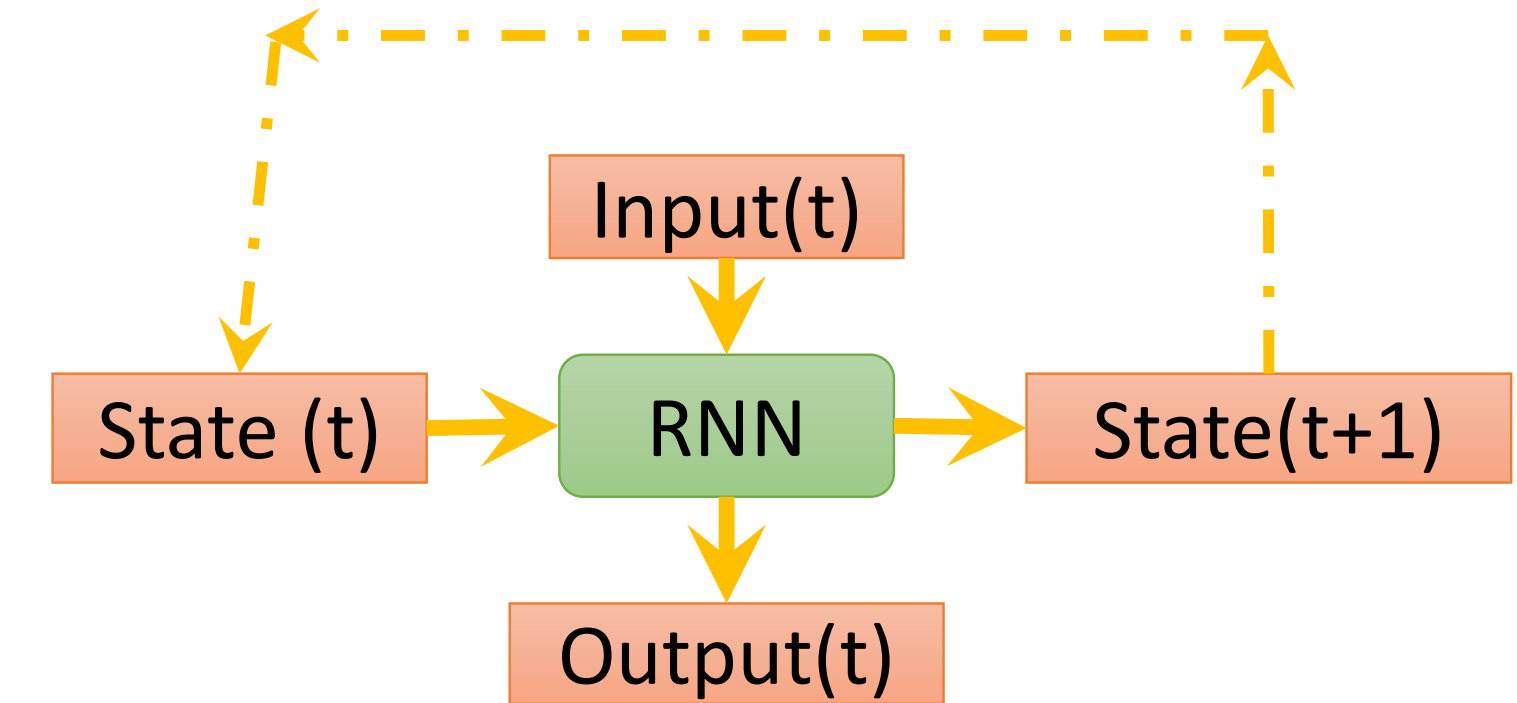
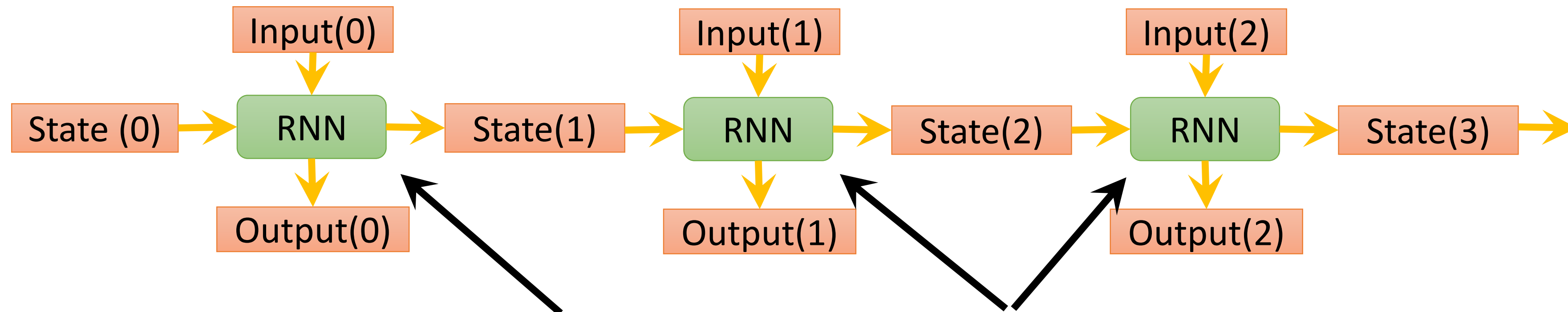


Recurrent Neural Networks

- These Recurrent architecture are very useful for processing sequences
- Each element of the sequence is successively given as input to the RNN
- Because of the “state” output going back to the input, the RNN does not “forget” the previous input
- This way, after being given the whole input, the final state of the RNN contains information about the whole sequence

Recurrent Network

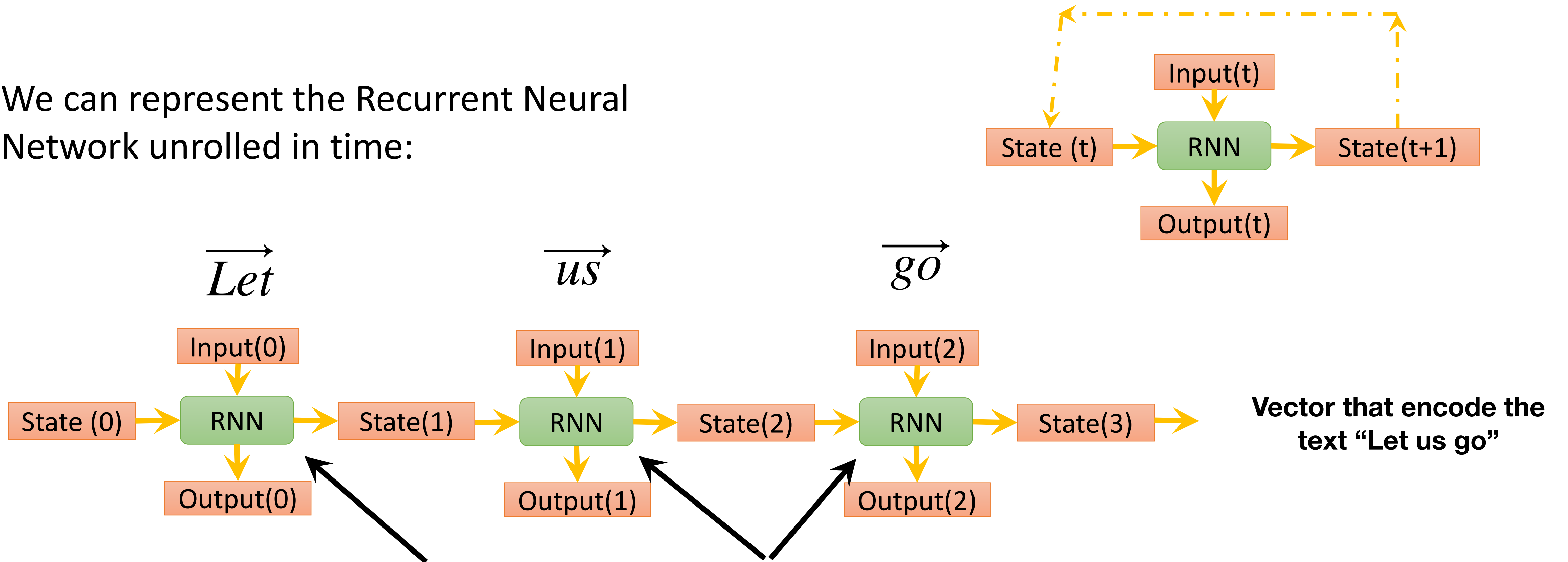
We can represent the Recurrent Neural Network unrolled in time:



Represented as different cells, but actually using the same parameters

Recurrent Network

We can represent the Recurrent Neural Network unrolled in time:



Represented as different cells, but actually using the same parameters

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
 - Mainly for 2 reasons:
 - Text is not “naturally” represented by numbers
 - Text can be of unlimited length
 - And it cannot be “naturally” reduced to a fixed size
- Actually, Word Embeddings works well for that

How to process text?

- Processing text with Neural Networks is actually more complex than processing images
- Mainly for 2 reasons:
 - Text is not “naturally” represented by numbers **Actually, Word Embeddings works well for that**
 - Text can be of unlimited length **Recurrent Neural Networks can process sequences of any size**
 - And it cannot be “naturally” reduced to a fixed size

Text Classifiers

- We now know how to process text as input, and we could easily build a “Sentiment classifier”, for example

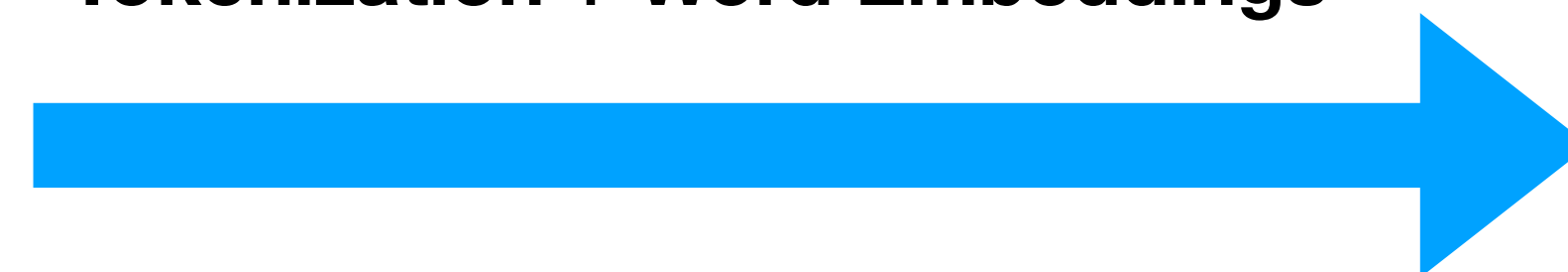
Imdb comment

Since Disney is incompetent of coming up with new ideas, and must resort to using older stories they did years ago, they certainly better live up to what they're doing. This does not happen with Mary Poppins Returns.

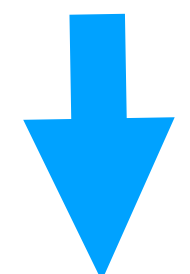
This user thinks the movie is bad or good?

Bad

Tokenization + Word Embeddings



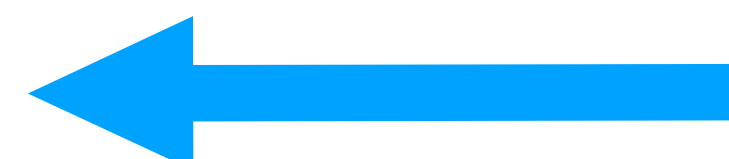
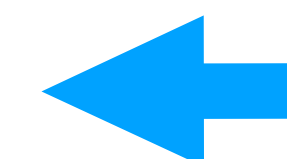
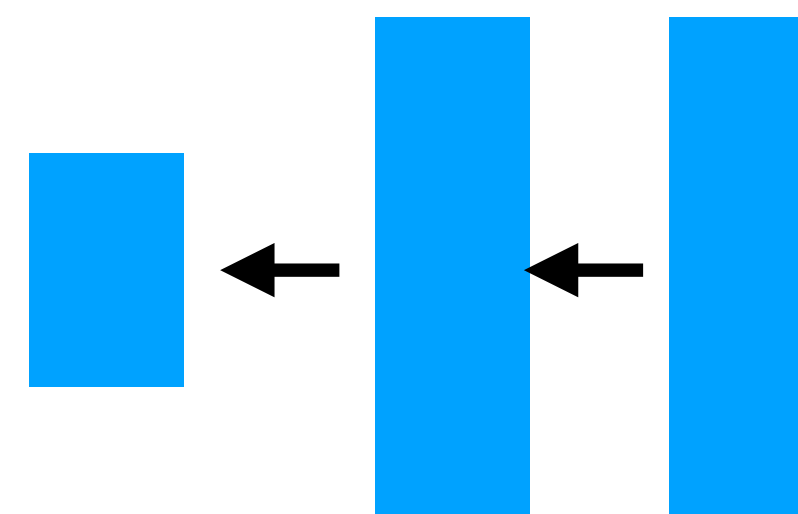
Sequence of vectors



RNN

Final State vector

Binary Classifier with Fully Connected Layers



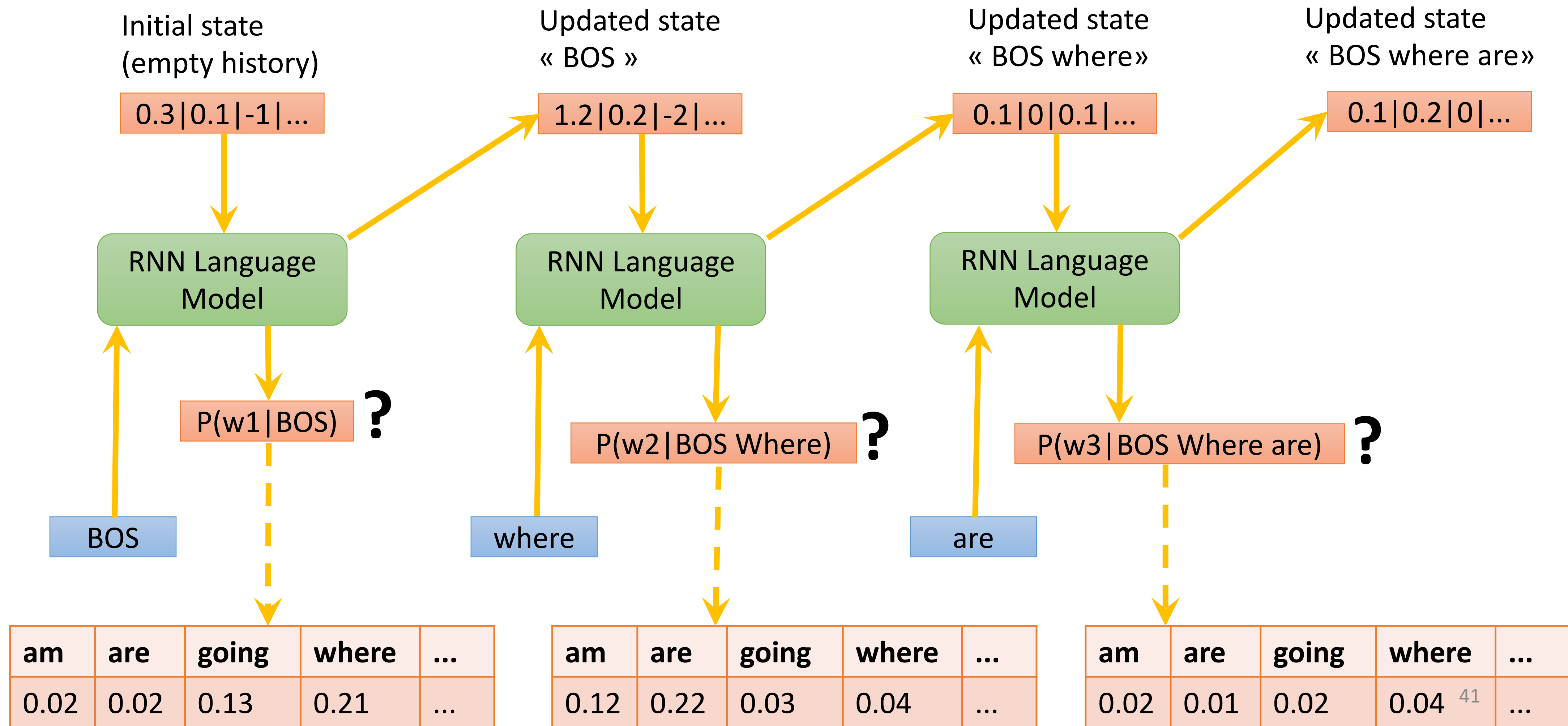
Generating Text

- We have seen how to use text as input
- What about text as output?
- One thing that is a bit easier with text than image is generation

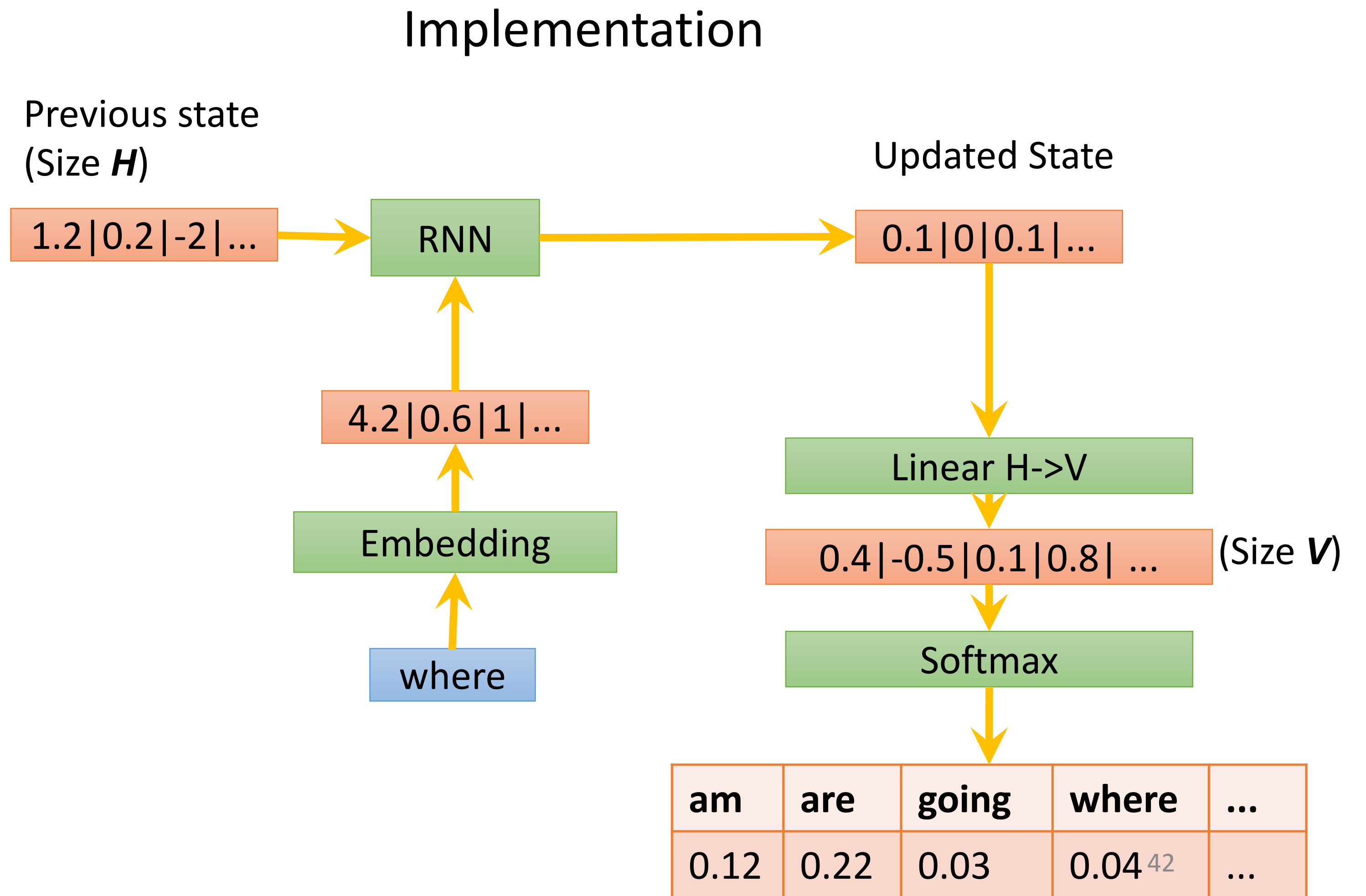
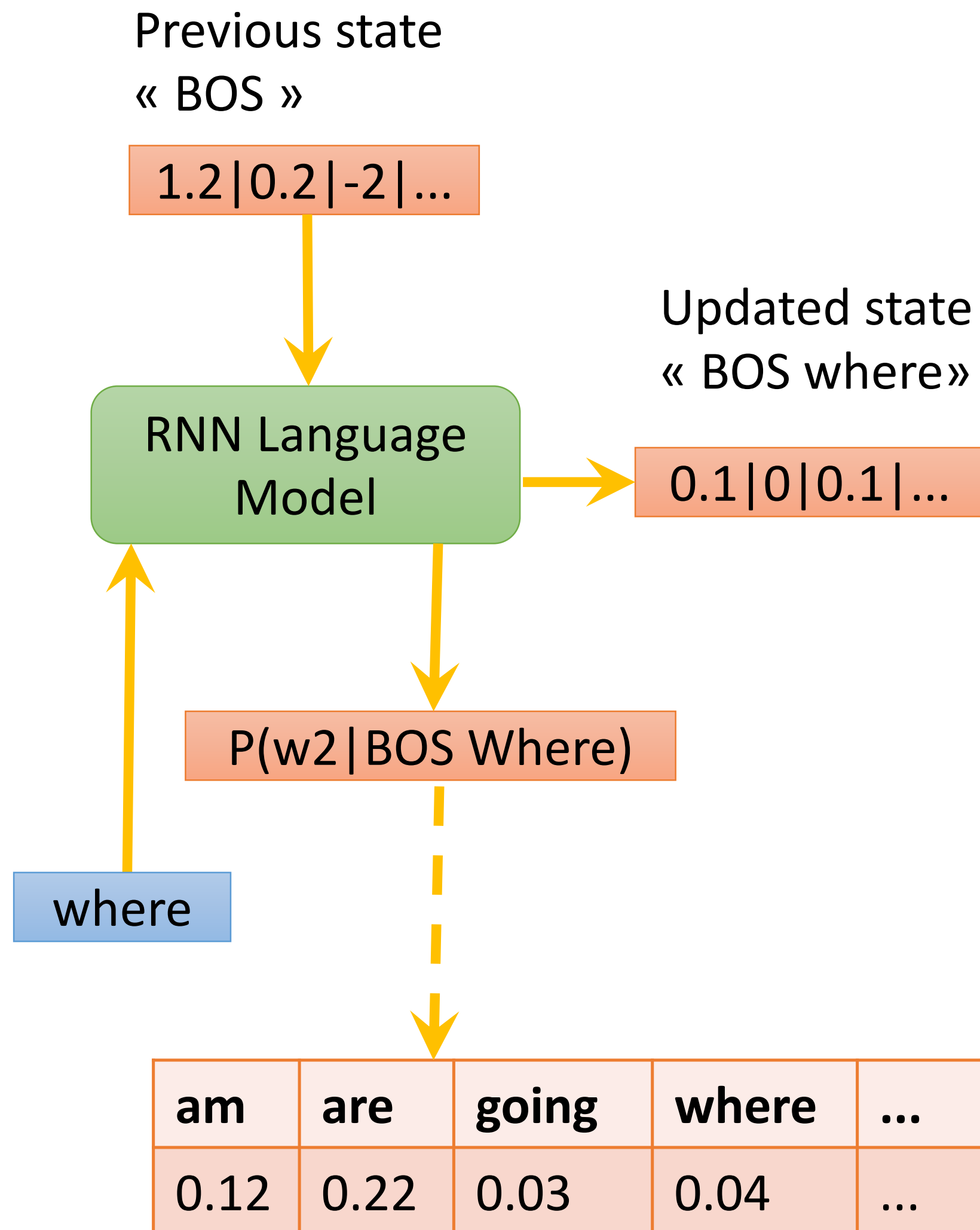
RNN Language Models

- A Language Model is a Model that can **predict the probability of the next word** in a sequence given the previous words
 - « Where is the ... » ? $P(\text{cat})=0.3$, $P(\text{sky})= 0.02$, ...
- How we do that with Neural Networks:
 - A **state representing the *history*** is maintained
 - here, history means the words of the translation that have been **already generated**
 - This state is simply a vector
 - The Language Model compute:
 - the **probably of the next word** given the history: $P(w_3 \mid w_0 w_1 w_2)$
 - the **updated history state** as we add more words to the translation

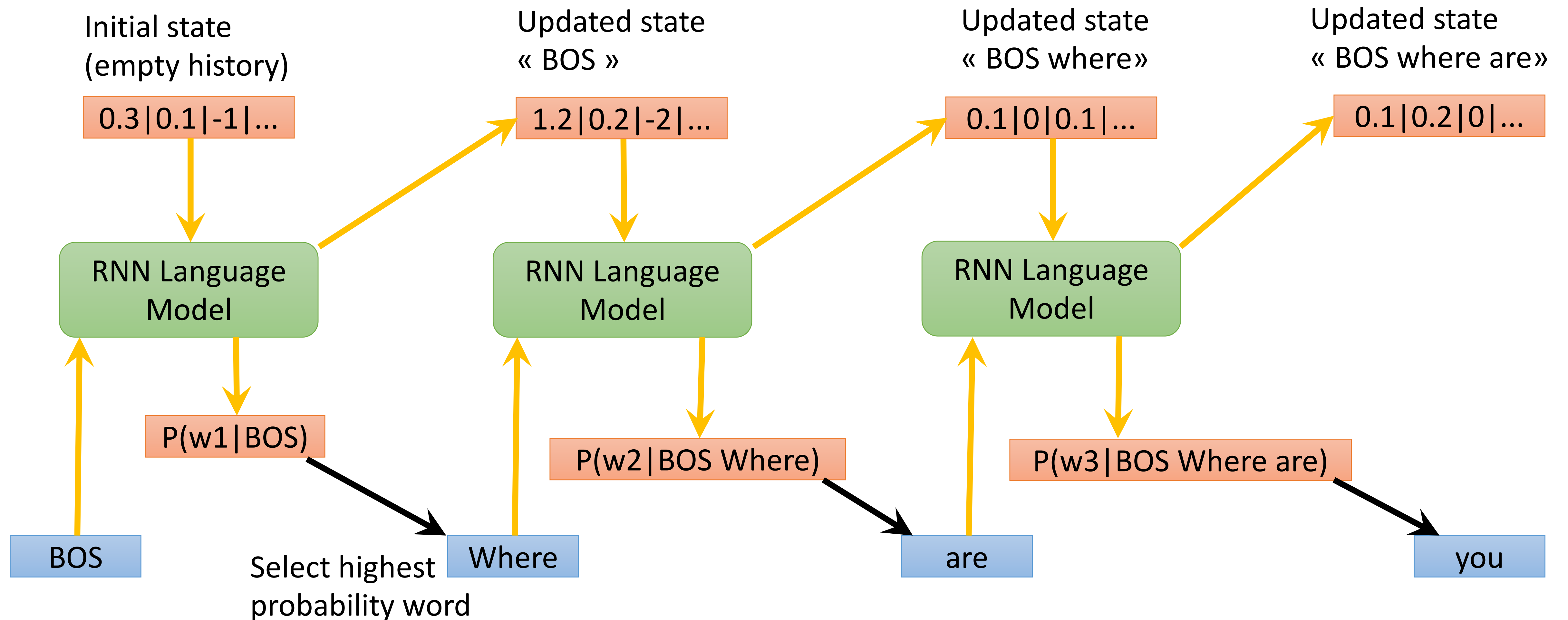
RNN Language Model



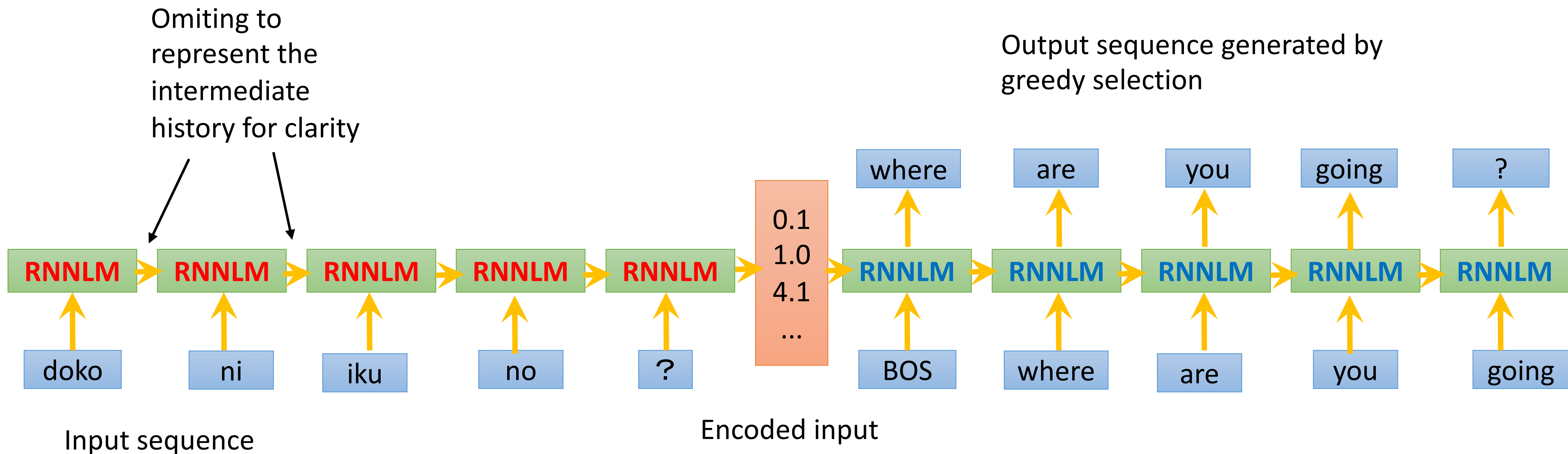
Implementation of the RNN LM



Generating Sequences: Greedy Decoding



Neural MT as a Neural Language Model



Natural Language Processing

- That will be all.
- If we have time, now, let us have a look at some advanced topics.