—Neural modeling of language—

ÿ What instruments for a connectionist modeling of language?
What relationship with the semantic paradigm?
distributional/vectoral for natural languages?

8a.1

1

Þ Semantic representation

q **Principle of compositionality** (recap)

ÿ the meaning of an expression is function
• the meaning of its subexpressions and • the way in which they are combined

q Cognitive motivation

ÿ finite means (mind-brain) can deal with object potentially infinite (language)

q Technological orientation

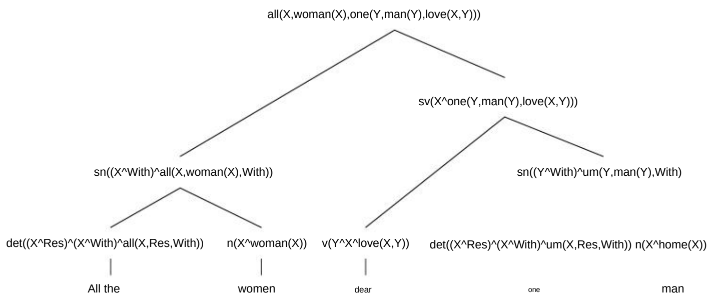
ÿ assigning semantic representations to words, through rules appropriate for their combination, the representation is obtained semantics for any sentence...

ÿ ...and from this representation we can build practical applications

8a.2

2

Inferential semantics: logical repr. (recap)



8a.3

3

Distributional semantics: vector repr. (recap)

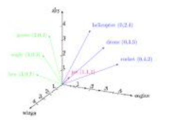
q **Lexical semantics**

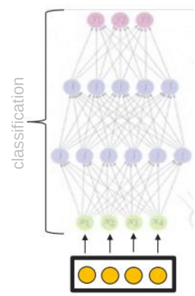
ÿ meaning of the word represented by a vector

ÿ vector (fixed size) of **high dimensionality** (eg 512)

ÿ vector **condenses information** about the respective distribution, ie co-occurrence with other words

ÿ suitable as **an input for networks neuronal**





8a.4

4

Sentence and textual semantics 1/2

q **What vector representation of the sentence/text?**

ÿ meaning of the text represented by vector: **what information** will be condensed by the respective vector?

ÿ how to deal with arbitrary and very varied text lengths: **the same** vector size for any text length?

ÿ compositionality principle: how to **combine** lexical vectors?

8a.5

5

Sentence and textual semantics 2/2

q **What neural architecture?**

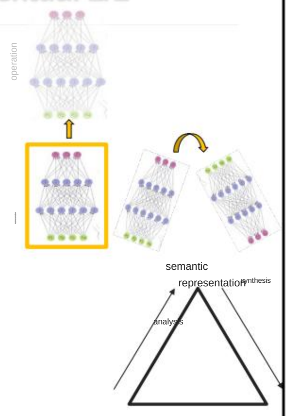
ÿ how to obtain suitable representation as input to neural networks?

- A: with **other neural (sub-)networks!**

ÿ how to get desired output?

- A: analogy with triangulation, but the training is joint **("end-to-end")**

ÿ **with what network architecture?** question motivating for this classroom



8a.6

6

Þ Bag of words (BOW)

q Concatenation

ÿ v = v1v2... = [v11,...,v1m,v21, ... ,v2m,...]

Just concastinate each vector i think

q Soma

ÿ v = v1 + v2 + ... =

[v11+v21+..., ... ,v1m+v2m+...]

q Average You sum the word vectors component-wise, resulting in a single vector of the same size as a word vector.

ÿ v = aver(v1,v2,...,vn) =

[v11+v21+.../n, ... ,v1n+v2n+.../n]

q Maximum Like the sum, but normalized by the number of words.

ÿ v = max(v1,v2,...) =

[max(v11,v21,...),...max(v1m,v2m,...)]

Take the maximum value in each dimension across all word vectors. This can emphasize salient features.

q "Zero degree" of learning

ÿ but in practice with interesting results, eg for similarity semantics, given the (little) sophistication required

8a.7

7

Þ Convolution ("convolution")

q **Rede Neuronal Convolucional** ("CNN")

ÿ the same network receives successive segments of *k* words

- another perspective: sliding window over k-grams

ÿ for each k-gram it generates a vector

A CNN will look at small groups of words at a time, for example, 2 or 3 words at once.

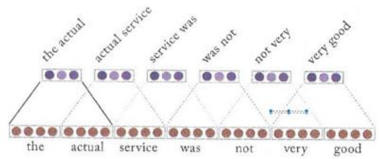
These small groups are called k-grams (k = number of words in the group).

q **Aggregation** ("pooling")

ÿ vector representations of k-grams are aggregated into a vector

- max or average of each dimension of the vectors

We combine them into a single vector to represent the whole sentence.



8a.8

8

Convolutional architectures

q Variants

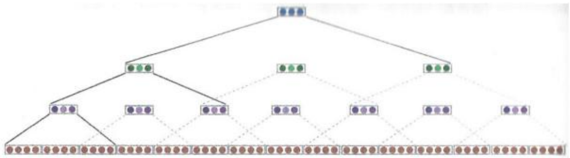
ÿ size of the k-gram, typically bi- and tri-gram

ÿ stride the window by 1 word or more

ÿ hierarchical convolution with 1 or more levels

ÿ parameter-tying between levels through the same network trained for all levels

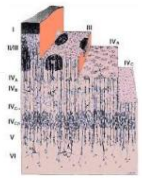
ÿ skip-connections that link non-immediate levels



8a.9

9

Rationale



q Inspiration

ÿ analogy with visual cortex ÿ
architecture successful in image recognition ÿ each pixel
window helps to detect different elements of the image (e.g. edge, nose, etc.)

q **Progress towards BOW**

ÿ arbitrary sentence size: **vector of the same dimension for any text**

ÿ **feature extractor:** able to identify more local aspects informative for the prediction in question

(eg words that make the sentence express the most positive/negative feeling)

8a.10

10

Shortcomings 1/2

q **Long-distance dependencies**

ÿ ability to extract localized features, but difficulty increasing in capturing linguistic dependencies of increasing distance

ÿ e,g, *The **cats_i** sleep on the **pillows_k.** Mary told us that the Pedro asked to wash it in the washing machine.*

q **Transduction**

ÿ ability to deal with arbitrary length of sentences/texts, but difficulty with tasks with textual output

ÿ eg continuation, translation, summarization, dialogue, ...

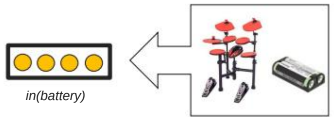
8a.11

11

Insufficiencies 2/2

q **Words are not disambiguated**

ÿ e.g., *Pedro bought an* electric ***battery .*** *To listen to it on wireless headphones without disturbing your neighbors, you just need to charge* their ***battery .***



ÿ undesirably, each word has a single representation (same vector) for all its meanings/acceptances

Does the disambiguation problem lead to sematnic blur? Or is it a seperate issue?

q **Semantic "Blur"**

ÿ insufficiently "defined" contribution to the compositionality of the semantic representation (i.e. vector) of sentences ÿ undesirably, any vectors (of any words in the text) combine in the same way (regardless of their role/position in the sentence)

12

words are lexically ambiguous (recap)


WordNet Search - 3.1
- WordNet home page - Glossary - Help

Word to search for: kite Search WordNet

Display Options: (Select option to change) Change
Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) **kite** (a bank check that has been fraudulently altered to increase its face value)
- S: (n) **kite** (a bank check drawn on insufficient funds at another bank in order to take advantage of the float)
- S: (n) **kite** (plaything consisting of a light frame covered with tissue paper; flown in wind at end of a string)
- S: (n) **kite** (any of several small graceful hawks of the family Accipitridae having long pointed wings and feeding on insects and small animals)

Verb

- S: (v) **kite** (increase the amount (of a check) fraudulently) *"He kited many checks"*
- S: (v) **kite** (get credit or money by using a bad check) *"The businessman kited millions of dollars"*
- S: (v) **kite** (soar or fly like a kite) *"The pilot kited for a long time over the mountains"*
- S: (v) **kite** (fly a kite) *"Kids were kiting in the park"; "They kited the Red Dragon model"*

Þ Recurrence

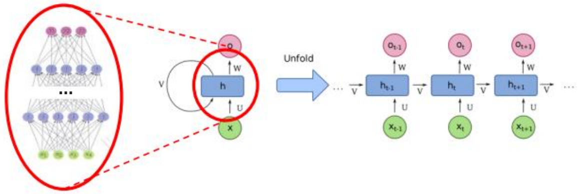
q **Recurrent Neural Networks** ("RNN")

ÿ internal state progressively compounded at each time step

ÿ at each time step, (the vector of) an input word and the state

previous internal contribute to the current internal state and to the output

ÿ in the final step, the internal state is the representation (vector) of the sentence



Compositionality

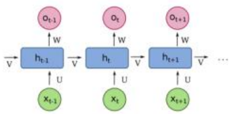
q **Maintain memory of the past (during training)**

ÿ h_t-1.V — contribution of the previous state

ÿ x_t.U — contribution of input at *t* to internal state

ÿ h_t = g(h_t-1.V + x_t.U) — joint contribution

ÿ h_t.W — exit



q **Model the language**

ÿ V, U and W parameters learned / trained networks

ÿ training task: continuation task: *Peter said <cont?>*

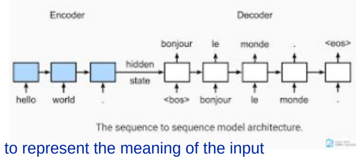
ÿ **learned model: language model (probabilistic)**

seq-to-seq

q **Transduction (during inference)**

ÿ eg translation, summarization, dialogue,...

ÿ final state of the encoder used as the basis for generating the output



The sequence to sequence model architecture.

q **Encoder**

ÿ RNN trained as a language model used to represent the meaning of the input

q **Decoder** ("decoder")

ÿ second RNN trained to generate output text

ÿ lexicon-sized output layer: next word predicted by the most active unit

Resilience

q **Transduction OK**

ÿ eg continuation, translation, summarization, dialogue

q **Long-distance dependencies OK**

ÿ e,g, *The **cats_i** sleep on the **pillows_k.** Mary told us that the Pedro **-as_k** washed it in the washing machine.*

ÿ same parameters/networks learned over each step temporal/input word allows weighing and accommodating non-local relations

8a.17

17

Shortcomings 1/3

q **Gradients**

ÿ learning/training involves many computations of the gradients and their factorization: with approximation values, errors accumulate

ÿ **creates bias to capture short-distance relationships**

q **Exploding** gradients

ÿ when many very large values > 1 occur ÿ **solution:** gradient clipping to lower the value

8a.18

18

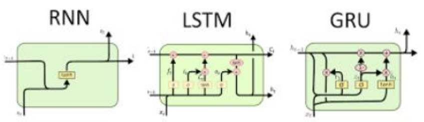
Shortcomings 2/3

q **Exploding** gradients q Vanishing **gradients**

ÿ when many very small values < 1 occur

ÿ **solution:** control the values that are brought from the past with gated cells: most popular proposals (advanced topic):

ÿ LSTM - Long Short Term Memory ÿ GRU – Gated Recurrent Unit



8a.19

19

I think this is word orderer dismabigatuon, or something with the same word not beeing able to meadn different things in different instances

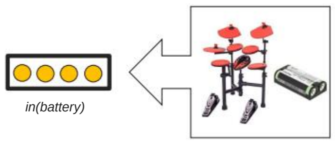
Shortcomings 3/3

q **There is still no WSD**

ÿ undesirably, each word continues to have a single representation (same vector) for all its meanings/meanings: "blur" undefined semantic

ÿ there is no word sense disambiguation (WSD: "word sense"). disambiguation") depending on the context of occurrence

ÿ e,g, *Pedro bought an* electric ***drum .** To listen to it in the wireless headphones, without disturbing your neighbors, you just need to charge their **battery***



8a.20

20

Þ Attention

q **Contextualized lexical disambiguation**

ÿ a word can have several possible meanings/meanings (in the lexicon)

ÿ typically, one meaning is conveyed (in one occurrence)

ÿ one of the meanings is favored, the others are ignored (by context)


WordNet Search - 3.1
- WordNet home page - Glossary - Help

Word to search for: kite Search WordNet

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations

Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) **kite** (a bank check that has been fraudulently altered to increase its face value)
- S: (n) **kite** (a bank check drawn on insufficient funds at another bank in order to take advantage of the float)
- S: (n) **kite** (plaything consisting of a light frame covered with tissue paper; flown in wind at end of a string)
- S: (n) **kite** (any of several small graceful hawks of the family Accipitridae having long pointed wings and feeding on insects and small animals)

Verb

- S: (v) **kite** (increase the amount (of a check) fraudulently) *"He kited many checks"*
- S: (v) **kite** (get credit or money by using a bad check) *"The businessman kited millions of dollars"*
- S: (v) **kite** (soar or fly like a kite) *"The pilot kited for a long time over the mountains"*
- S: (v) **kite** (fly a kite) *"Kids were kiting in the park"; "They kited the Red Dragon model"*

8a.21

21

Þ Attention

q **Contextualized lexical disambiguation**

ÿ a word can have several possible meanings/meanings (in the lexicon)

ÿ typically, one meaning is conveyed (in one occurrence)

ÿ one of the meanings is favored, the others are ignored (by context)

q **Alternative: Contextualized lexical representation**

ÿ semantic "blur" of a word is made more definite

ÿ **components of your vector are weighted according to the vectors of the other words in the context (and the task to be learned)** ÿ How?

a solution in the following slides...

8a.22

22

dot product (recap)

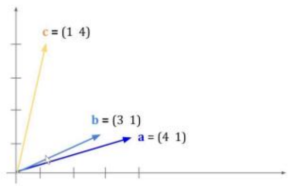
q **Scalar product of vectors** ("dot product")

ÿ z = v1.v2 = v11.v21 + ... + v1m.v2m

ÿ exs.:

ÿ a.b = 13 = 3x4+1x1

ÿ a.c = 8 = 1x4+4x1



ÿ **highest result, closest vectors**

ÿ recap: word embeddings: closer vectors, higher similarity semantics

8a.23

23

Attention to context 1/3

1. Obtain semantic representation

ÿ vectorize("embedding"):

vectors (dimension m) for each of the n words wd1...wdn of a given sentence

ÿ **v1** = [v11,...,v1m]

ÿ ...

ÿ **vn** = [v1n,...,v1m]

2. Score semantic proximity

ÿ **scalar product** of each word for each word (between vectors vi) ÿ **s1** = [s11,...,s1n] = [v1.v1,...,v1.vn]

ÿ ...

ÿ **sn** = [sn1,...,snn]

8a.24

24

Attention to context 2/3

3. Obtain weights with softmax normalization

ÿ **normalize:** each component lies within [0,1] and sums to 1

ÿ **wg1** = softmax(s1) = [wg11, ..., wg1n]

ÿ ...

ÿ **wgn** = softmax(sn) = [wgn1,...,wgnn]

4. Add heavy starting vectors

ÿ contextualized representation of each wdi: sum of the vectors of departure of the wdi words in the context after being weighed by the weight vector of each wdi

ÿ **y1** = wg11.v1+...+wg1n.vn

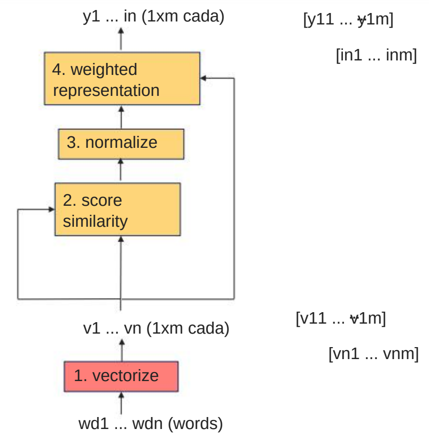
ÿ ...

ÿ **in** = wgn1.v1+...+wgnn.vn

8a.25

25

Attention to context 3/3



8a.26

26

Attention to task 1/3

q OK: Context-sensitive weighting wd1...wdn

ÿ based on the dot product

ÿ lexical vectors

ÿ terminology: self-attention: only input information taken into consideration

q **BUT ALSO: Task-based weighting**

ÿ eg summarization, translation,...

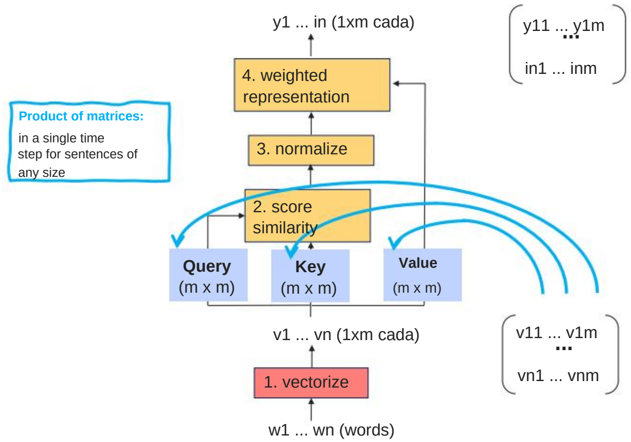
ÿ based on the matrix product

ÿ **matrices/network weights learned during training for the task**

8a.27

27

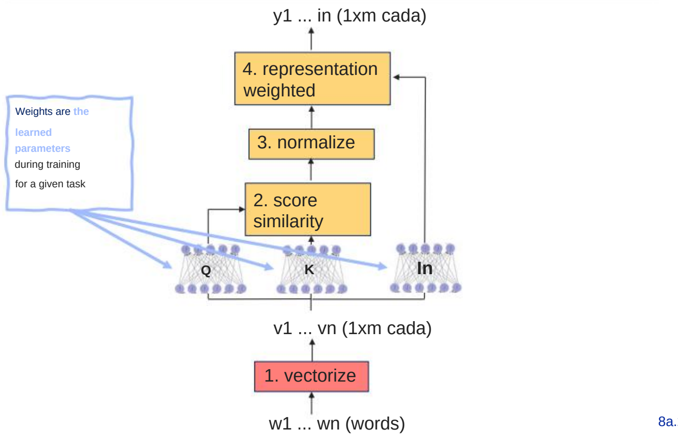
Attention to task 2/3



8a.28

28

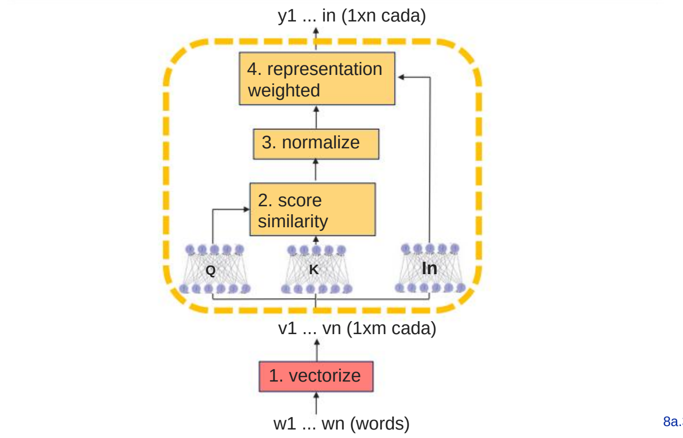
Attention to task 3/3



8a.29

29

Attention head



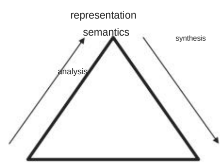
8a.30

30

Þ Transformer Architecture

q **Triangulation**

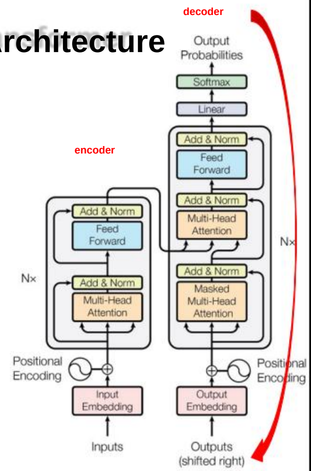
- ÿ based on the triangulation of meaning, with encoder-decoder



- ÿ in inference: decoder

  concatenates its exit to its next

  Prohibited



8a.31

31

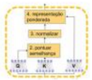
Transformer 2/3

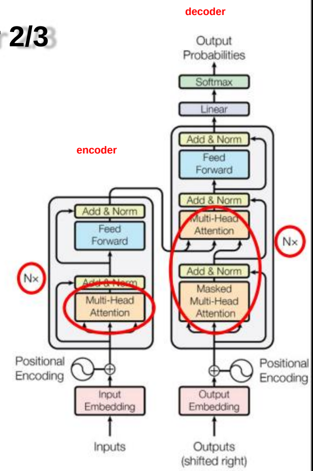
q **Triangulation**

q **Percetron**

- ÿ multi-layer perceptron based fully connected

- ÿ enriched with attention sub-layers





8a.32

32

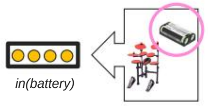
Transformer 3/3

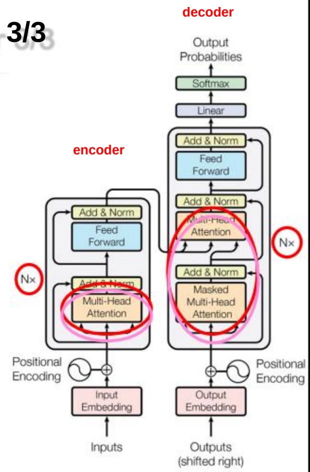
q **Triangulation**

q **Percetron**

q **Disambiguation**

ÿ based on "defined" semantic representations, i.e. contextualized, i.e. reciprocally disambiguated





8a.33

33

Key innovations

q **parallelism I**

ÿ **input words processed in parallel, in a single self-attention time step (GPU)**

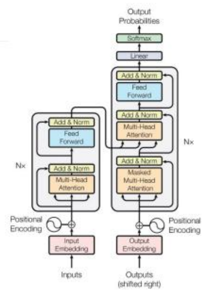
ÿ in training: codif/desc; in test: codif

ÿ Overcoming explosive gradients of RNNs

q **parallelism II**

ÿ **contextualized word representations do not need to be collapsed and diluted into a semantic representation of the sentence (before the last output layer)**

ÿ overcoming the problem of RNNs in long-range forgetting



8a.34

34

Key adaptations

q **relative word order**

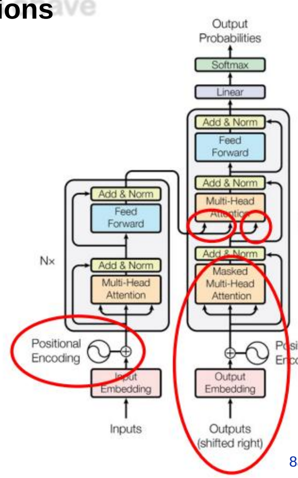
ÿ reassured with vectors that represent the positions in the sentence ("positional embeddings"), to be added to the word vectors

q **transduction**

ÿ parallel decoder trained with masked: for each word, hide the words after that

what **attention**

ÿ no decoder: V receives representation of previously output words, Q and K receive representation of all input words by the encoder



8a.35

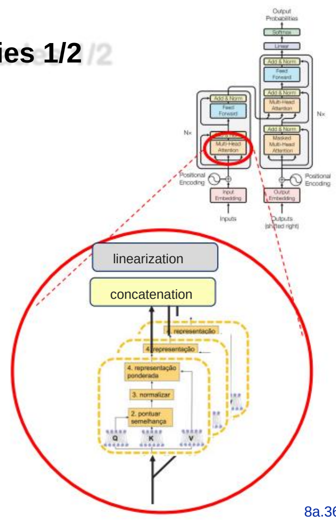
35

Technicalities 1/2

q **Multi-head attention**

ÿ At each layer, the input (set of words) is passed through several heads in parallel, outputs are concatenated and linearized

ÿ each head learns differently regularities and linguistic relations of the context and the task



8a.36

36

Technicalities 2/2

q **Residual connections**

ÿ bypass each sub-layer to
add your input to your output

ÿ facilitate backpropagation during
training

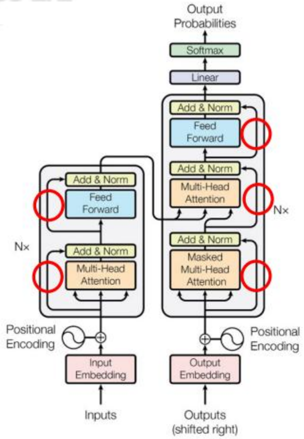
q **Some hyperparameters**

(no original paper from 2017)

ÿ lexical vectors size 512

ÿ 6 layers

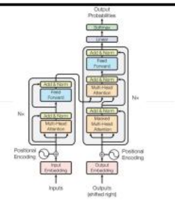
ÿ 8 heads per layer (size 64)



8a.37

37

Reference



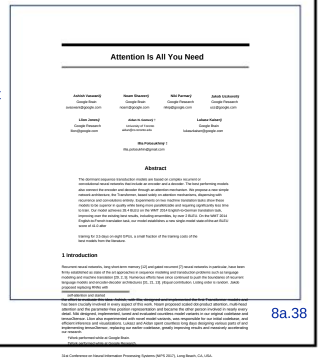
q **Reference**

ÿ Vaswani et al., 2017, *Attention Is All You Need,*
Google Research

q **Mainstream**

ÿ Transformer is the mainstream architecture, with best
performance for a wide range of NLP tasks

ÿ Transformer variants: next lessons


Attention Is All You Need

Abstract

1 Introduction

8a.38

38

- Conclusion -

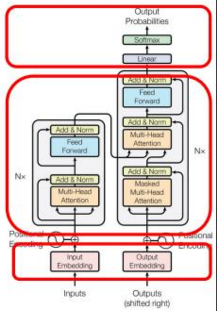
q Index

ÿ Vector representation of texts and semantic triangulation

ÿ Neural network architectures: CNN, RNN

ÿ Attention

ÿ Transformer Architecture



How to get the input word vectors?

How to get the output from the last layer?

How to train a model?

8a.39

39