Language Technologies Institute

Carnegie Mellon University

Multimodal Machine Learning

Lecture 2.1: Unimodal Representations

Louis-Philippe Morency

** Original course co-developed with Tadas Baltrusaitis. Major revision co-developed with Paul Liang. Co-lecturers included Yonatan Bisk, Daniel Fried and Carlos Busso.*

Administrative Stuff

Upcoming Course Assignments

Project preferences (deadline Tuesday 9/1 at 5pm ET)

- Let us know about your project preferences, including datasets, research topics and potential teammates
  - Preferences will be shared with other students (two spreadsheets)
- We will reserve a moment on Thursday 9/3 to help with team matching
- Instructions shared via Canvas (released today at 11am ET)

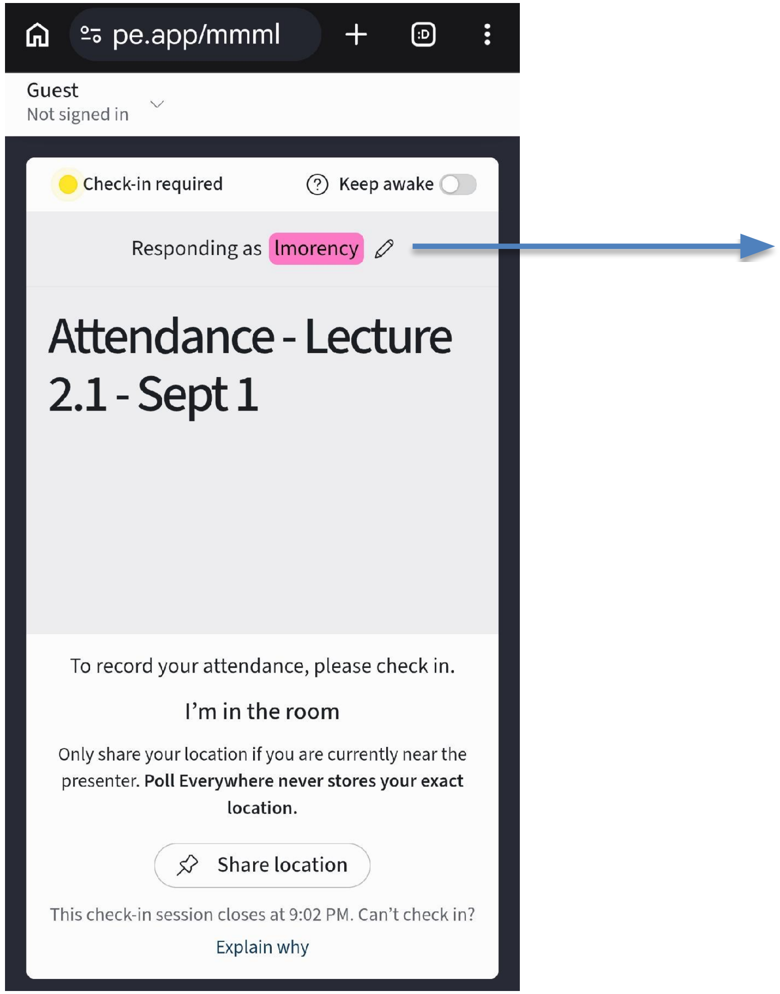
Reading Assignment (due Friday 9/4 at 8pm ET)

- Two papers: MMML paper + a second paper of your choice
- Instructions shared via Canvas (released today at 11am ET)

Lecture Participation (starting today)

- Using your cellphone to confirm attendance
  - We plan to use Poll Everywhere for class participation.

Attendance


pe.app/mmml
Guest
Not signed in
Check-in required
Keep awake
Responding as lmorency
Attendance - Lecture 2.1 - Sept 1
To record your attendance, please check in.
I'm in the room
Only share your location if you are currently near the presenter. Poll Everywhere never stores your exact location.
Share location
This check-in session closes at 9:02 PM. Can't check in?
Explain why

Respond at
pe.app/mmml

Attendance - Lecture 2.1 - Sept 1

0

Presenter instructions

Check-in method

Location only

Location verification
Participants must share their device location to check in. They must be within range of a saved location.

Location

CMU Posner Hall

Set the location you want your audience to check in from. Manage saved locations.

Duration

15 minutes

How long should they have to check in?

Open check-in session now

When presented the check-in session will open automatically.

Language Technologies Institute

Carnegie Mellon University

Multimodal Machine Learning

Lecture 2.1: Unimodal Representations

Louis-Philippe Morency

** Original course co-developed with Tadas Baltrusaitis. Major revision co-developed with Paul Liang. Co-lecturers included Yonatan Bisk, Daniel Fried and Carlos Busso.*

Lecture Objectives

- Examples of sentence modeling tasks
  - Sequence predictions and language models
- Syntax and language structure
  - Phrase-structure and dependency grammars
- Self-attention models (transformers)
  - Position embeddings
- Pre-training and language transformers
  - BERT: Bidirectional Encoder Representations from Transformers

Examples of Sentence Modeling Tasks

Sentence Modeling: Sequence Prediction


Masterful!

By Antony Witheyman - January 12, 2006

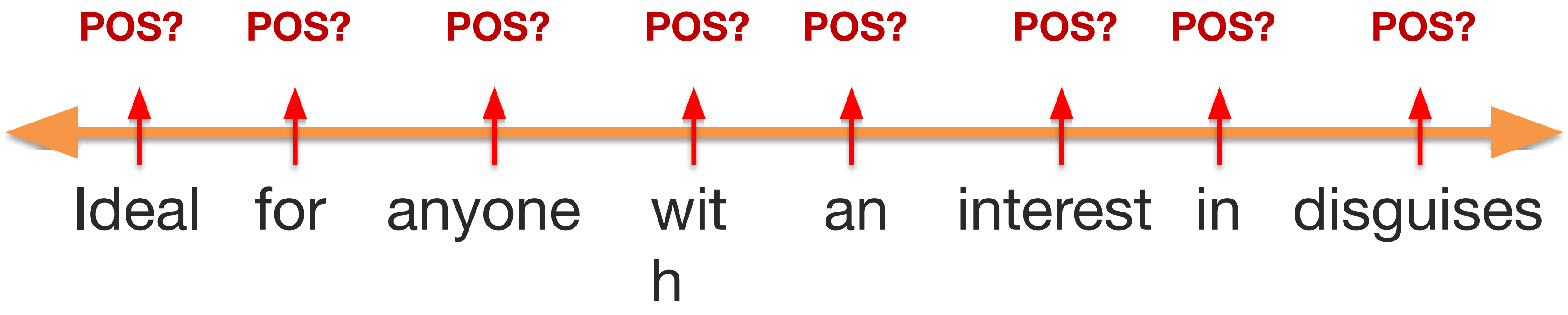
Ideal for anyone with an interest in disguises who likes to see the subject tackled in a humourous manner.

0 of 4 people found this review helpful

Prediction

Part-of-speech ?
(noun, verb, ...)

Sentiment ?
(positive or negative)



Sentence Modeling: Sequence Label Prediction

Masterful!

By Antony Witheyman - January 12, 2006

Ideal for anyone with an interest in disguises who likes to see the subject tackled in a humourous manner.

0 of 4 people found this review helpful

Prediction

Sentiment ?
(positive or negative)

Sentiment label?

Ideal for anyone wit h an interest in disguises

Sentence Modeling: Language Model

Masterful!

By Antony Witheyman - January 12, 2006

Ideal for anyone with an interest in disguises who likes to see the subject tackled in a humourous manner.

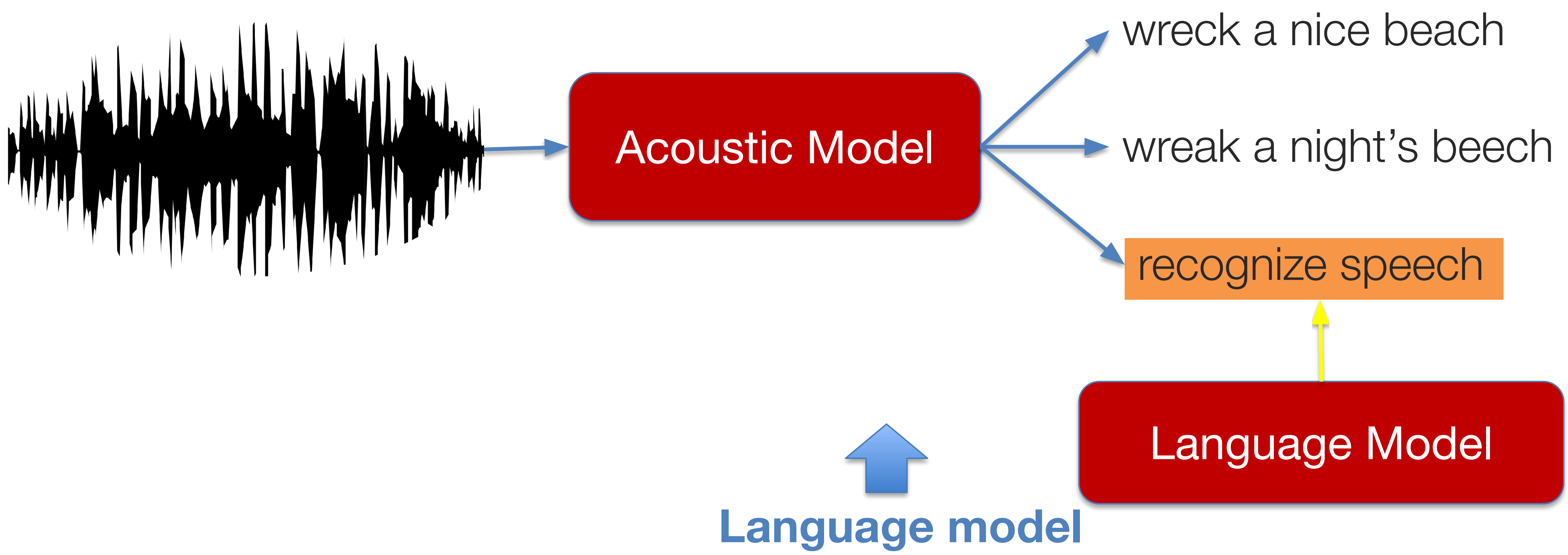
0 of 4 people found this review helpful

Prediction → Next word

Next word?

Ideal for anyone wit h an interest in disguises

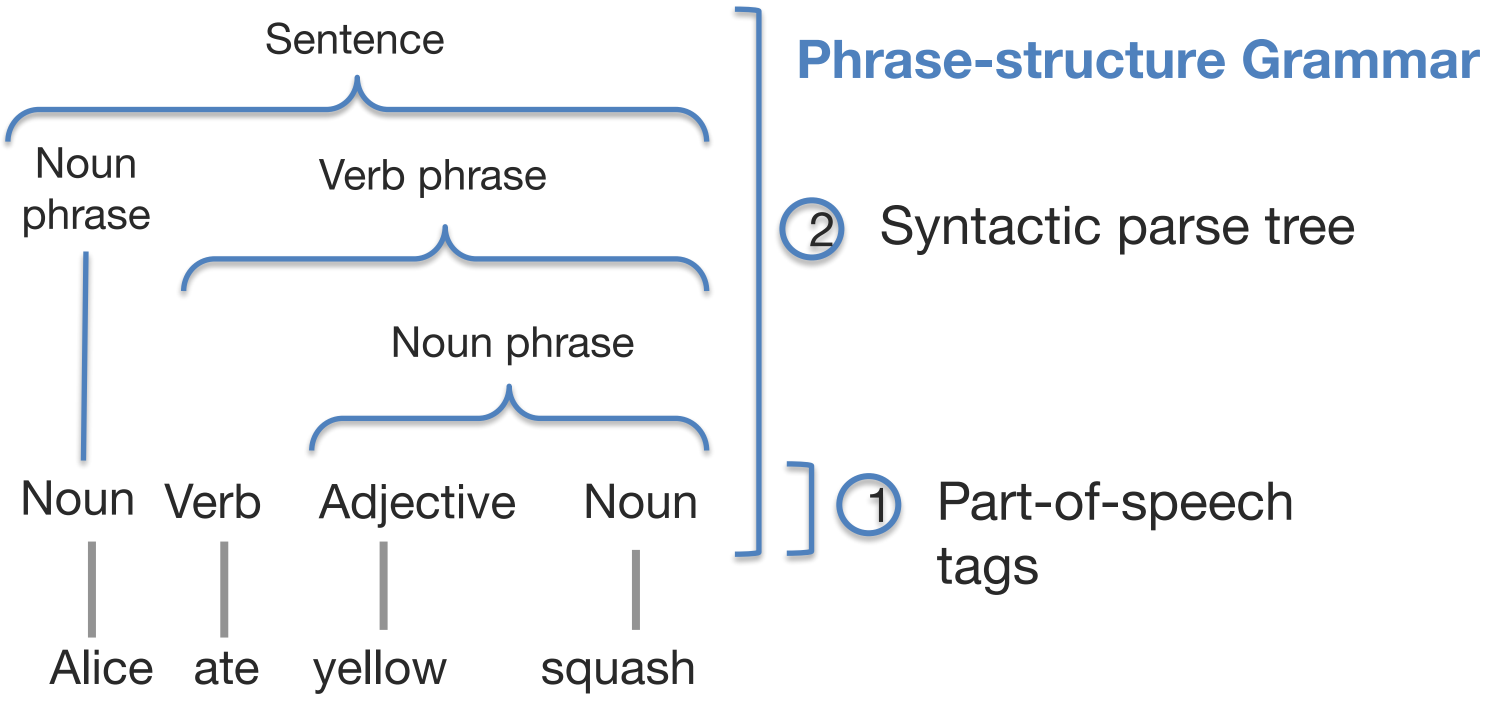
Language Model Application: Speech



Syntax and Language Structure

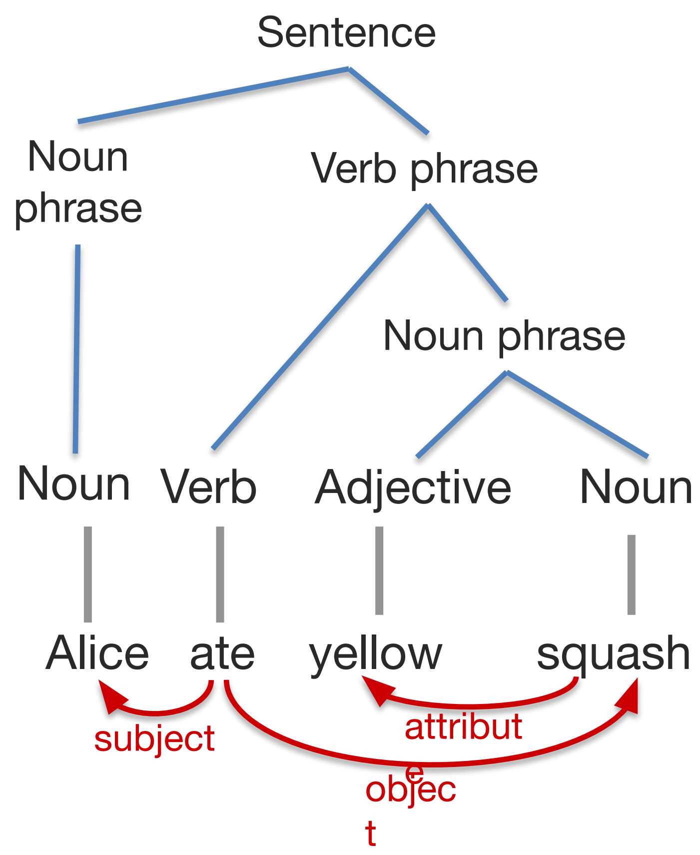
Syntax and Language Structure

What can you tell about this sentence?



Syntax and Language Structure

What can you tell about this sentence?



Phrase-structure Grammar

2 Syntactic parse tree

1 Part-of-speech tags

Dependency Grammar 3

Ambiguity in Syntactic Parsing

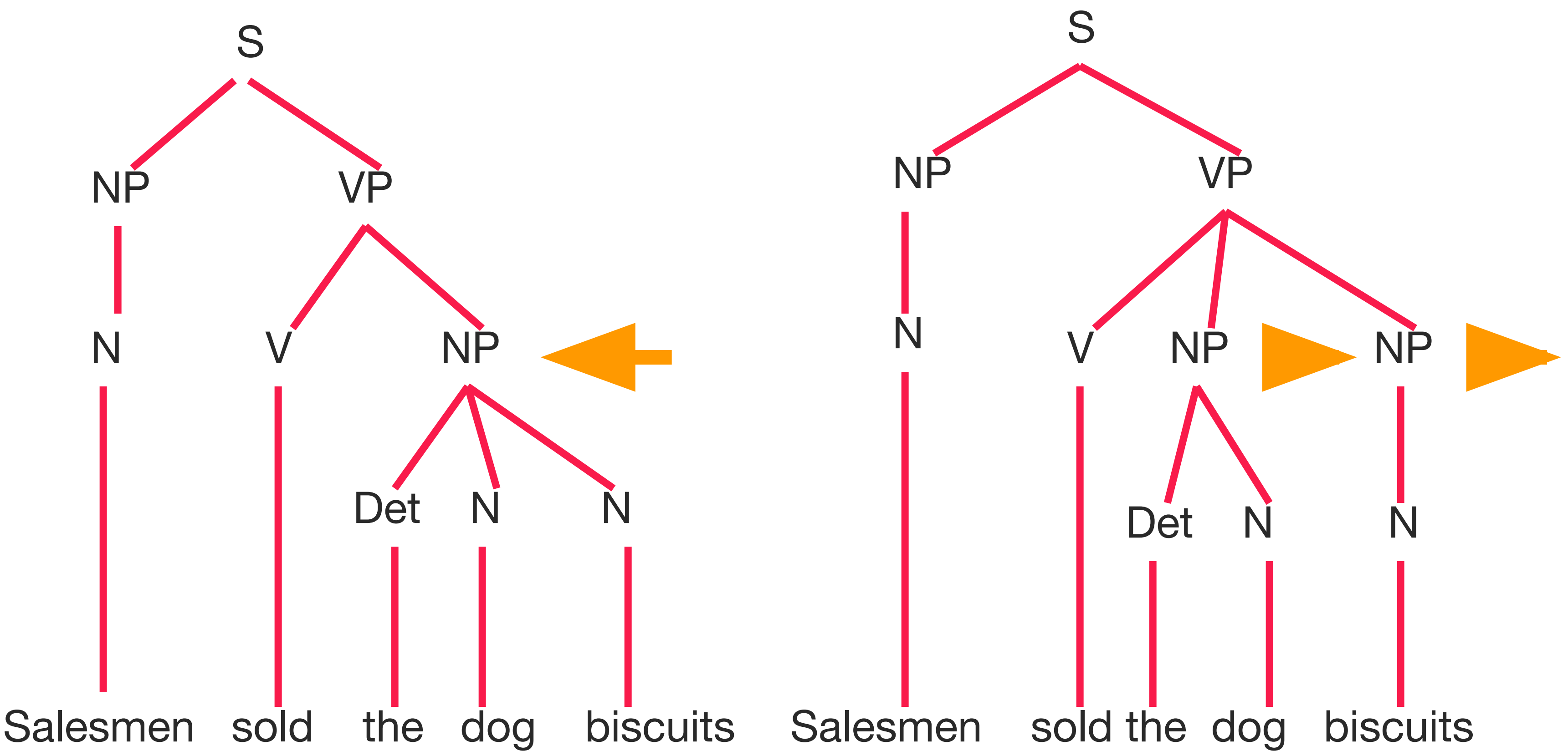
"Like" can be a verb or a preposition

- I like/VBP candy.
- Time flies like/IN an arrow.

"Around" can be a preposition, particle, or adverb

- I bought it at the shop around/IN the corner.
- I never got around/RP to getting a car.
- A new Prius costs around/RB $25K.

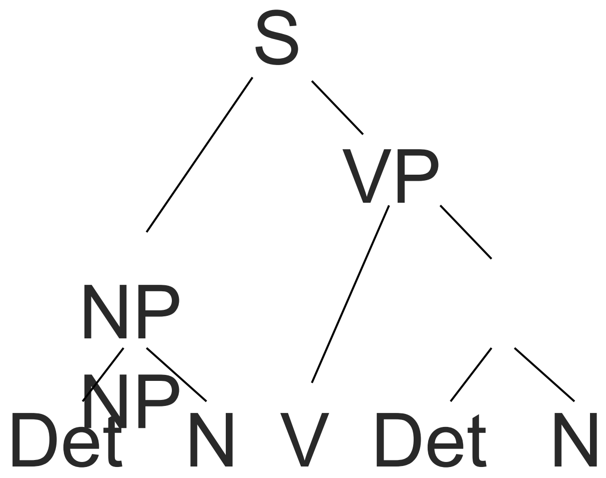
Language Ambiguity



Language Syntax – Examples

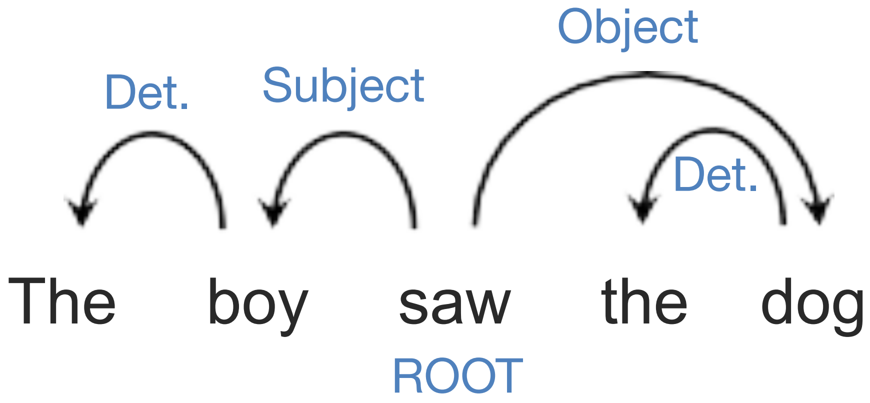
Det Noun Verb Det Noun Prep Det Noun
The boy saw the dog in the park

Part of Speech tagging



The boy saw the dog

Constituency Parsing



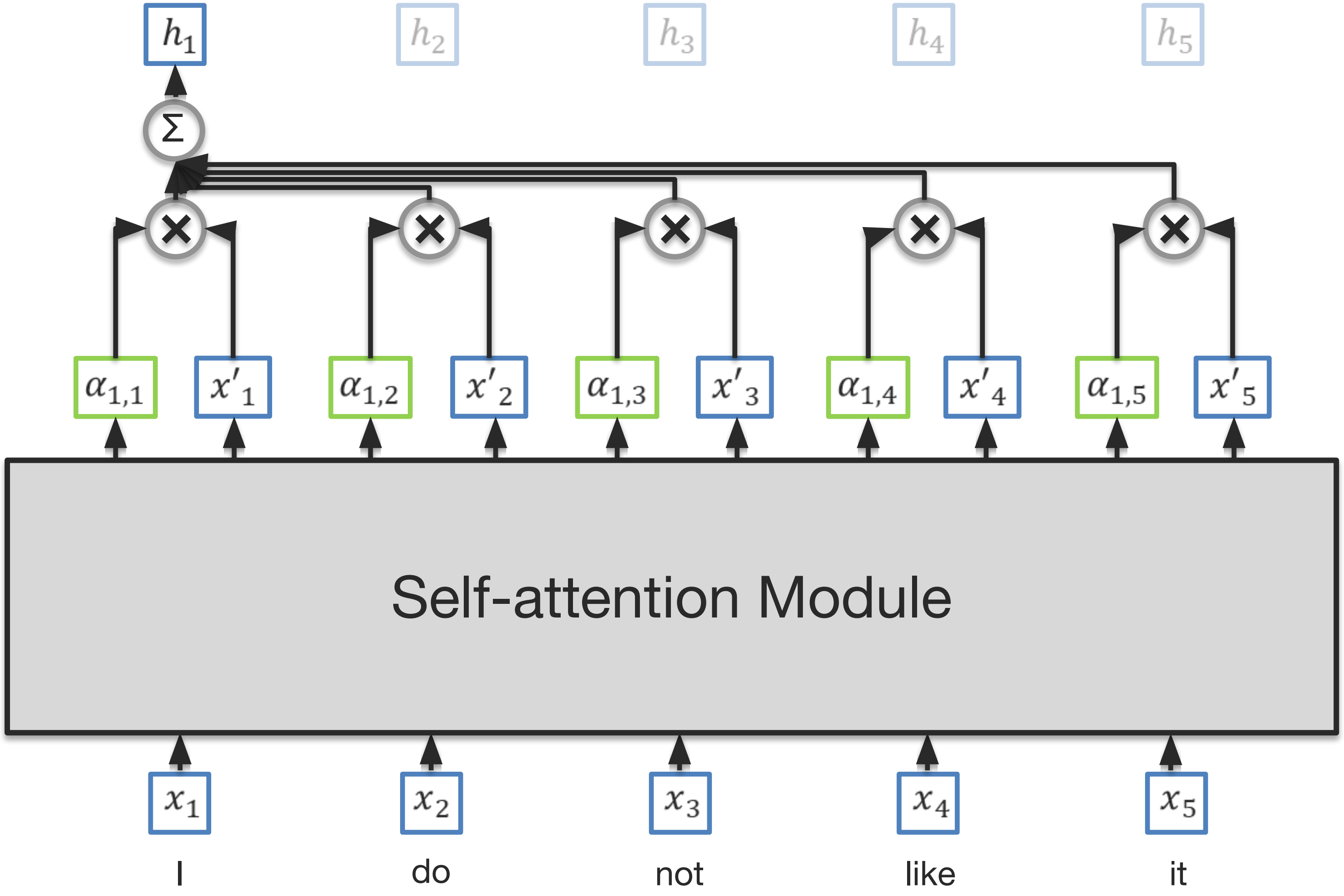
Dependency Parsing

How to model language structure and syntax with neural networks?

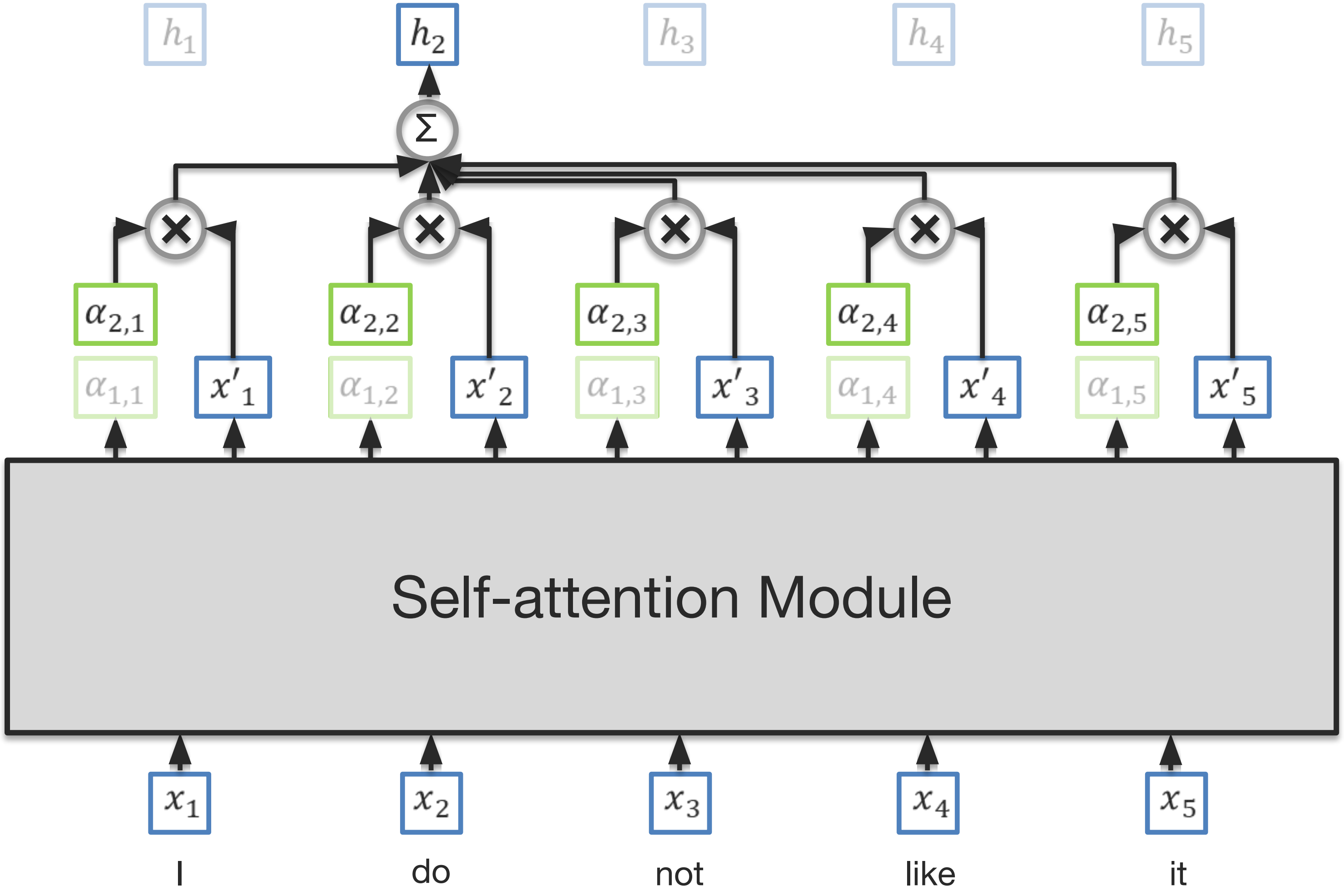
Self-Attention Models

(aka Transformers)

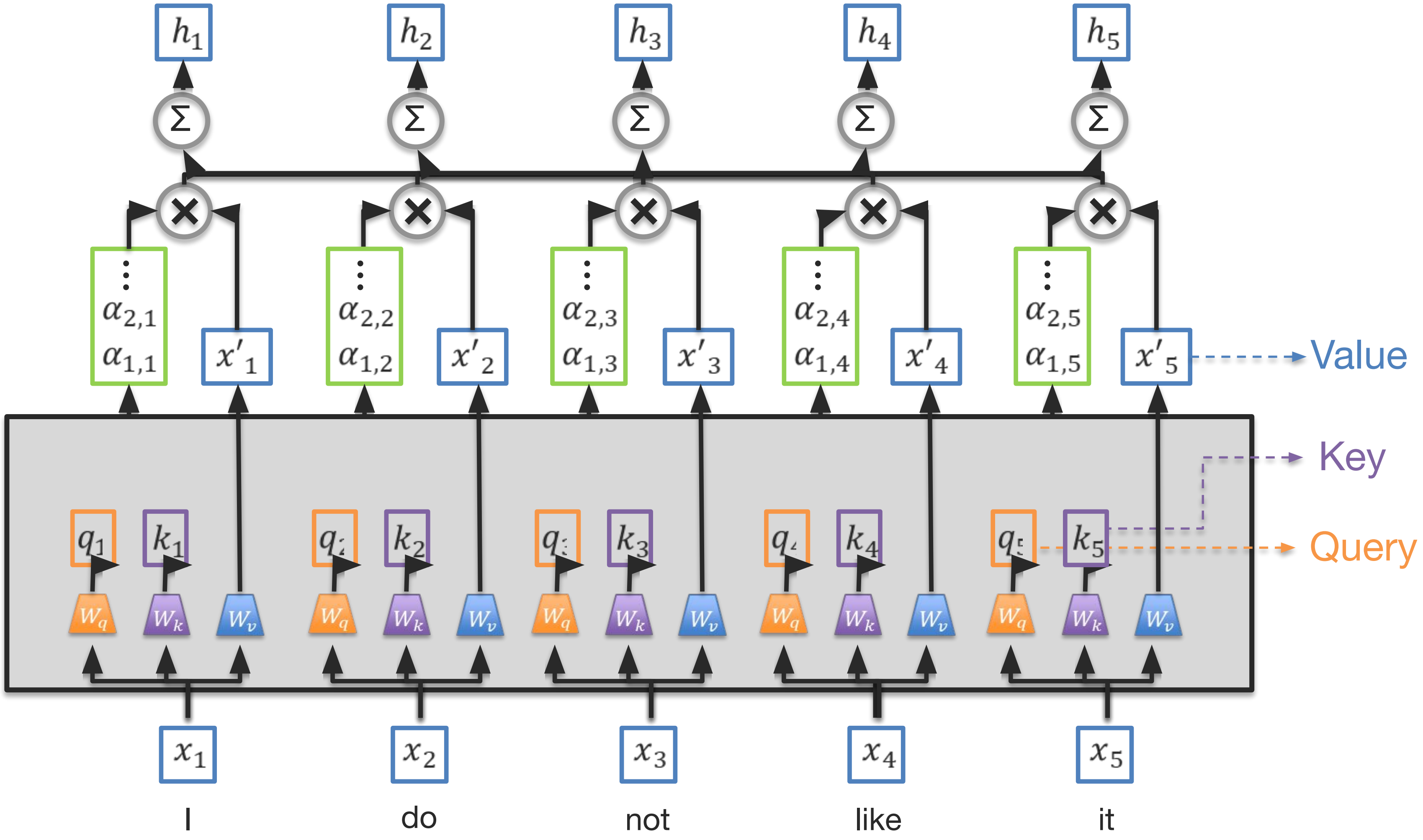
Self-Attention



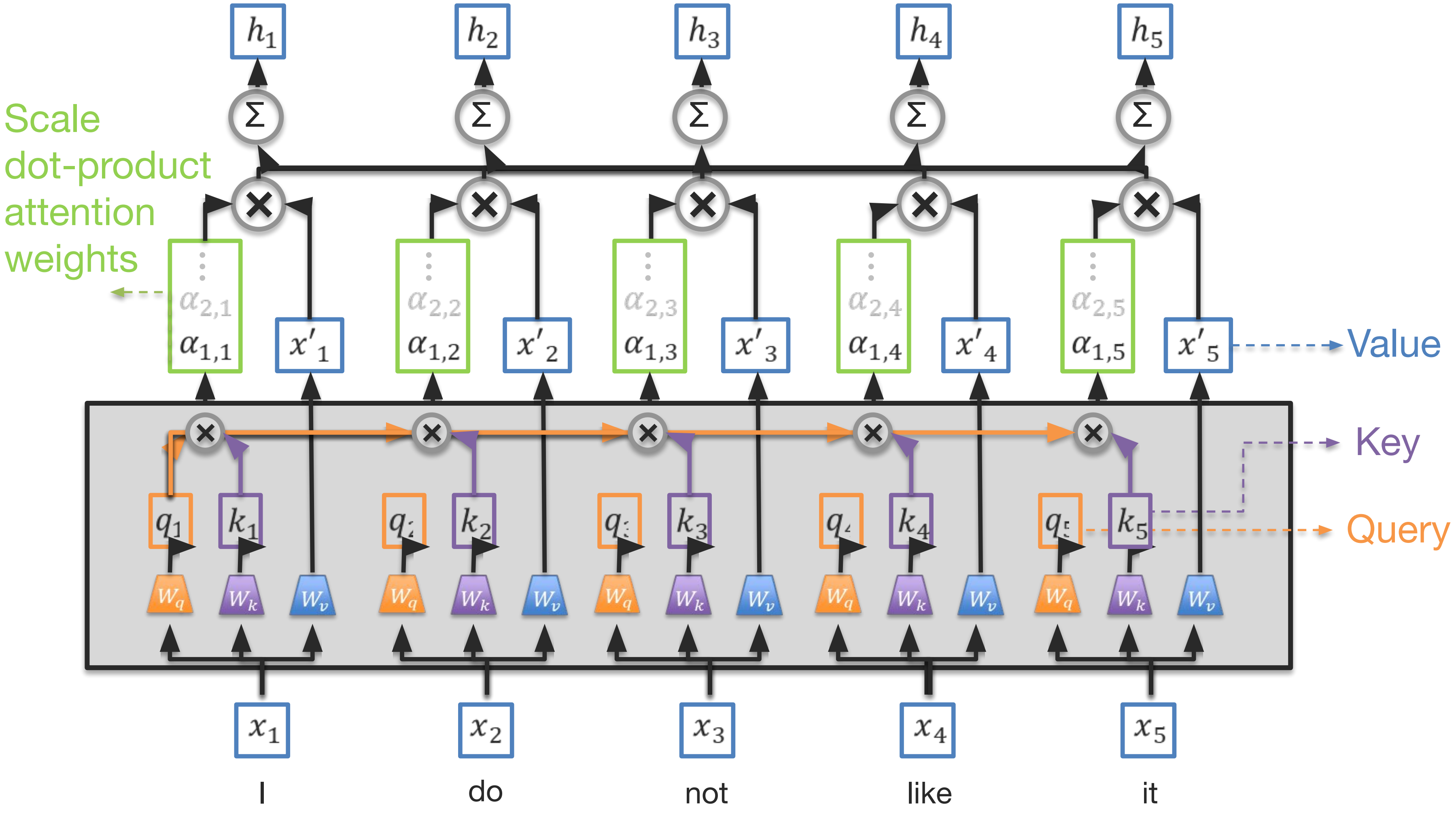
Self-Attention



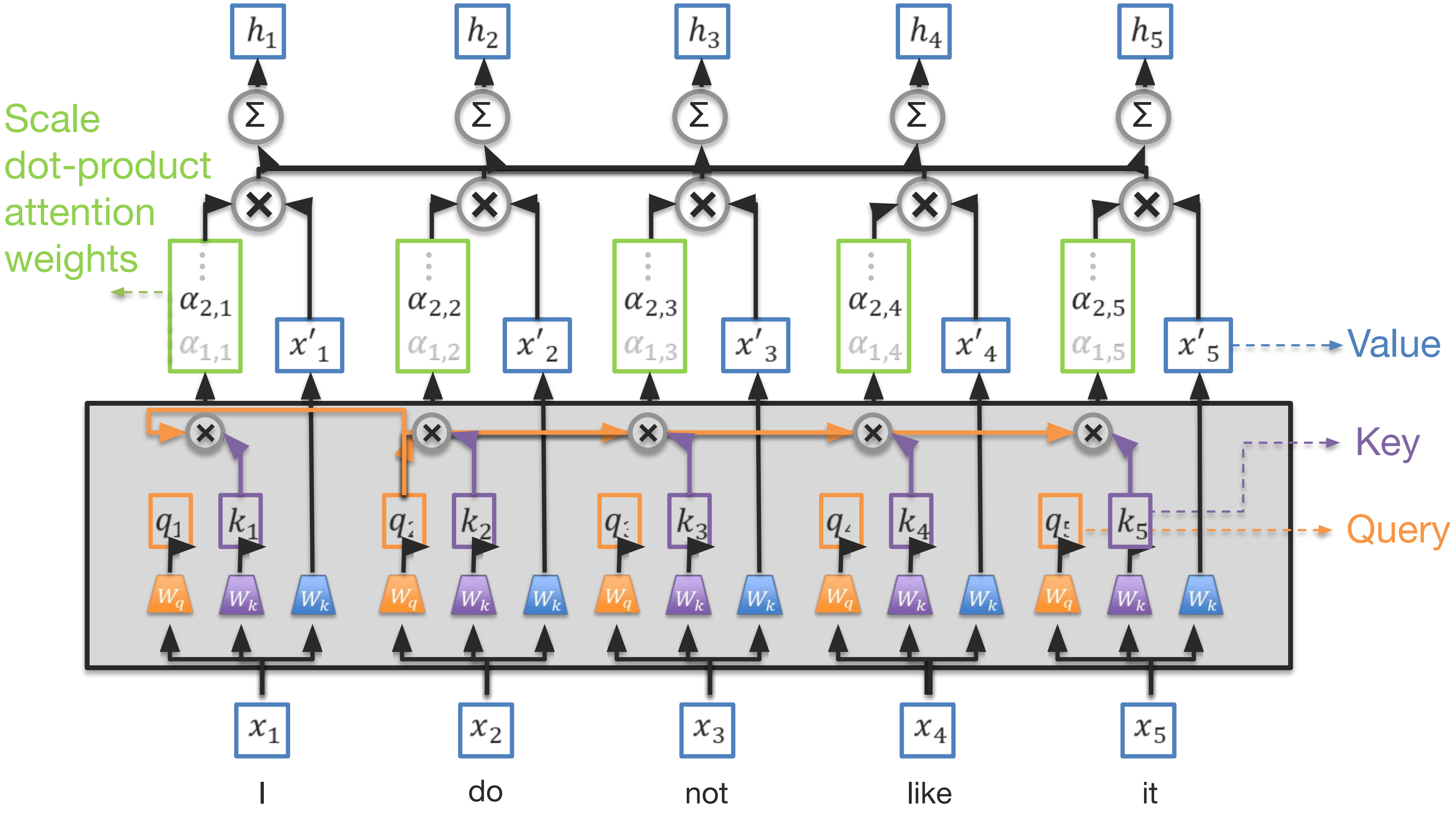
Transformer Self-Attention



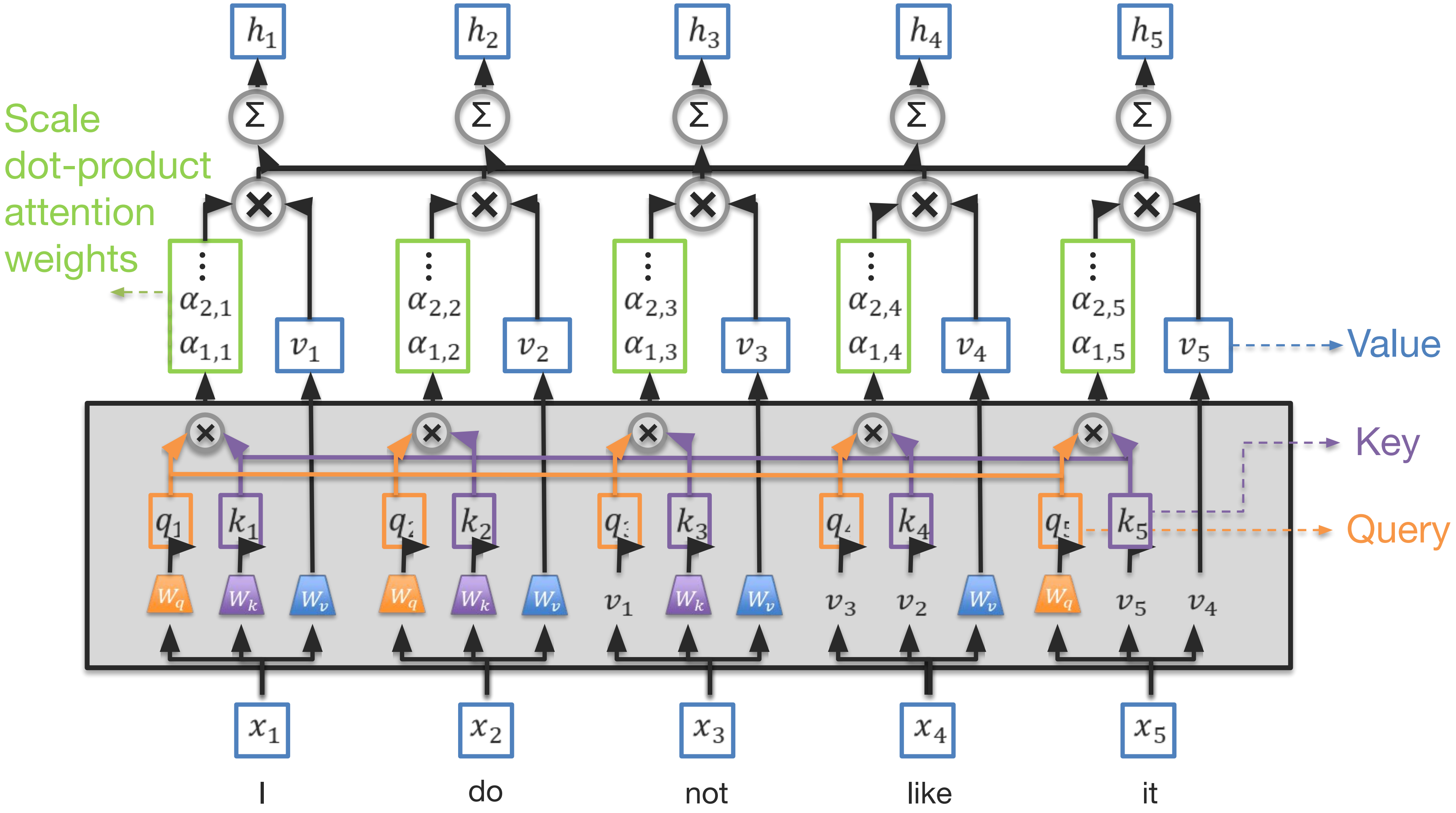
Transformer Self-Attention



Transformer Self-Attention

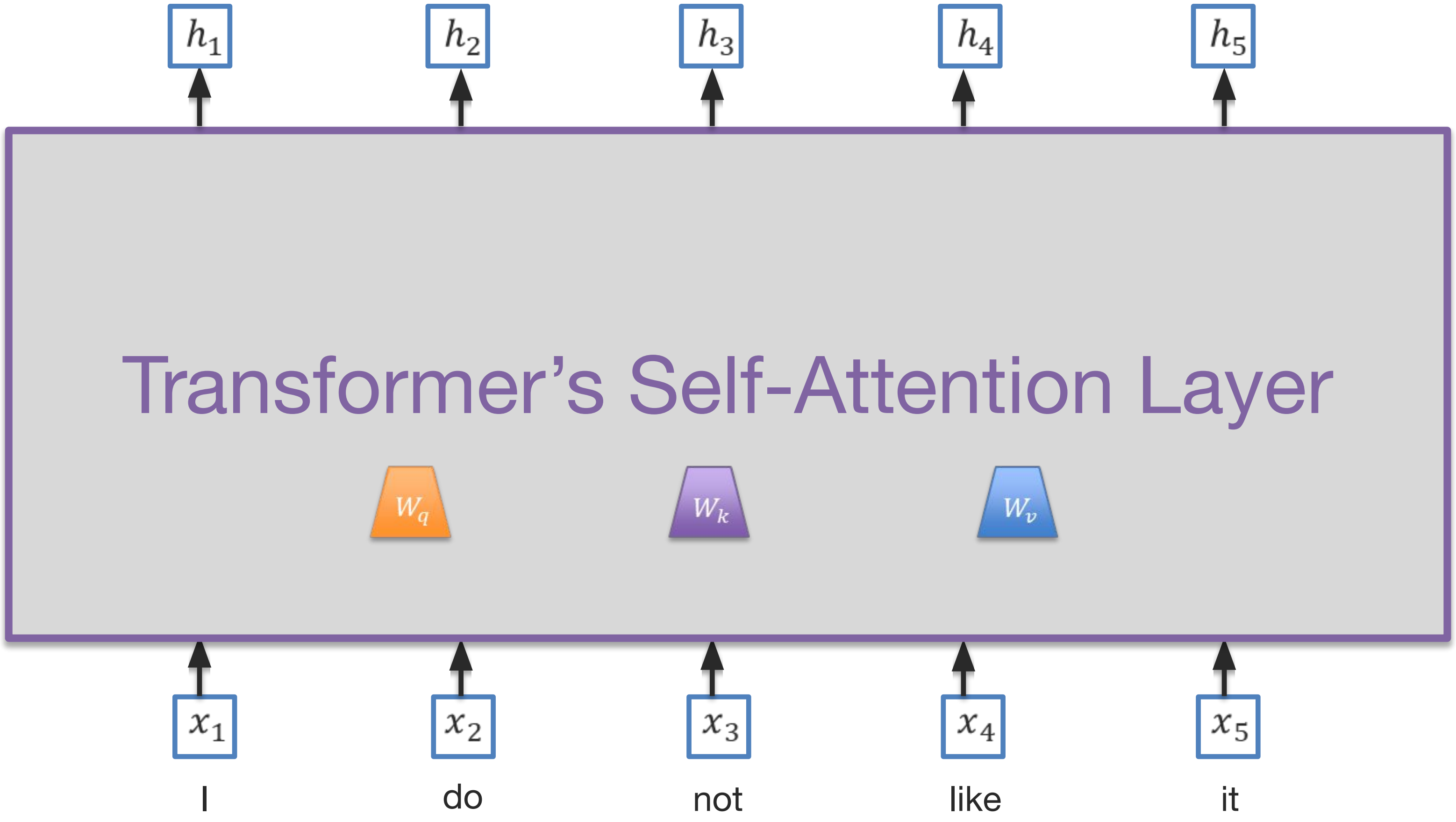


Transformer Self-Attention

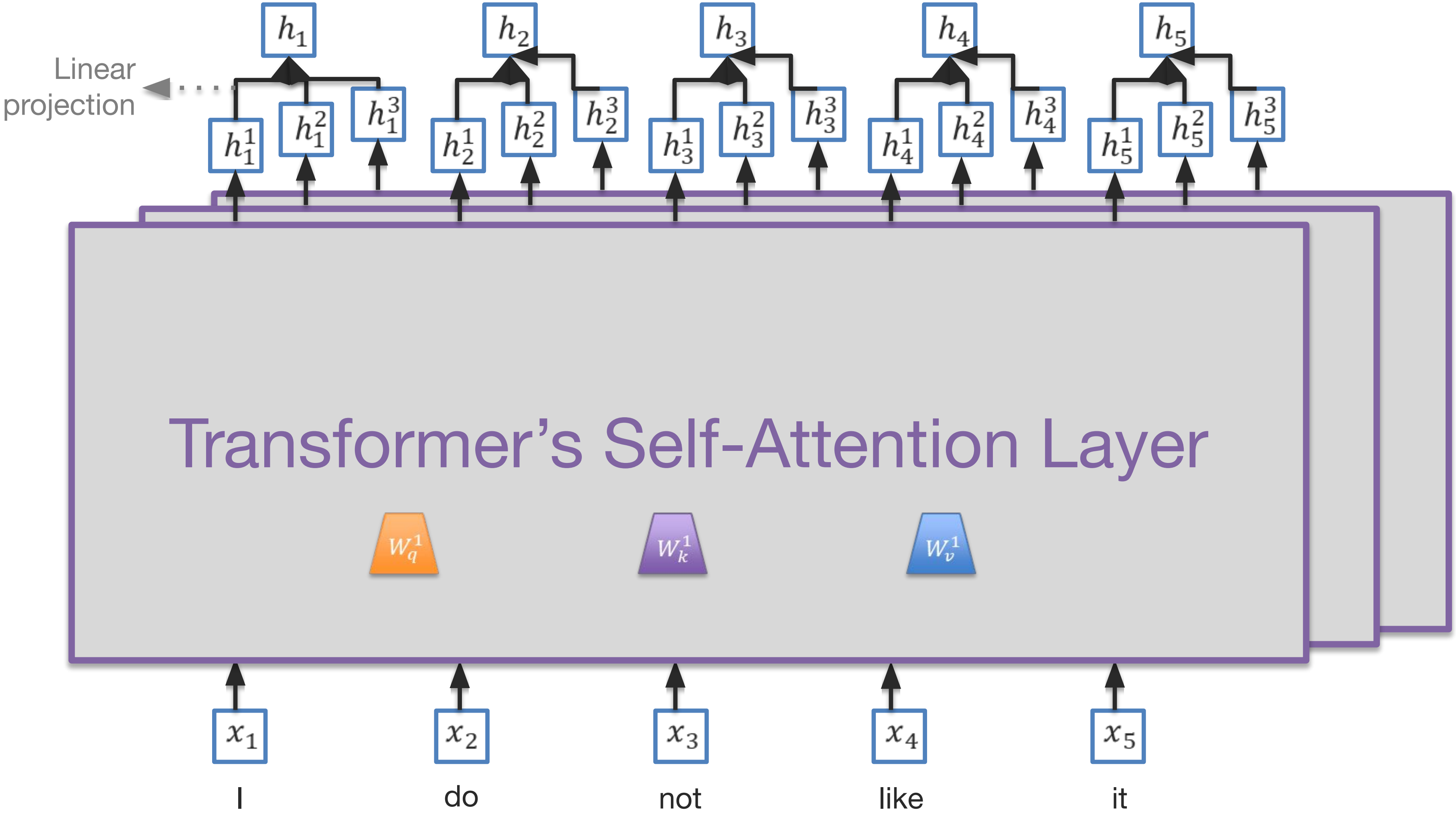


Transformer Self-Attention

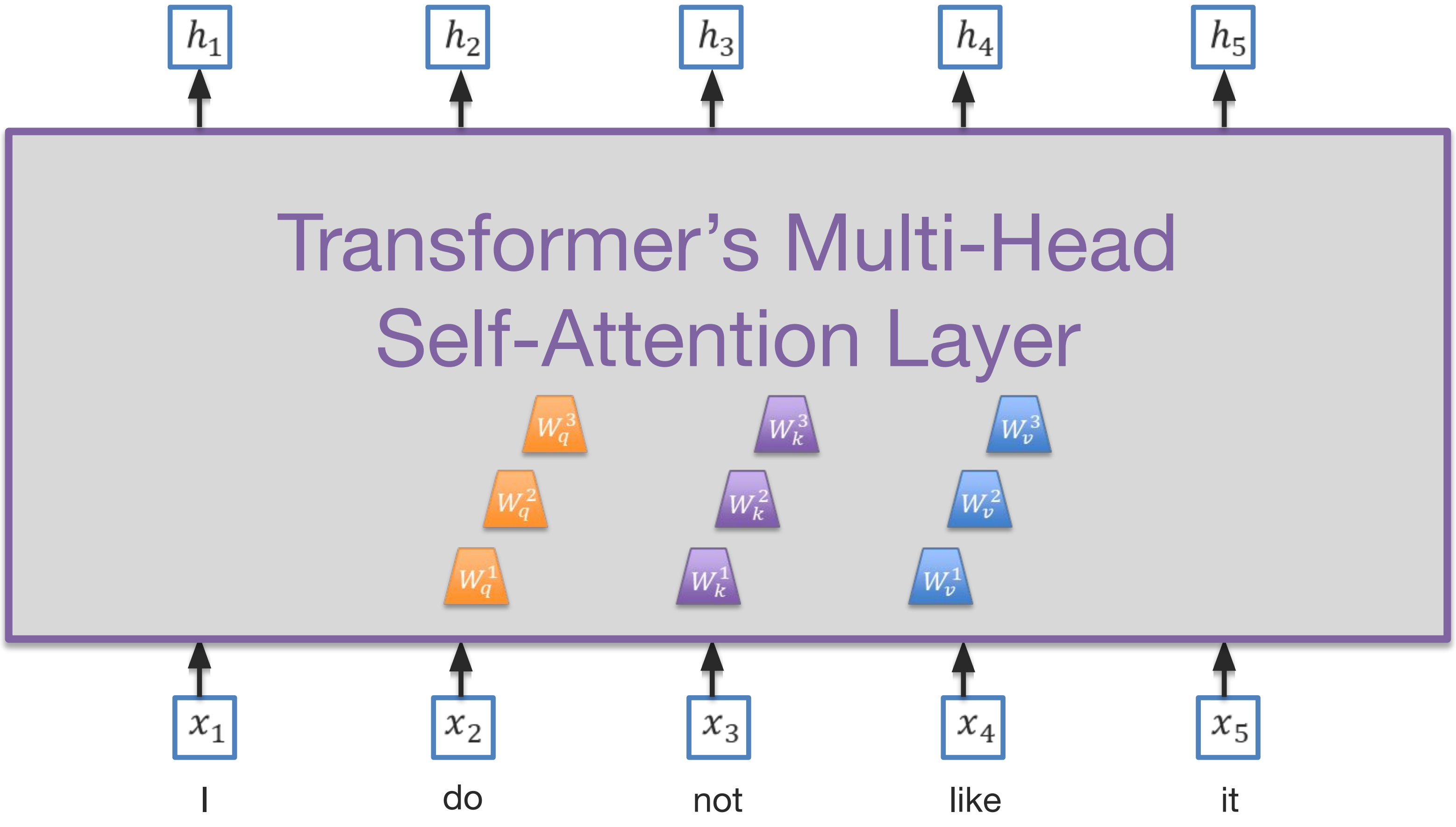
What if we want to attend simultaneously to multiple subspaces of $\boldsymbol{x}$?



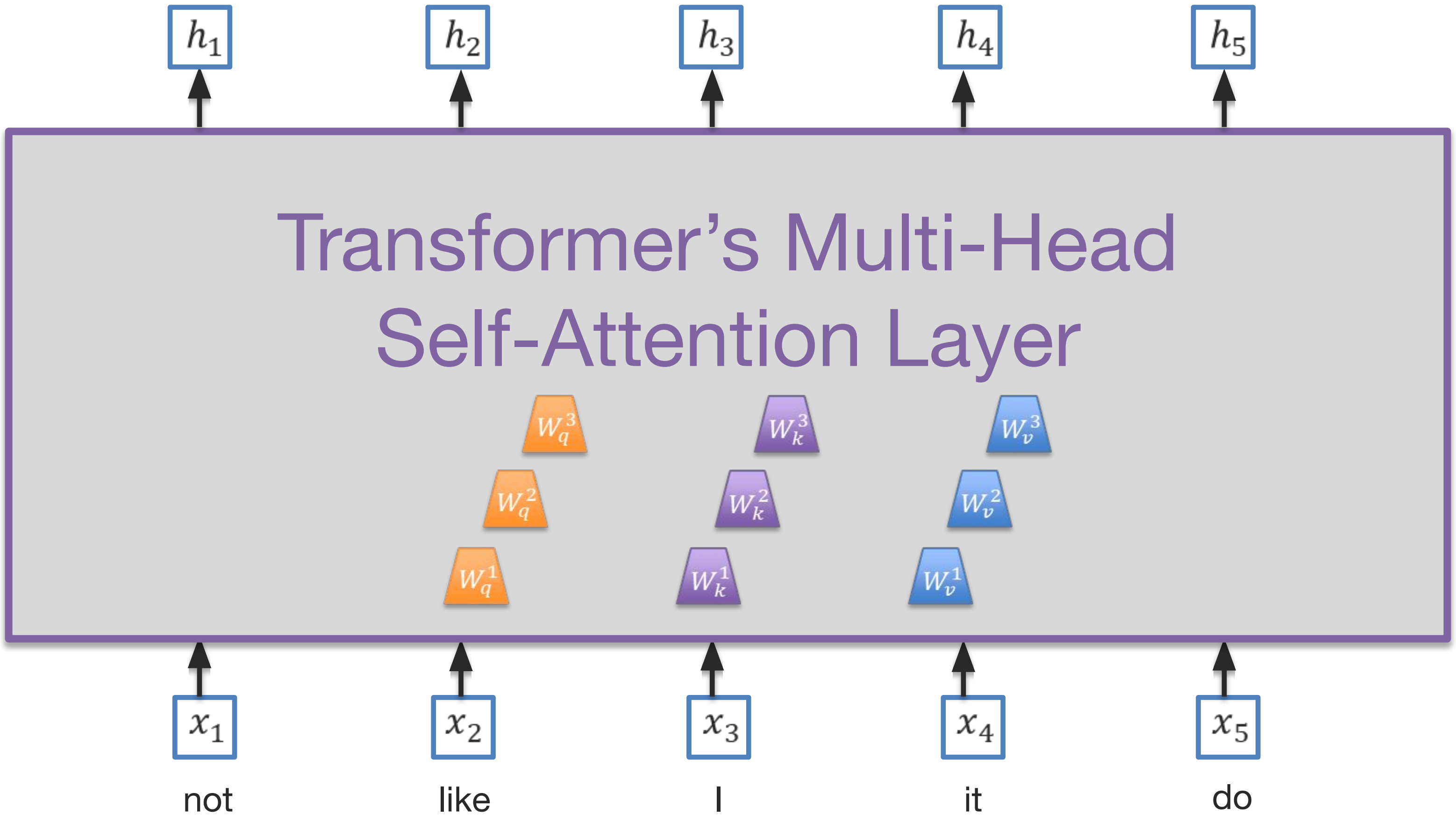
Transformer Multi-Head Self-Attention



Transformer Multi-Head Self-Attention



Transformer Multi-Head Self-Attention



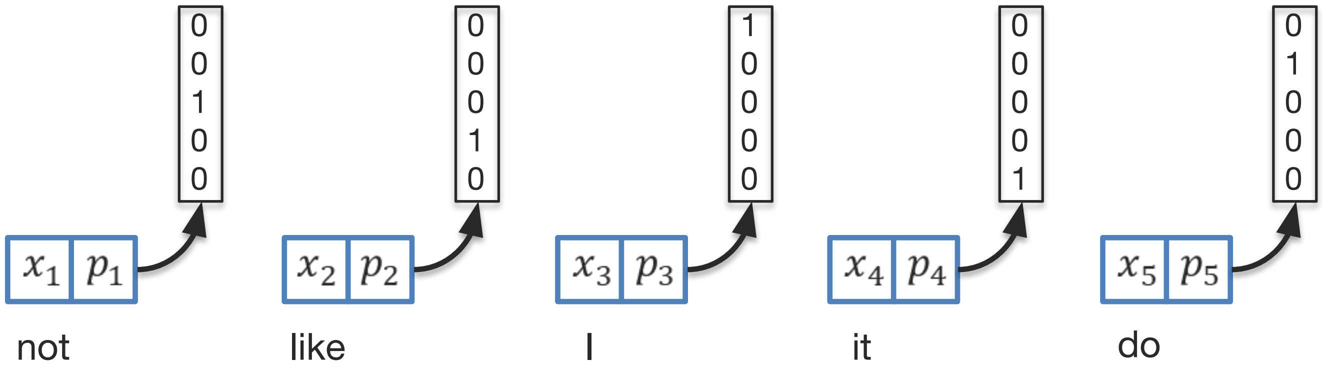
What happens if the words are shuffled?

Position embeddings

- Position information is not encoded in a self-attention module

  How can we encode position information?

Simple approach: one-hot encoding

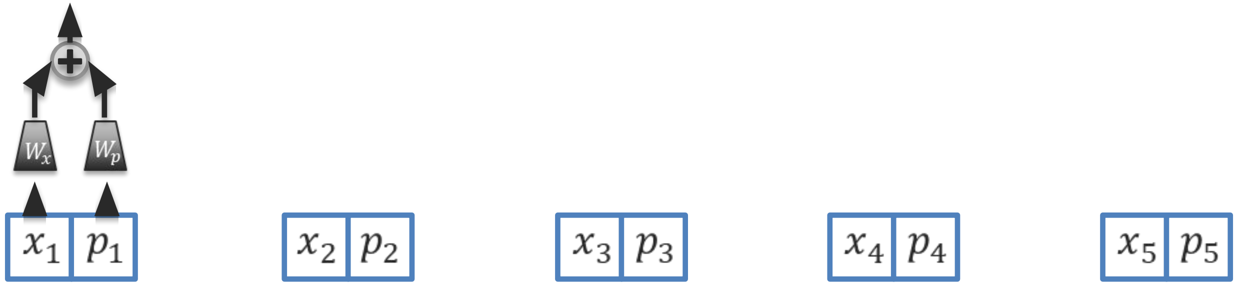


Position embeddings

- ❑ Position information is not encoded in a self-attention module

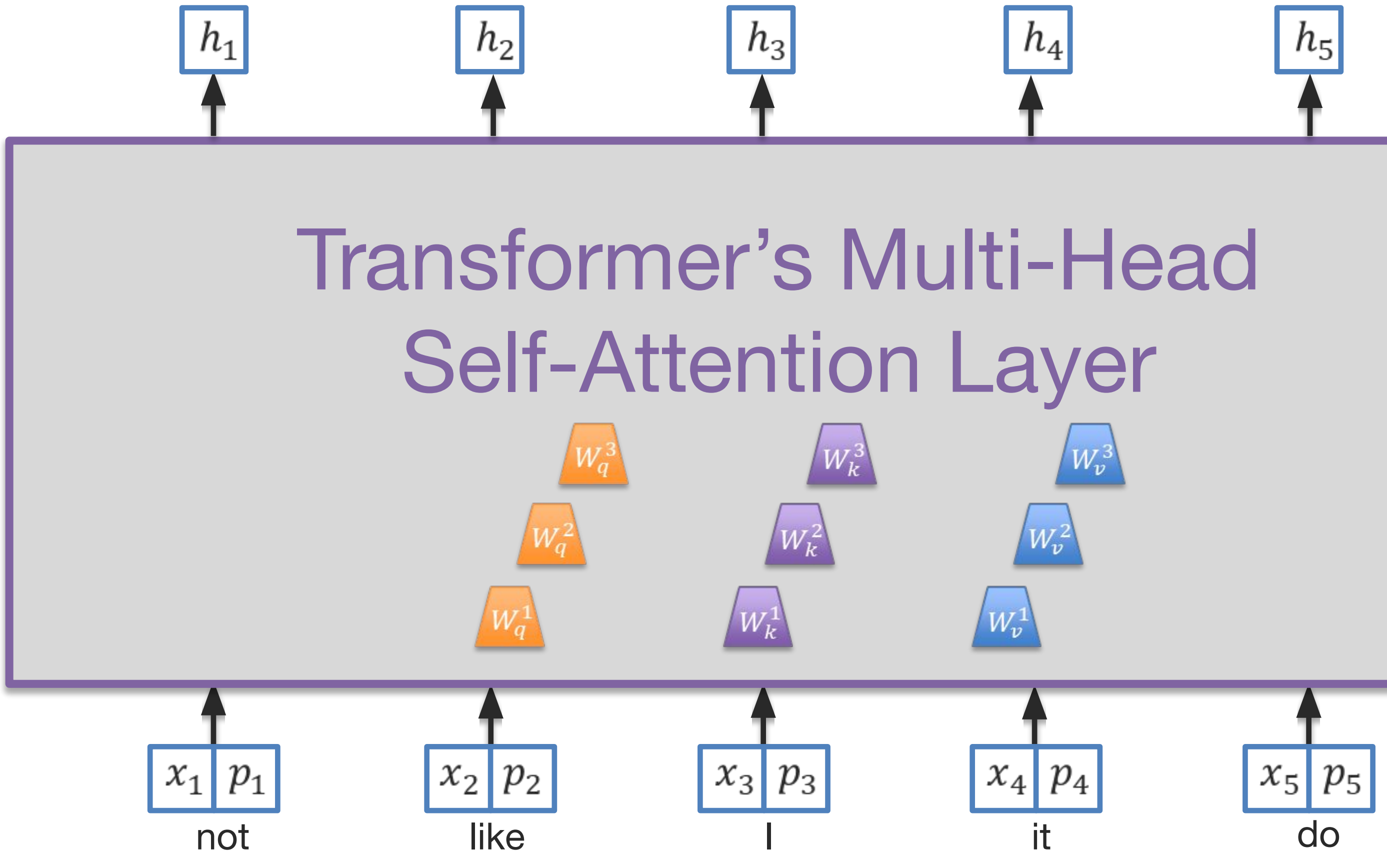
How can we encode position information?

Simple approach: one-hot encoding + linear embeddings+ { Sum - or - concat }



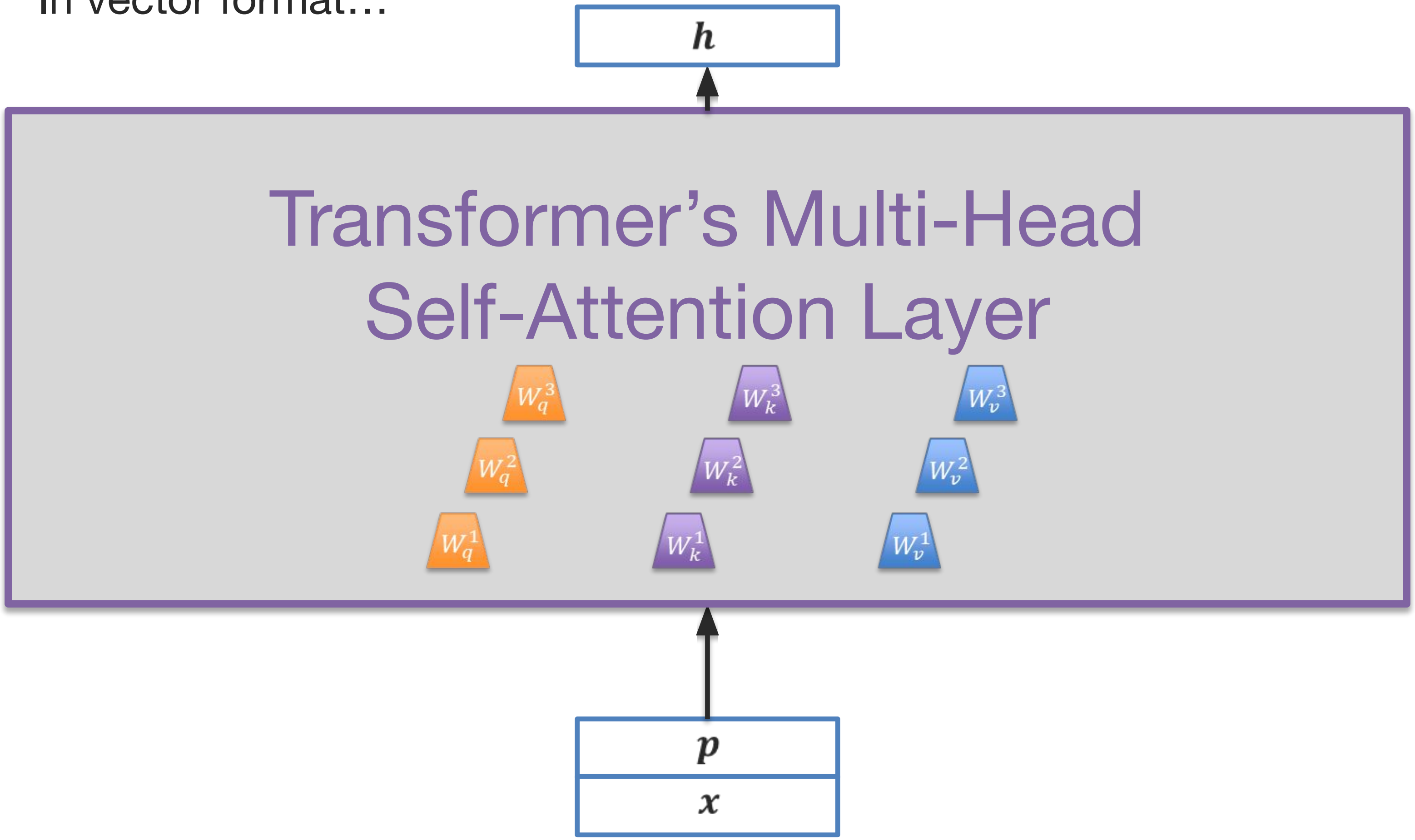
not like I it do

Transformer Multi-Head Self-Attention

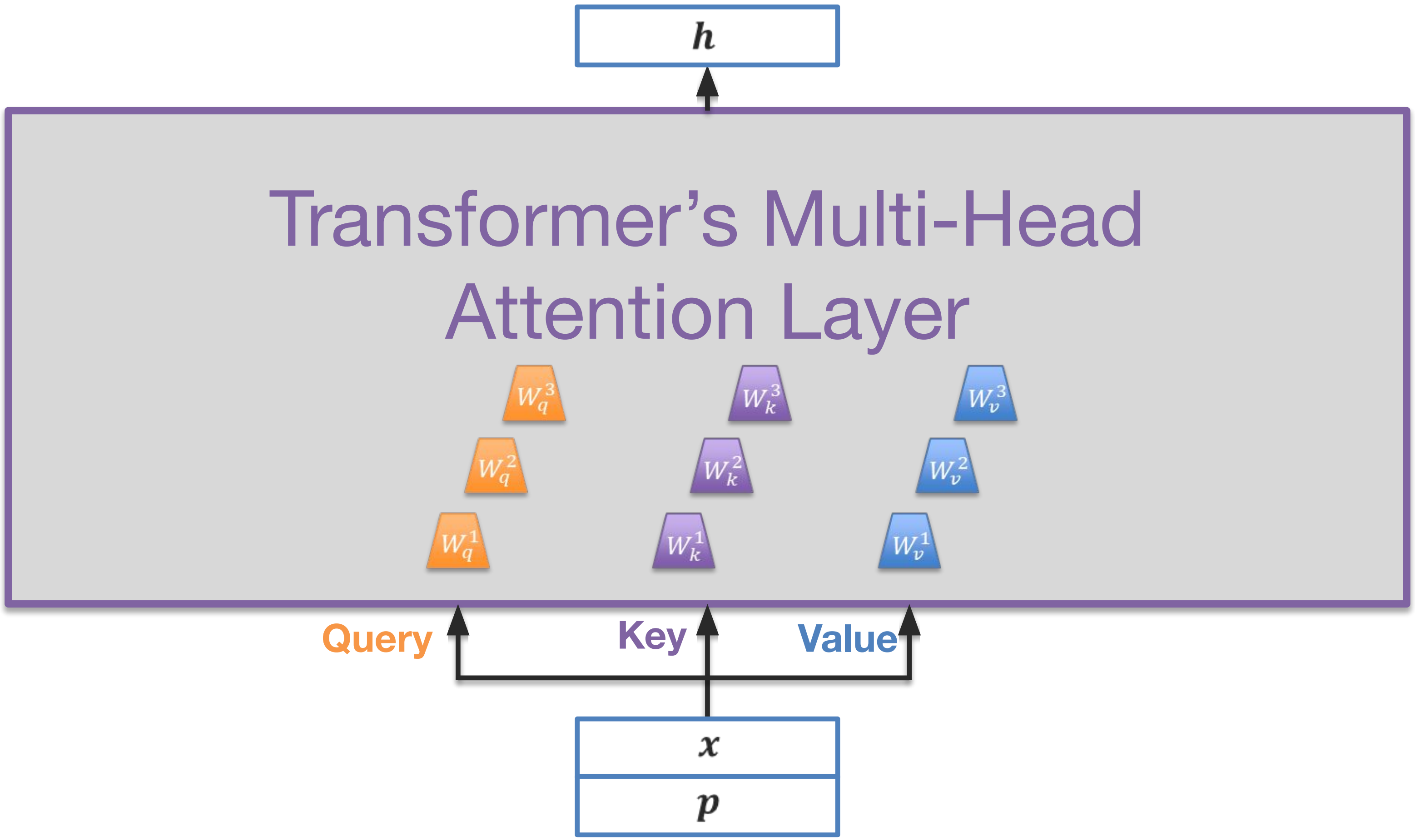


Transformer Multi-Head Self-Attention

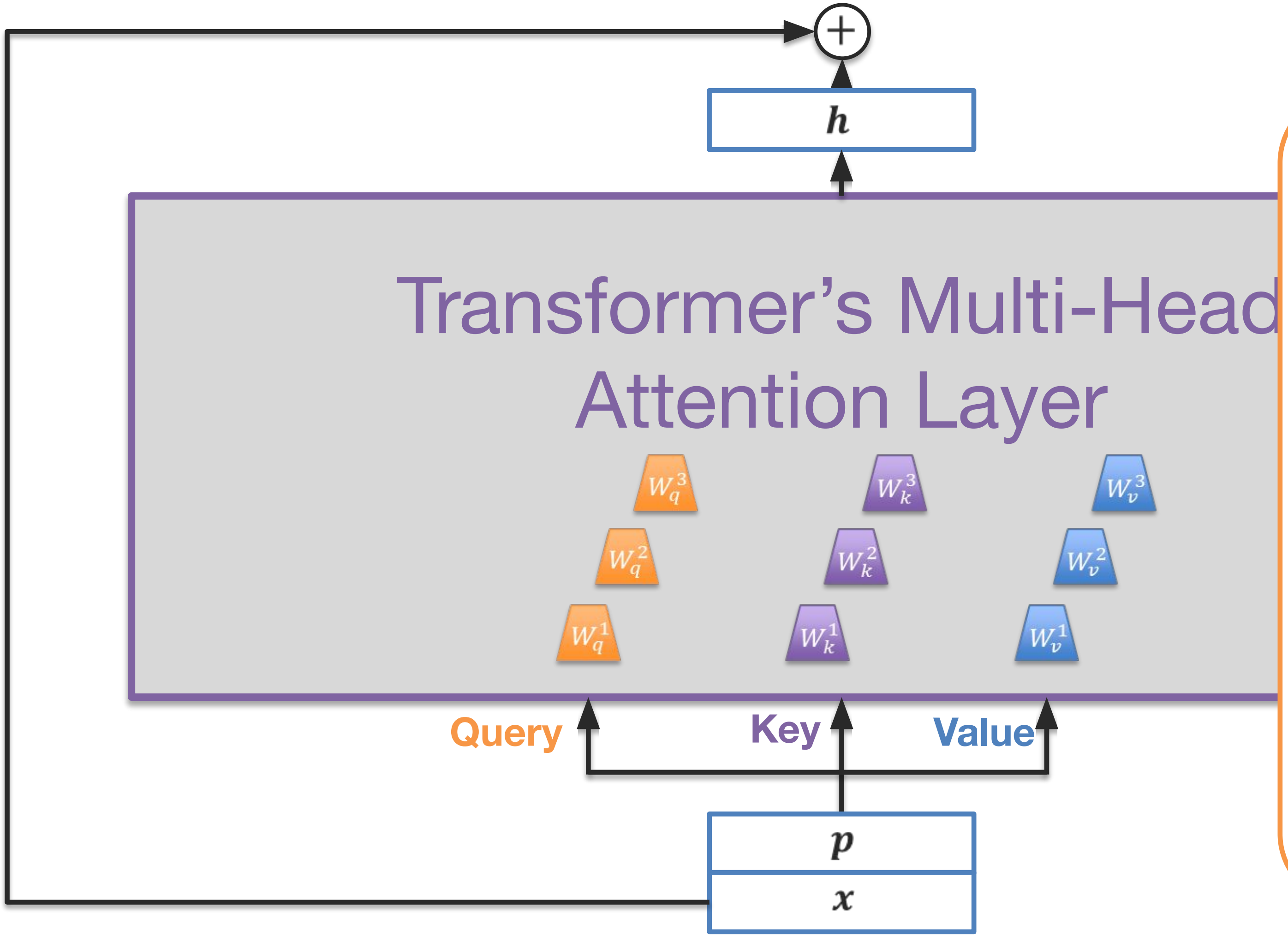
In vector format…

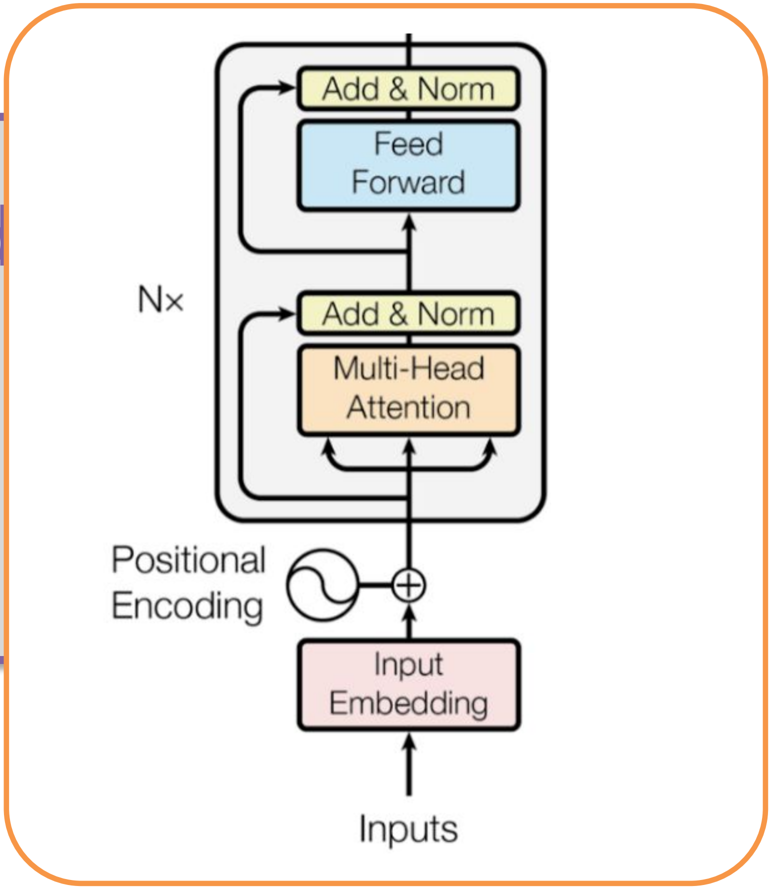


Transformer Multi-Head Attention



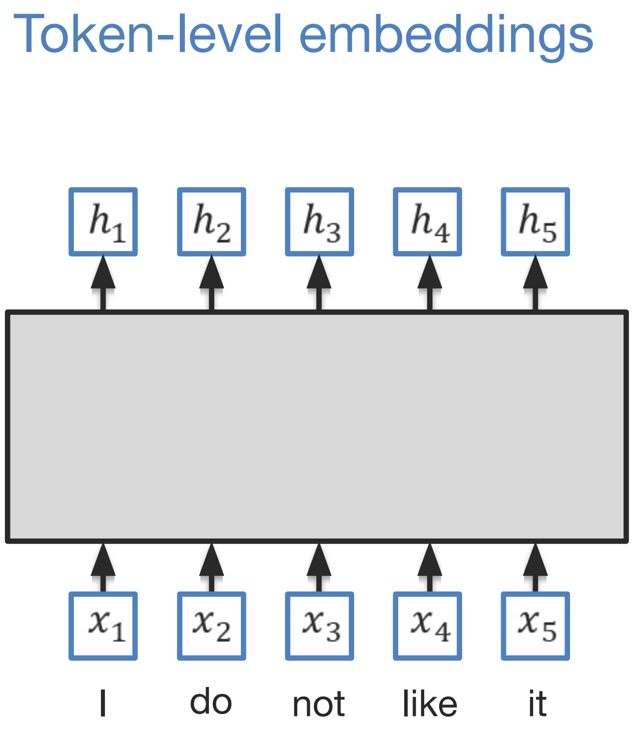
Transformer – Residual Connection

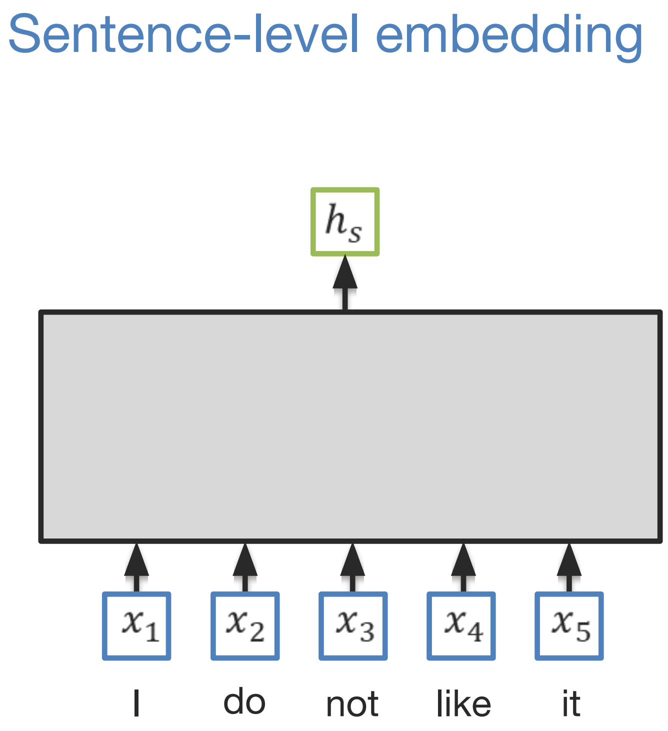




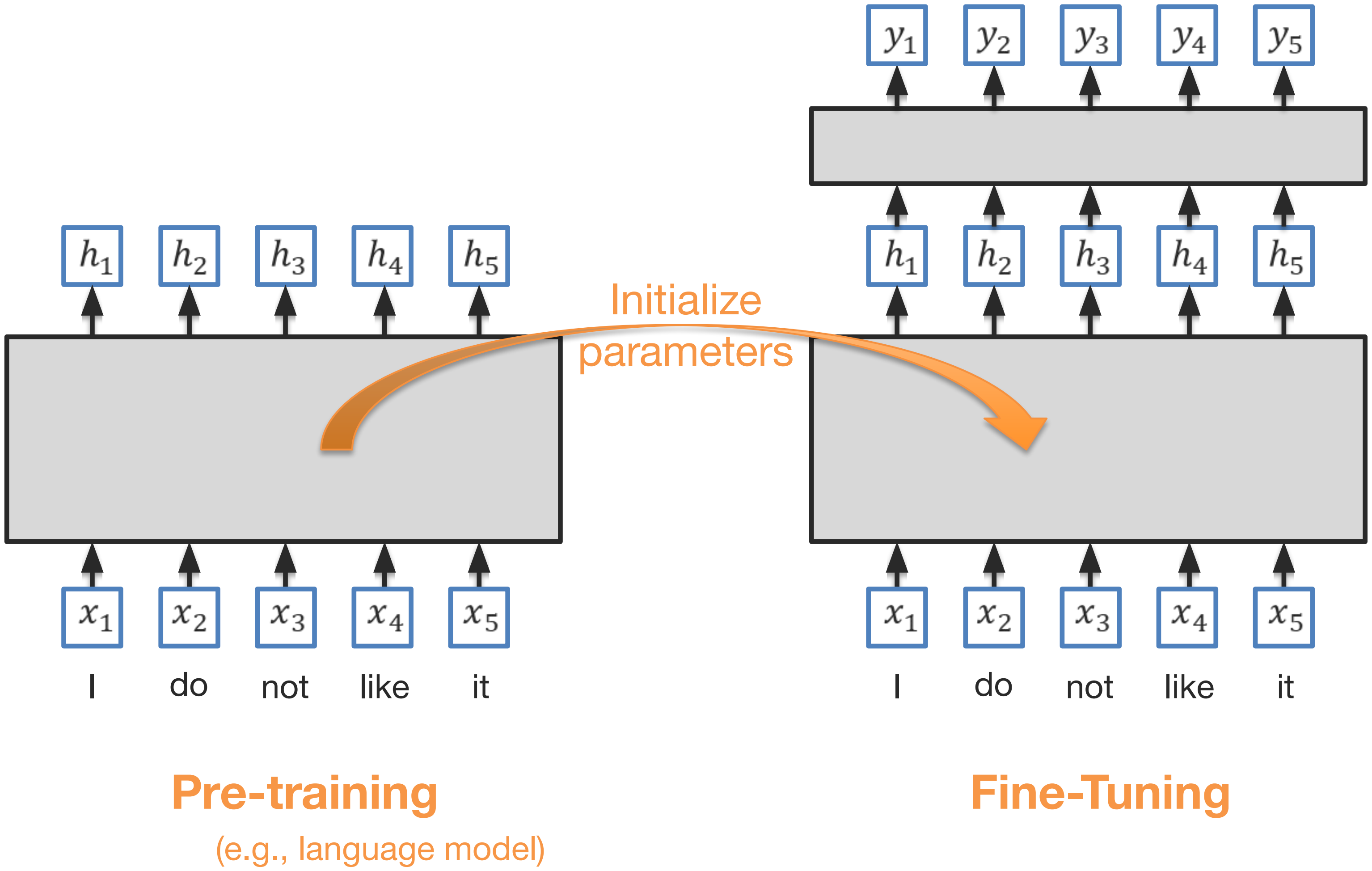
Pre-training and Language Transformers

Token-level and Sentence-level Embeddings





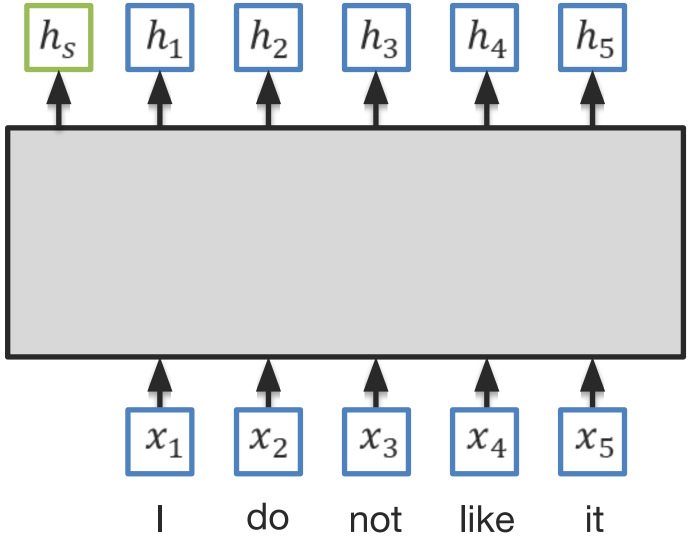
Pre-Training and Fine-Tuning



BERT: Bidirectional Encoder Representations from Transformers

Advantages:

1. Jointly learn representation for token-level and sentence level
2. Same network architecture for pre-training and fine-tuning

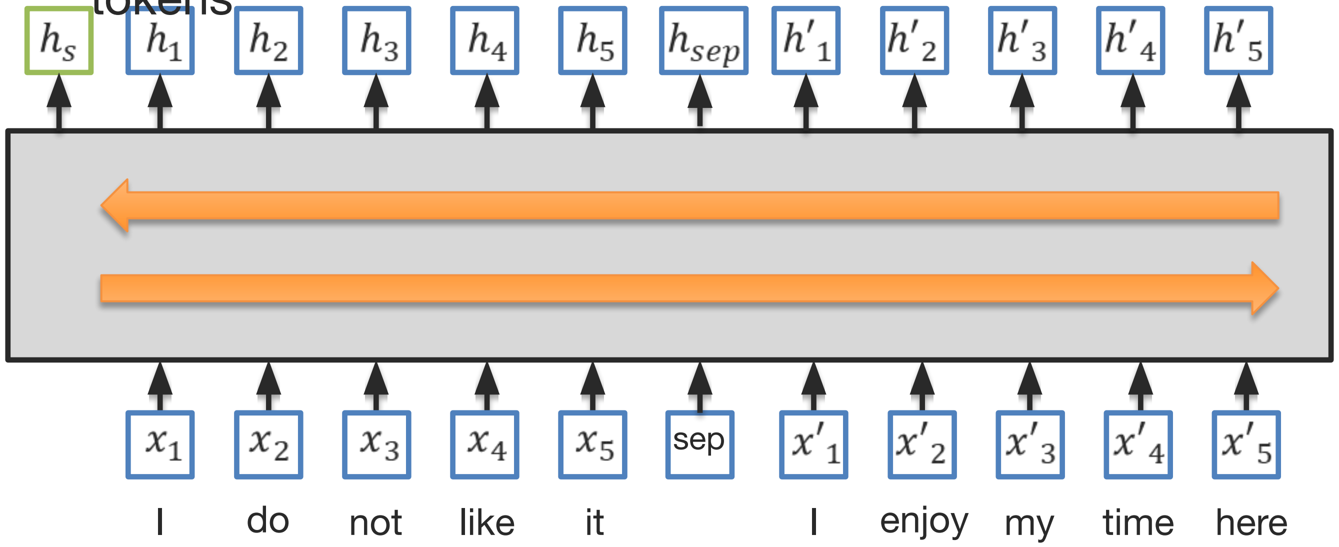


BERT: Bidirectional Encoder Representations from Transformers

Advantages:

1. Jointly learn representation for token-level and sentence level
2. Same network architecture for pre-training and fine-tuning
3. Can be used learn relationship between sentences
4. Models bidirectional and long-range interactions between tokens

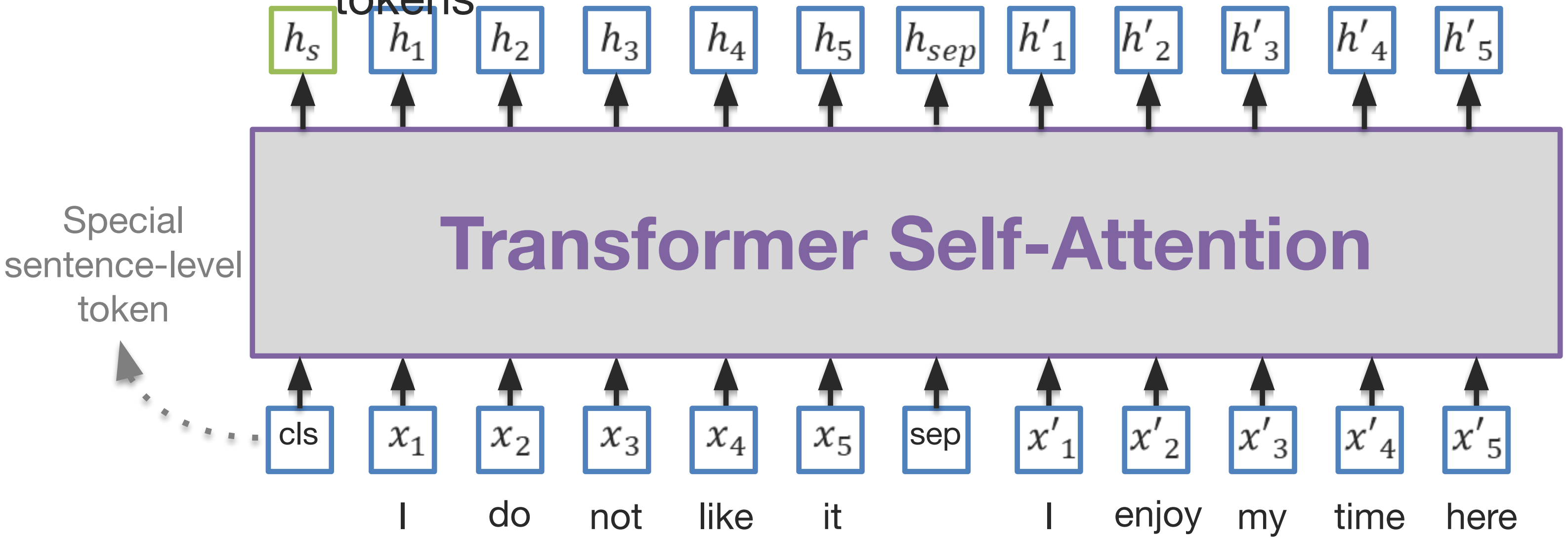
How can we do all this?



BERT: Bidirectional Encoder Representations from Transformers

Advantages:

1. Jointly learn representation for token-level and sentence level
2. Same network architecture for pre-training and fine-tuning
3. Can be used learn relationship between sentences
4. Models bidirectional interactions between tokens



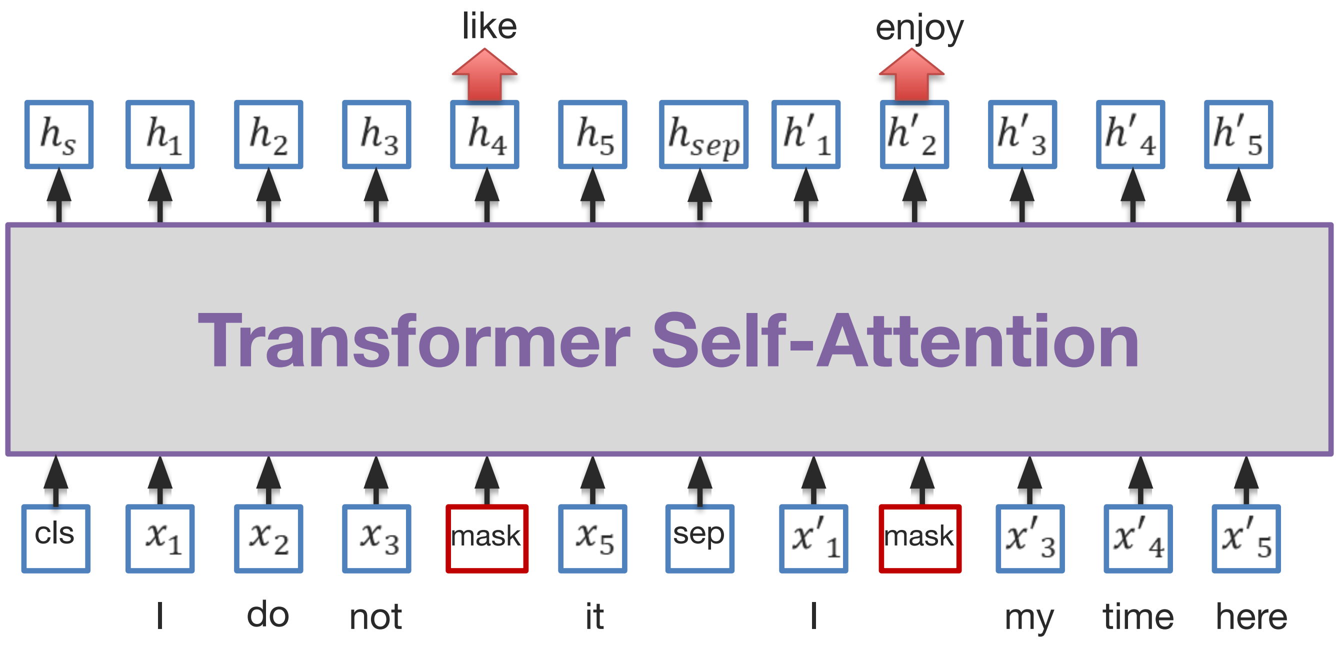
But how to train self-supervised?

Pre-training BERT Model

1 **Masked Language Model**

Randomly mask input tokens and then try to predict them

What is the loss function?



Pre-training BERT Model

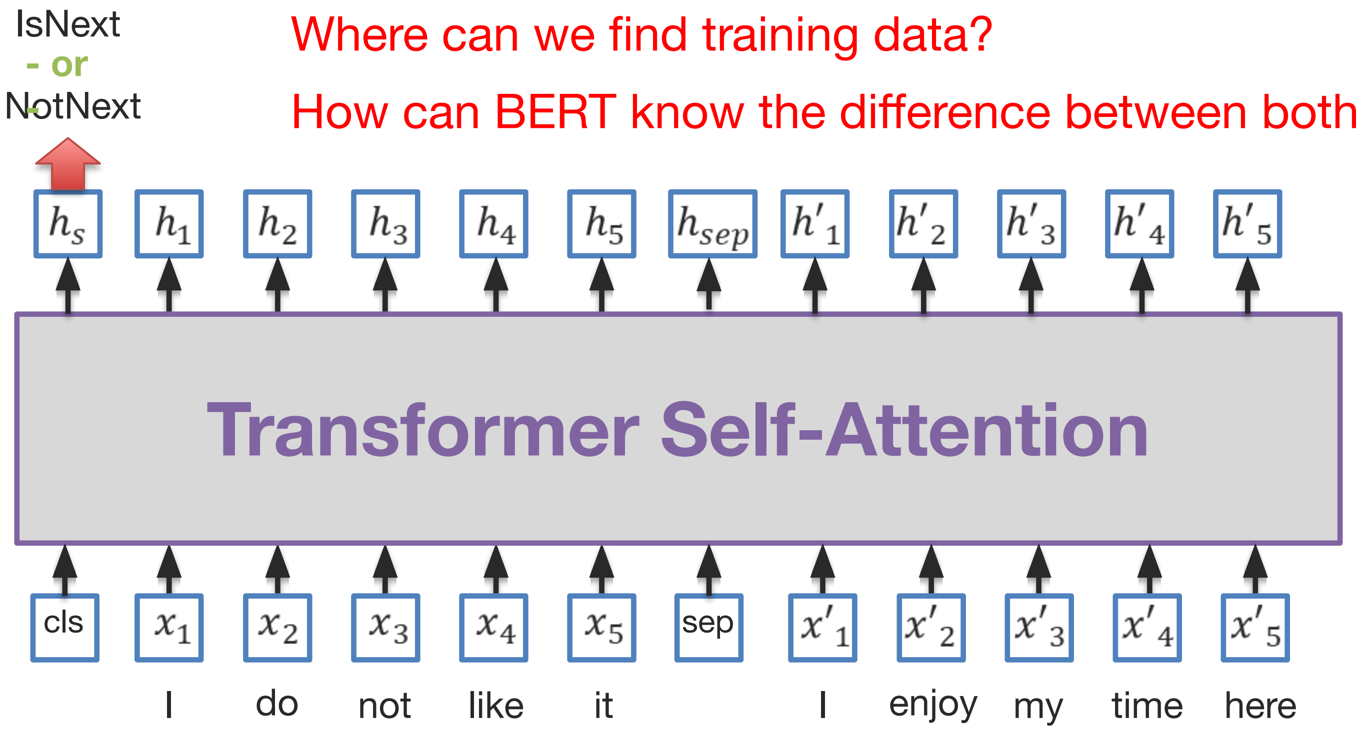
2 Next Sentence Prediction

Given two sentences, predict if this is the next one or not

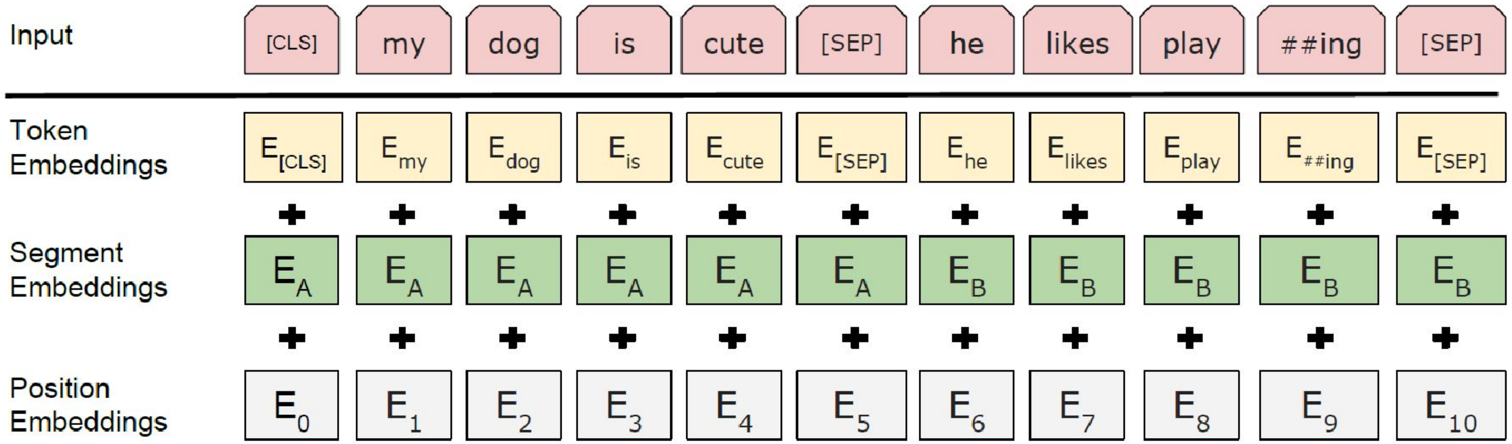
What is the loss function?

Where can we find training data?

How can BERT know the difference between both sentences?



Three Embeddings: Token + Position + Sentence

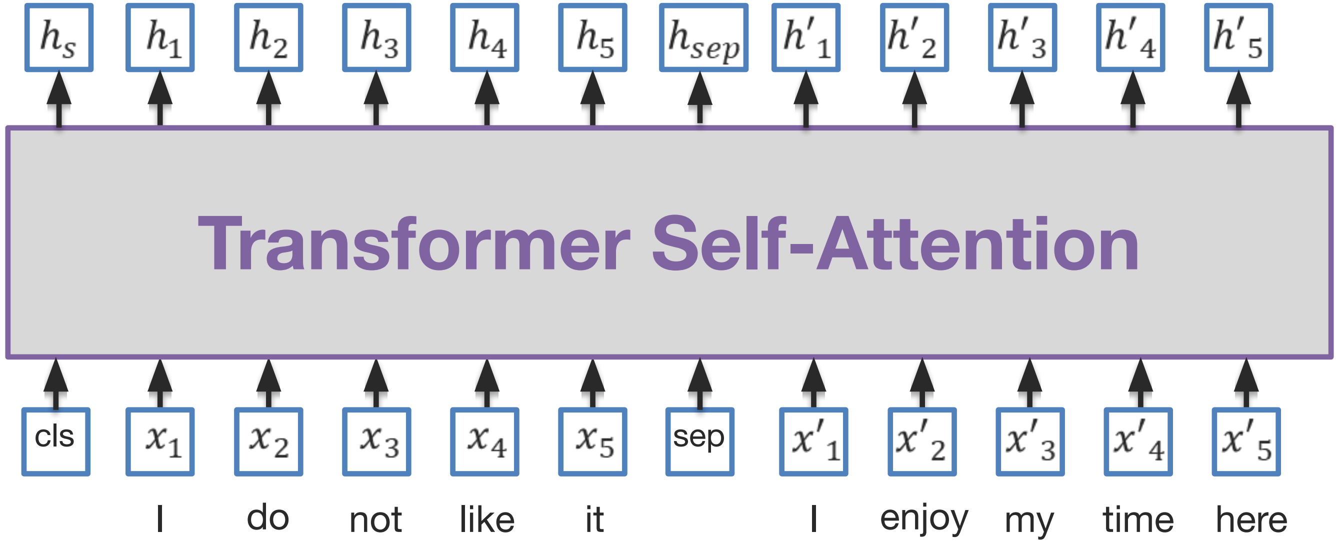


Fine-Tuning BERT

1 Sentence-level classification for only one sentence

Examples: sentiment analysis, document classification

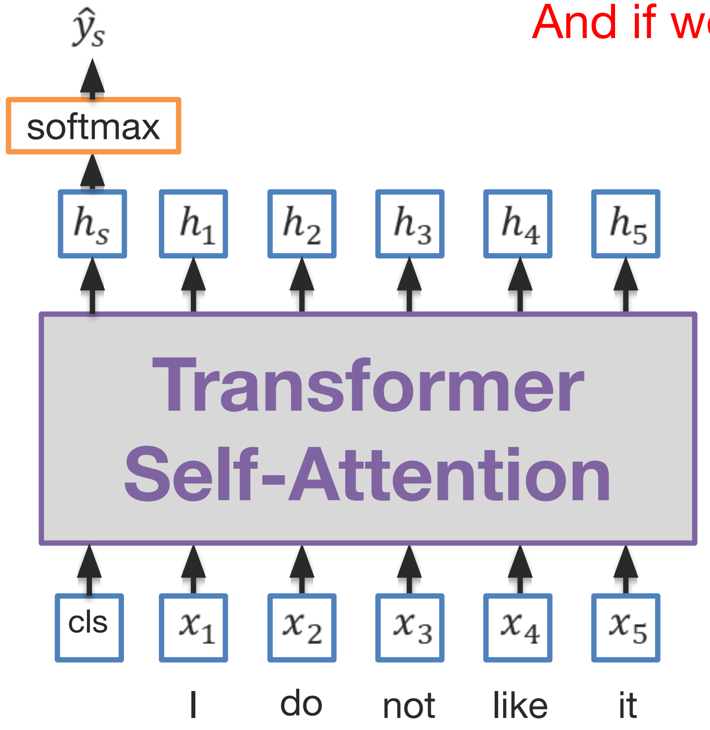
How?



Fine-Tuning BERT

1 Sentence-level classification for only one sentence

Examples: sentiment analysis, document classification

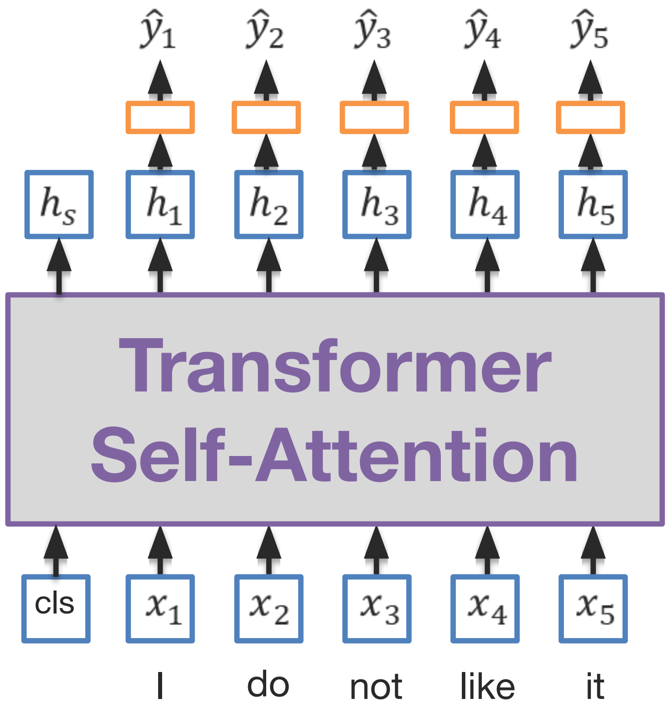


And if we have a label for each token?

Fine-Tuning BERT

2 Token-level classification for only one sentence

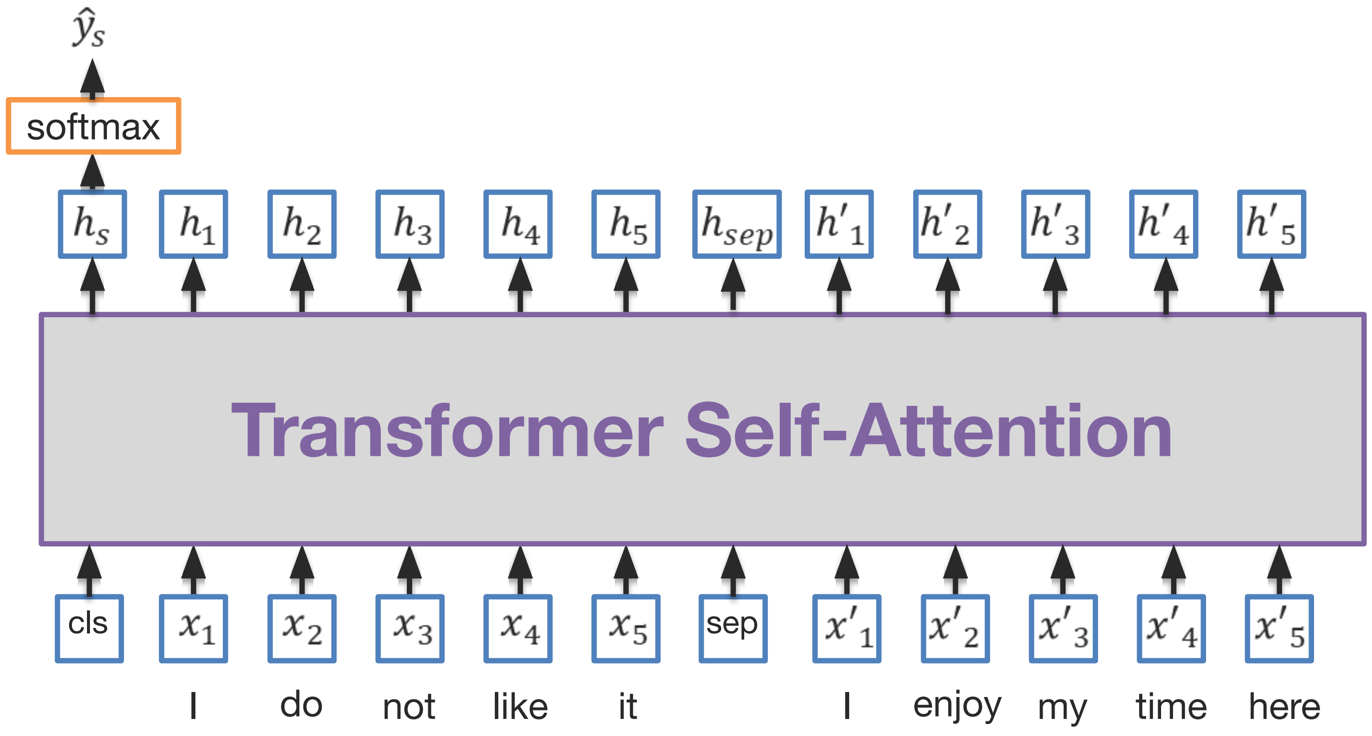
Examples: part-of-speech tagging, slot filling



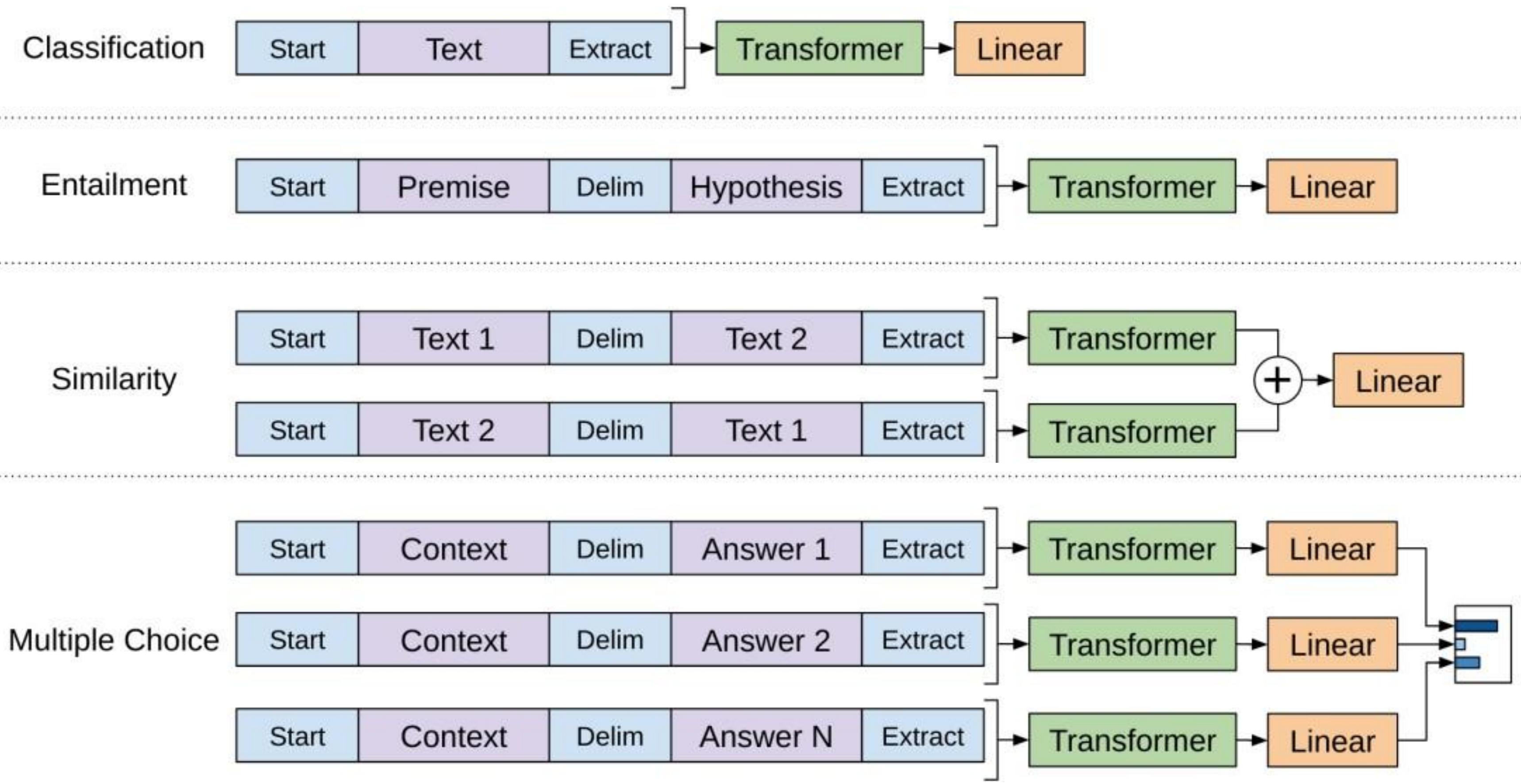
How to compare two sentences?

Fine-Tuning BERT

3. Sentence-level classification for two sentences
   Examples: natural language inference



Other Fine-tuning Approaches



https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

Resources

Resources

- spaCy (https://spacy.io/)
  - POS tagger, dependency parser, etc.
- Berkeley Neural Parser (Constituency Parser)
  - Software: https://github.com/nikitakit/self-attentive-parser
  - Demo: https://parser.kitaev.io/
- Stanford NLP software
  - Stanza: https://stanfordnlp.github.io/stanza/index.html
  - Others (some are outdated): https://nlp.stanford.edu/software/

Word Representation Resources

Word-level representations:

- Word2Vec (Google, 2013)
  - https://code.google.com/archive/p/word2vec/
- Glove (Stanford, 2014)
  - https://nlp.stanford.edu/projects/glove/
- FastText (Facebook, 2017)
  - https://fasttext.cc/

Contextual representations:

- ELMO (Allen Institute for AI, 2018)
  - https://allennlp.org/elmo
- BERT (Google, 2018)
  - https://github.com/google-research/bert
- RoBERTa (Facebook, 2019)
  - https://github.com/pytorch/fairseq

Factorizing co-occurrence matrix

Uses sub-word information (e.g. *walk* and *walking* are similar?)

Word representations are contextualized using all the words in the sentence. And sentence reps.

Lexicon-based Word Representation

LIWC: Language Inquiry & Word Count

Manually created dictionaries for different topics and categories:

- Function words: *pronouns, preposition, negation…*
- Affect words: *positive, negative emotions*
- Social words: *family, friends, referents*
- Cognitive processes: *Insight, cause, …*
- Perceptual processes: *Seeing, hearing, feeling*
- Biological processes: *Body, health/illness,…*
- Drives and needs*: Affiliation, achievement, …*
- Time orientation: *past, present, future*
- Relativity: *motion, space, time*
- Personal concerns: *work, leisure, money, religion …*
- Informal speech: *swear words, fillers, assent,…*

LIWC can encode individual words or full sentences.

https://liwc.wpengine.com/

➡ Commercial software. Contact TAs in advance if you would like to use it.