FDUVIS

VIS 2023

X-AGI

第 16 届中国 R 会议暨 2023 X-AGI 大会

TransforLearn: Interactive Visual Tutorial for the Transformer Model

Lin Gao[1], Zekai Shao[1], Ziqin Luo[1], Haibo Hu[2], Cagatay Turkay[3], Siming Chen[1,4]

[1] School of Data Science, Fudan University

[2] School of Big Data & Software Engineering, Chongqing University

[3] Centre for Interdisciplinary Methodologies, University of Warwick

[4] Shanghai Key Laboratory of Data Science



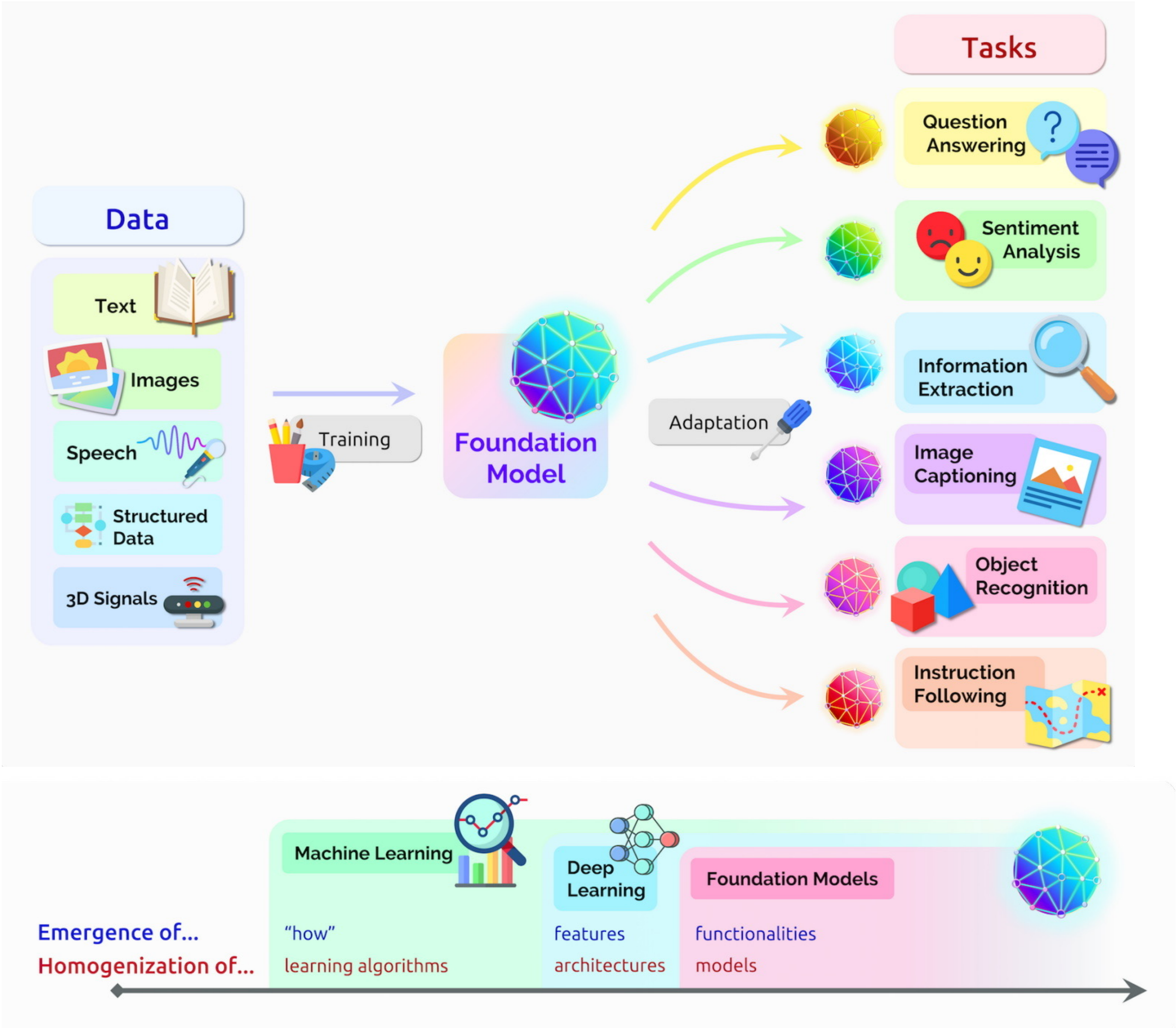



Background

Transformers are already used with many data sources for applications.

Transformers mark the next stage of AI's development, what some call the era of transformer AI.

The popularity of Transformer has sparked significant interest in learning its working mechanisms.



[1] https://blogs.nvidia.com/blog/what-is-a-transformer-model/

Background

Visualization for understanding deep learning models

- how the models make decisions & what they learned
- model improvement & debugging

M2lens [1] (TVCG 2021)

VEQA[2] (TVCG 2023)

for experts, not suitable as tutorial tools

[1] Wang X, He J, Jin Z, et al. M2lens: Visualizing and explaining multimodal models for sentiment analysis[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 28(1): 802-812.
[2] Shao Z, Sun S, Zhao Y, et al. Visual Explanation for Open-domain Question Answering with BERT[J]. IEEE Transactions on Visualization and Computer Graphics, 2023.

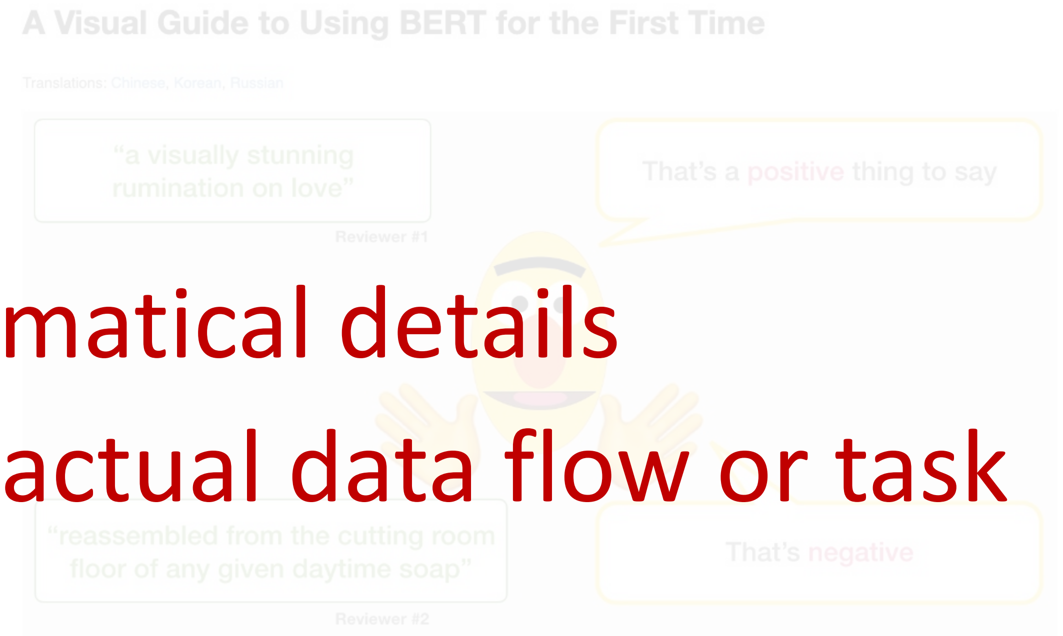
Background

Visual interpretation of Transformers

- interpretation of embedding and attention mechanisms

- blogs & videos for tutorial

Dodrio[1] (ACL 2021)



Jalammar's blogs [2]

lack of mathematical details

& interaction with the actual data flow or task

[1] Wang Z J, Turko R, Chau D H. Dodrio: Exploring transformer models with interactive visualization[J]. arXiv preprint arXiv:2103.14625, 2021.
[2] https://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/

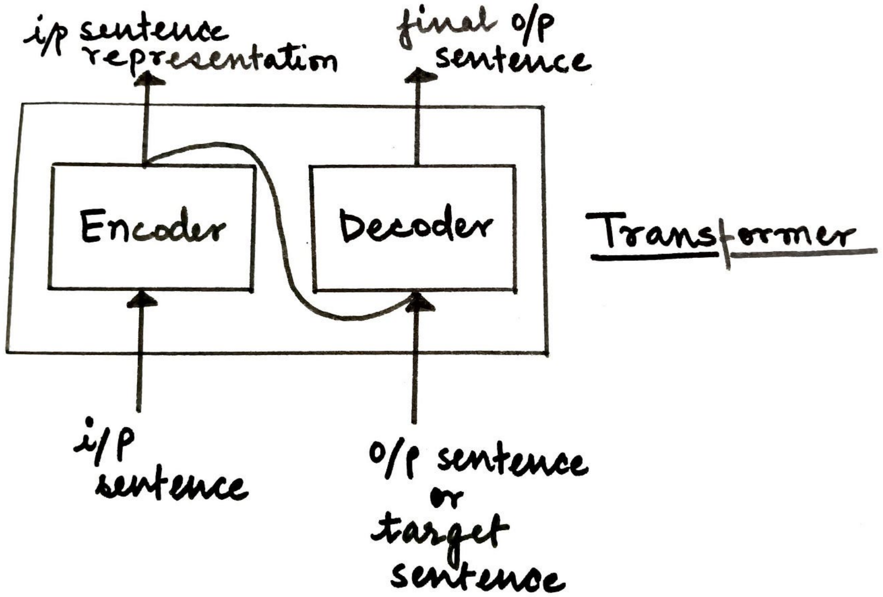
Preliminary Study

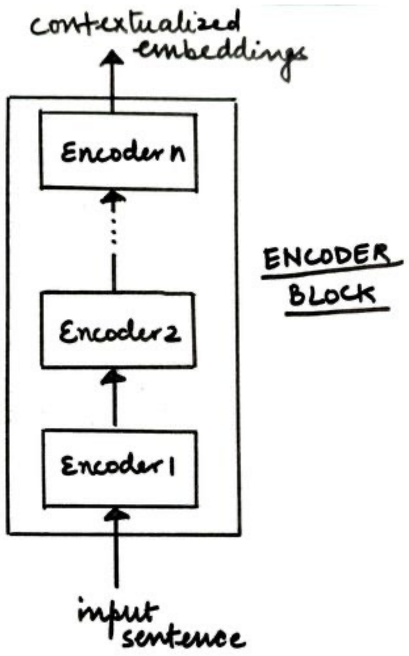
For lecturers, they need to manually break down Transformer into multiple steps and discuss them in a sequence of slides.

- Theoretical learning -> Dynamic thinking combined with practice

- Contextual interactions

- Class engagement





Preliminary Study

Beginners face difficulties in comprehending and learning Transformers due to its complex structure, data transformation and abstract downstream task.

- Encoder/Decoder, Attention ……

- Embedding, Dimension ……

- Alignment, Process ……

Calculation & Principles

Alignment of Output/Input

Layer Normalization

Sequence & Embedding

Multi-head Self-Attention

Input/Output Presentation

Segmentation & Scaling

Interaction & Combination

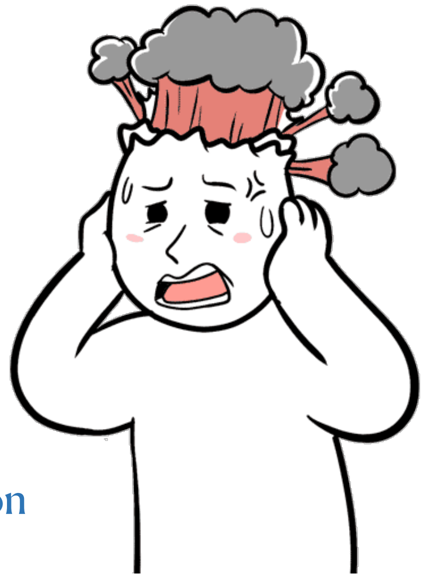
Specific Task Requirements

Data Flow Between Layers

Encoder-Decoder

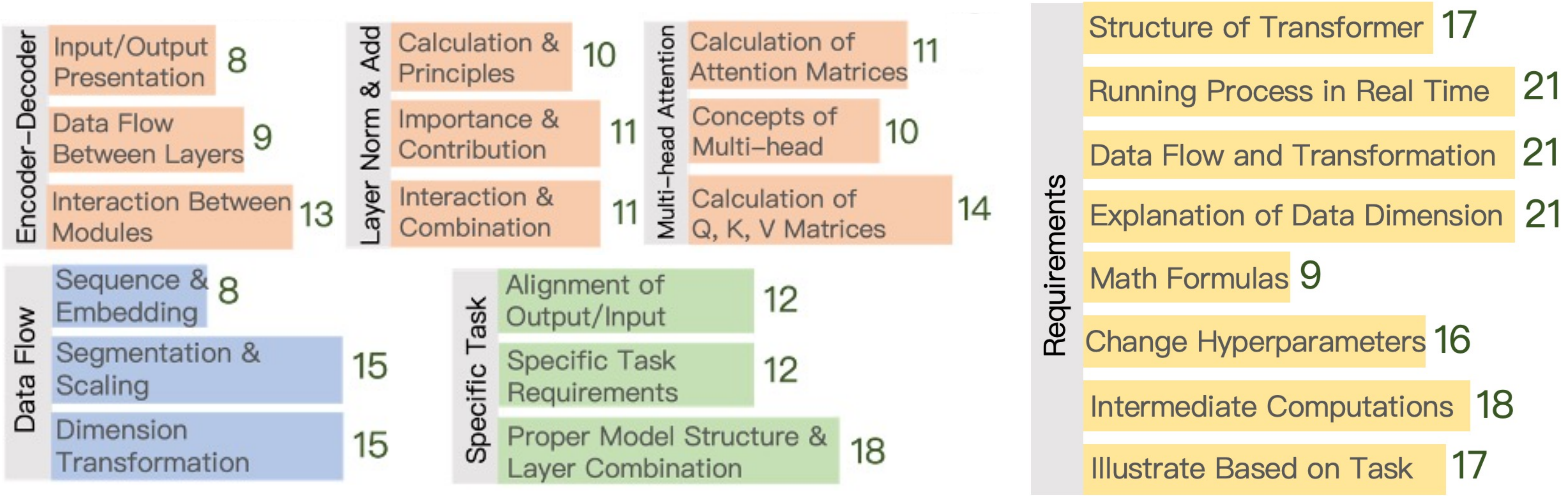
Proper Model Structure & Layer Combination

Dimension Transformation



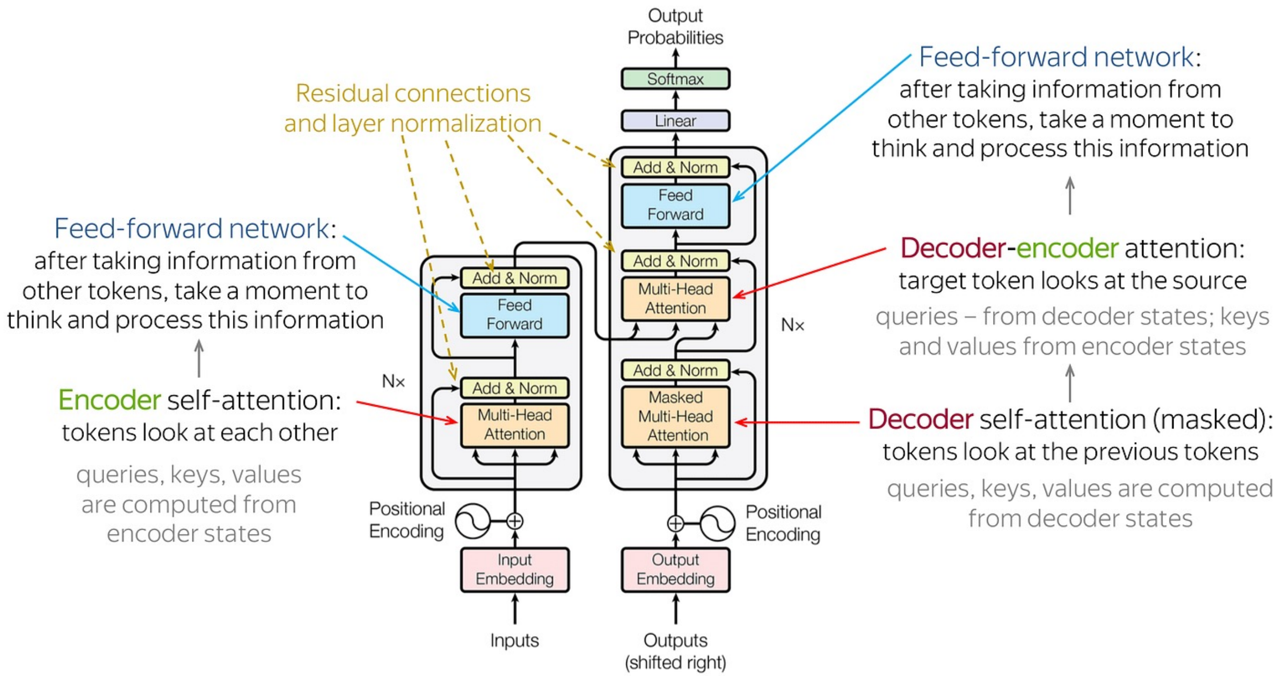
Preliminary Study

The survey asked about the key challenges in learning and applying Transformers from various aspects, and what features would be helpful in an interactive tool for beginners.



Consequently, an interactive visual tutorial is needed for deep learning beginners and non-experts to comprehensively learn about Transformers.

What can TransforLearn do?

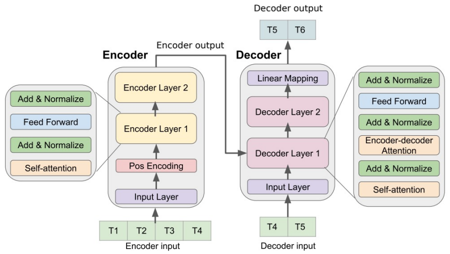


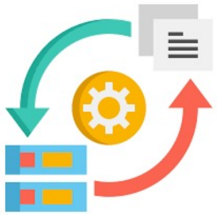
Why does Transformer has such a complex architecture[1]

[1] https://stats.stackexchange.com/questions/512242/why-does-transformer-has-such-a-complex-architecture

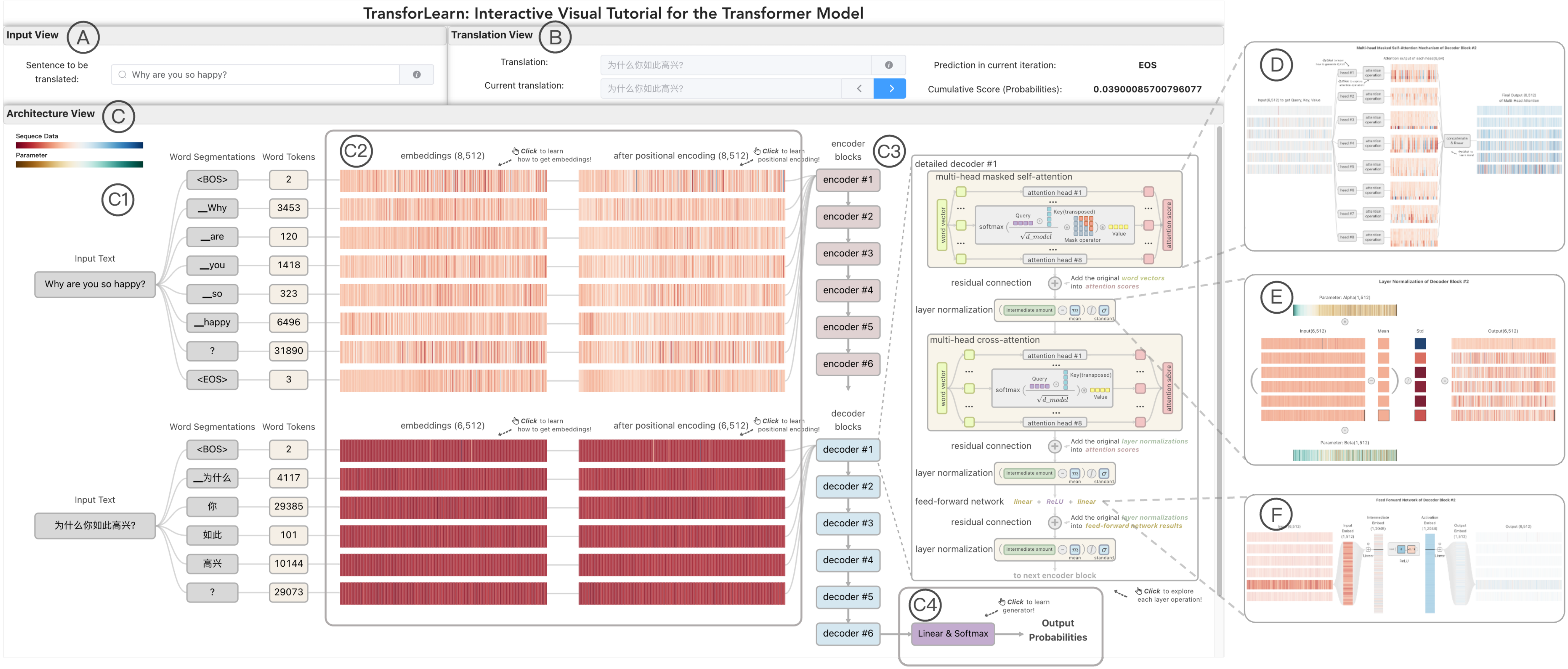
Tasks & Requirements

Task-1	Task-2	Task-3	Task-4
complex structure & layer operations	data flow & transformation	practical use in downstream tasks	guidance & feedback
Requirement-1	Requirement-2	Requirement-3	Requirement-4
A visual summary of the model architecture and data flow.	An interactive interface for layer operations and mathematical formulas.	Exploration mode between module levels based on downstream tasks.	Self-directed and immersive learning experiences.





Visual Design

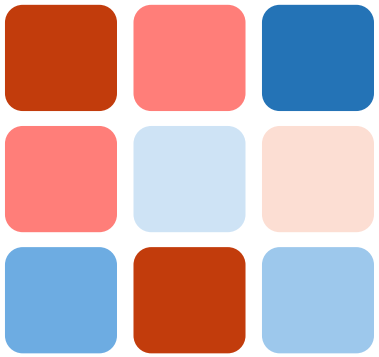


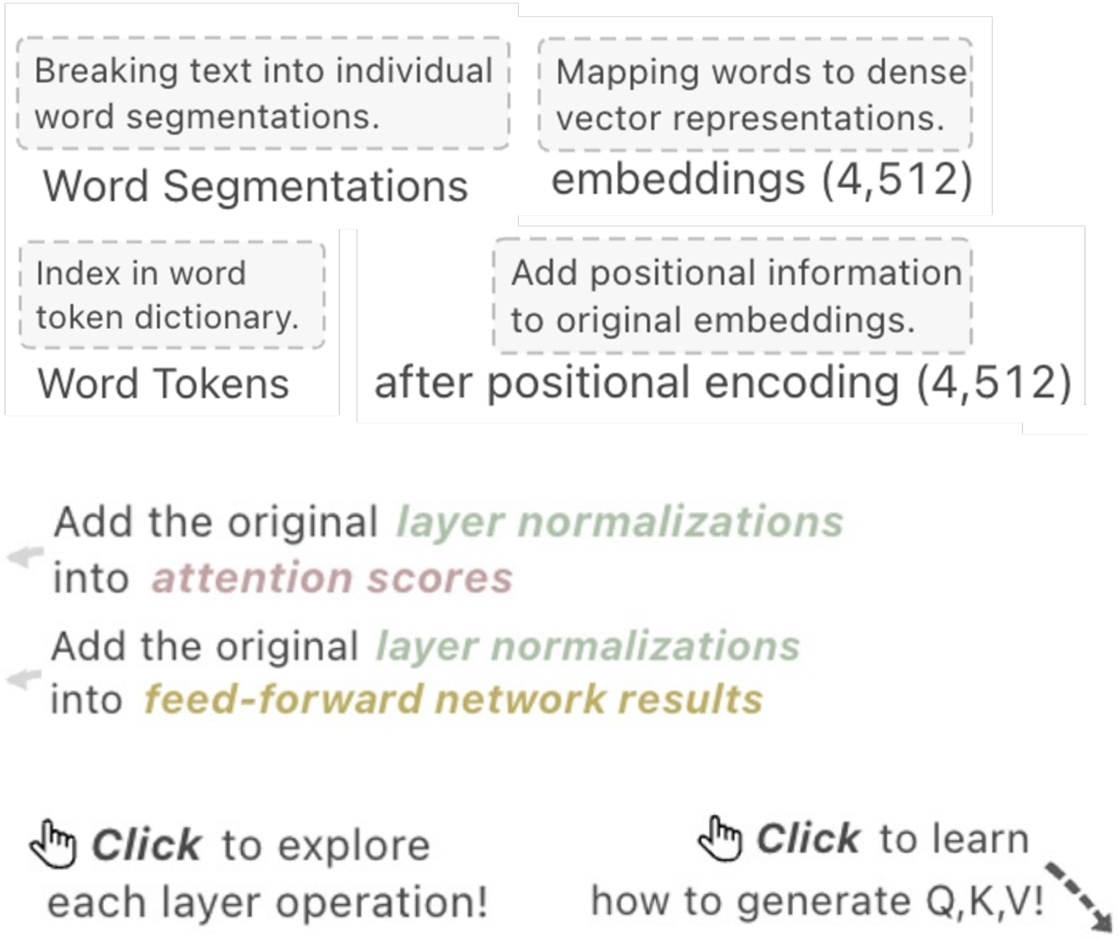
Visual Design - Overview

Architecture Overview ⇒ Module Detailed Views

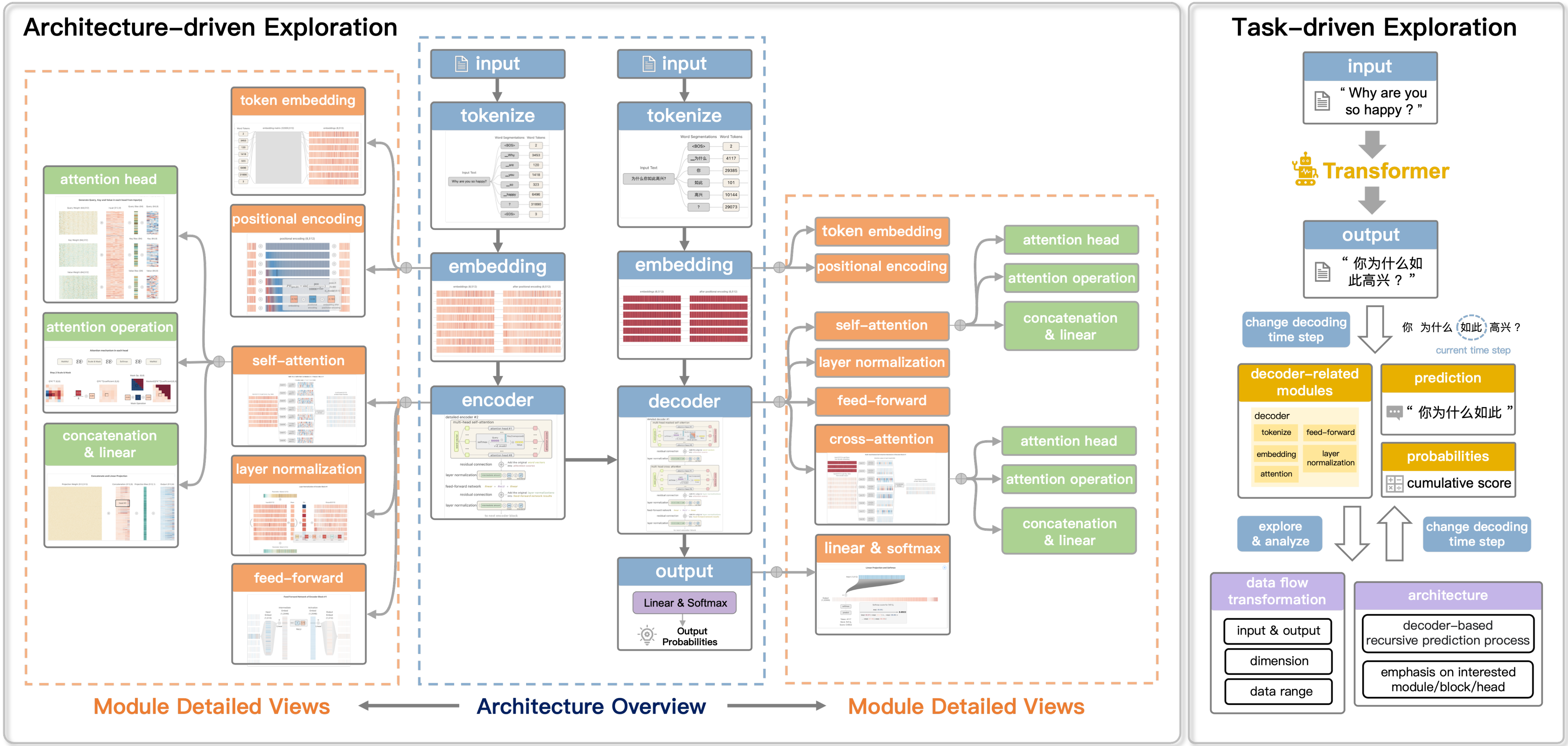
$$\begin{bmatrix} 11 & 20 & 109 \\ 21 & 54 & 37 \\ 74 & 11 & 60 \end{bmatrix}$$

Sequence Data
Parameter Data

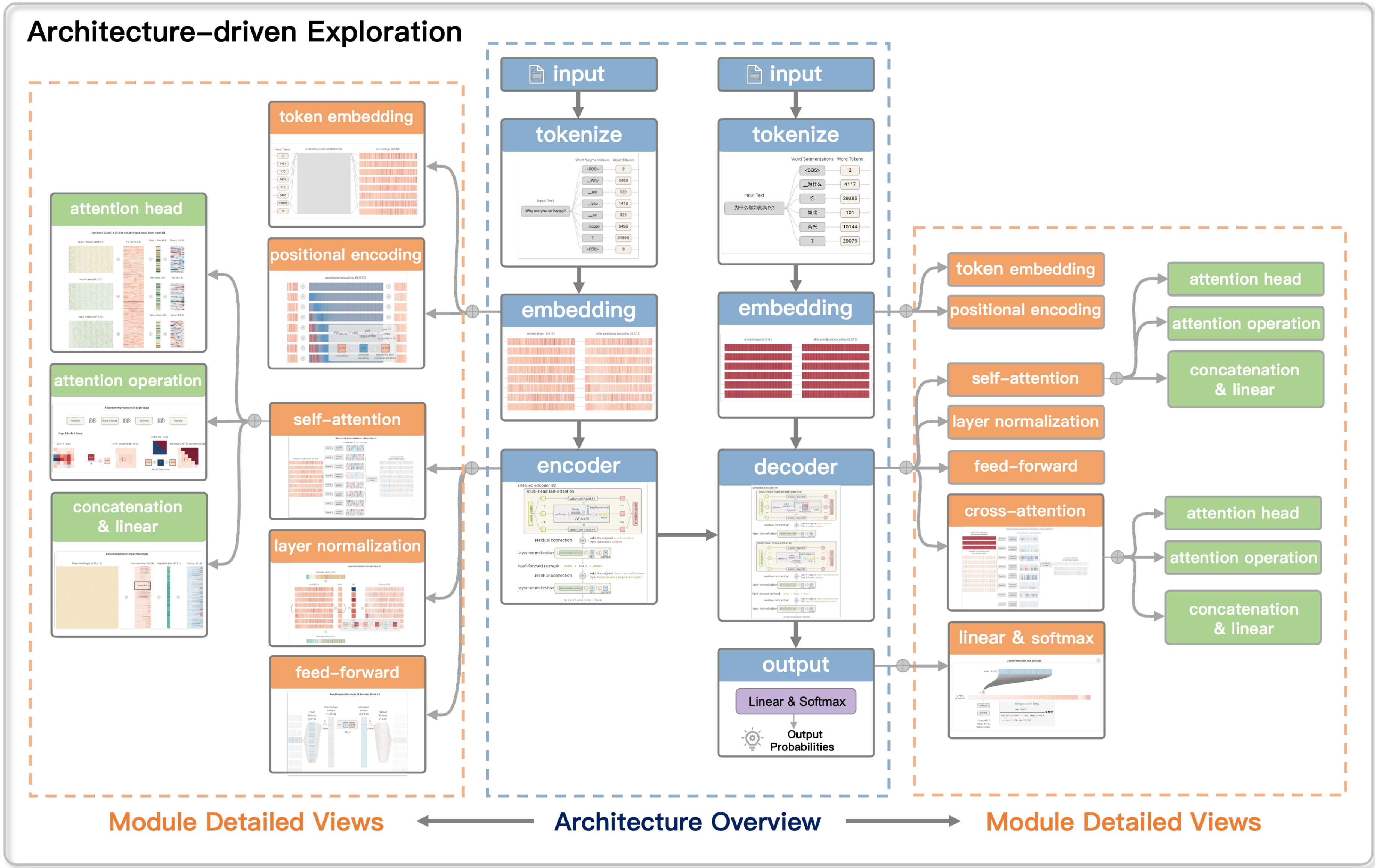


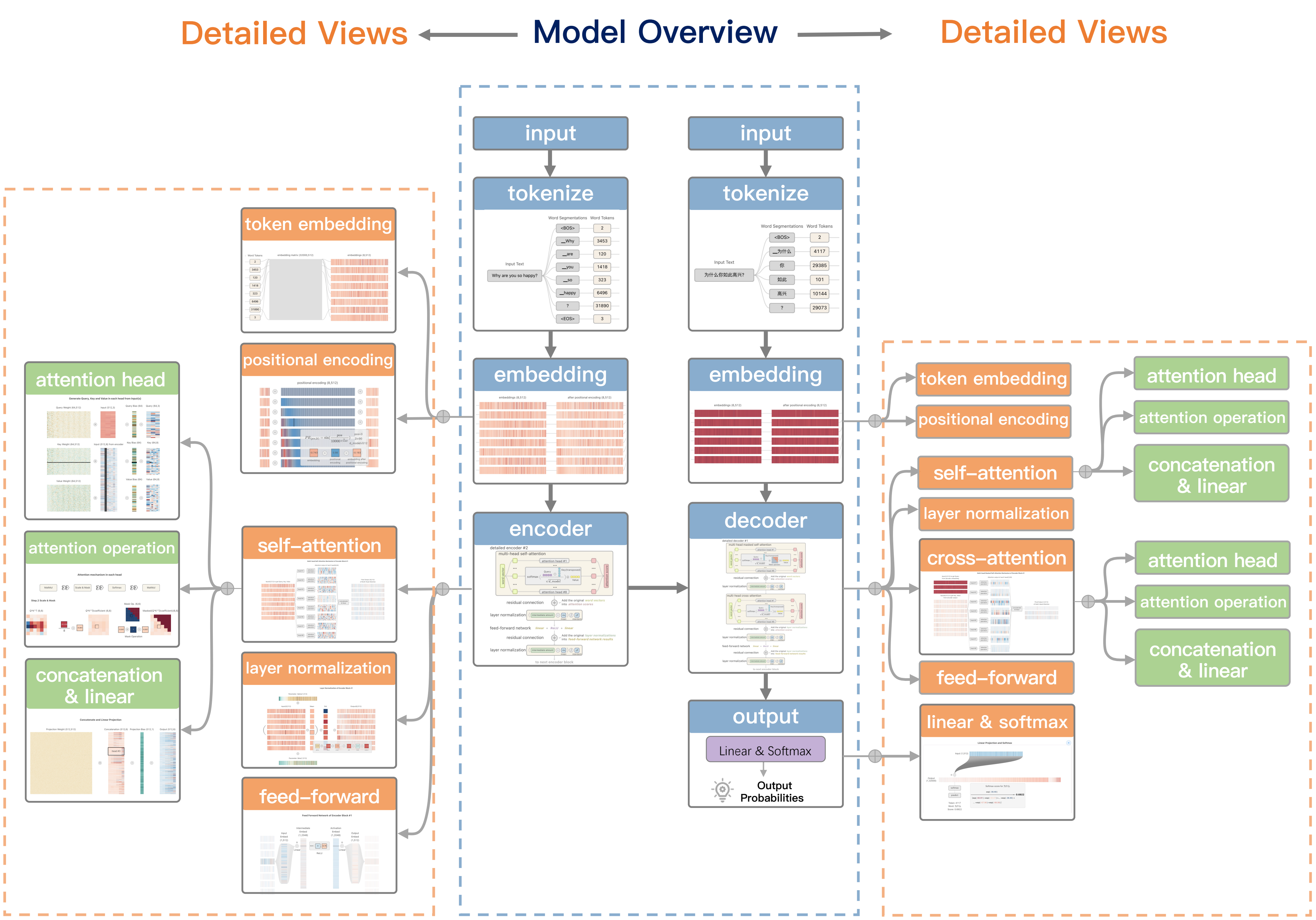


Visual Design - Overview

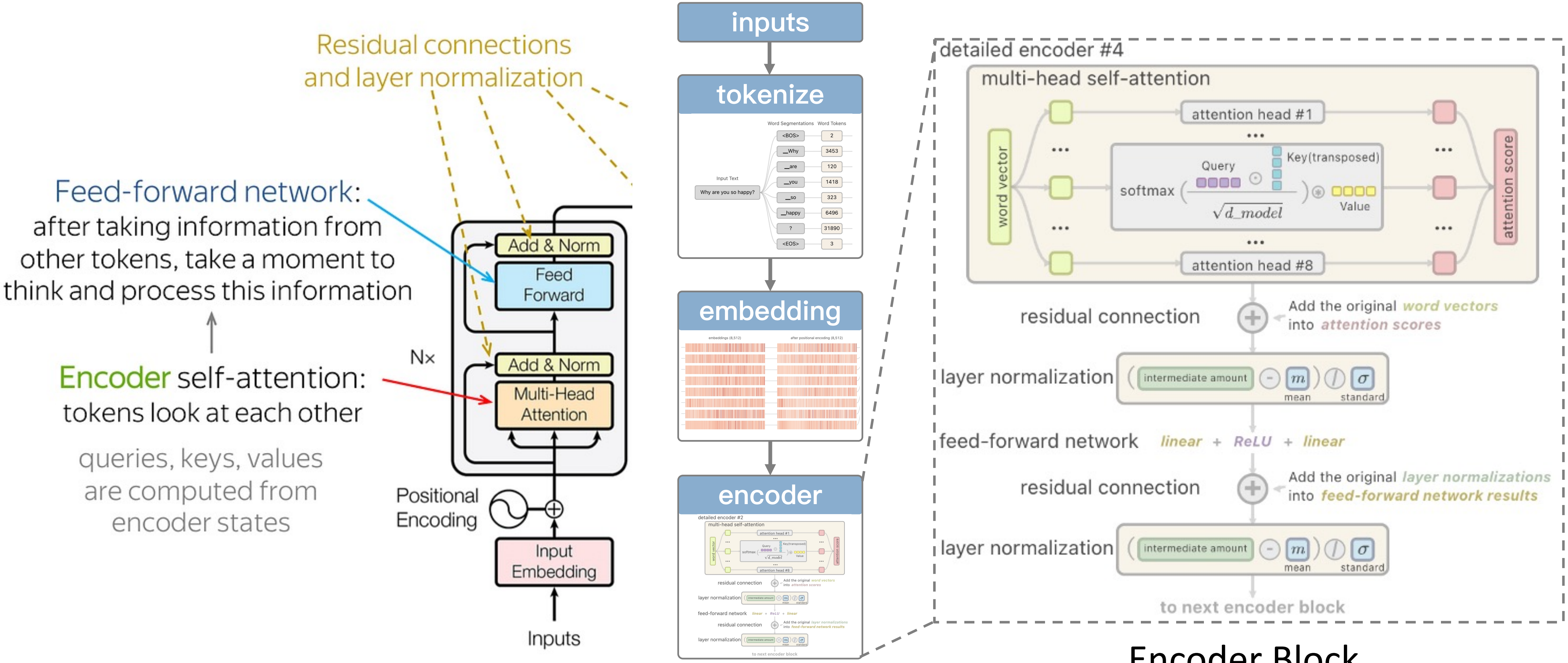


Architecture-driven Exploration



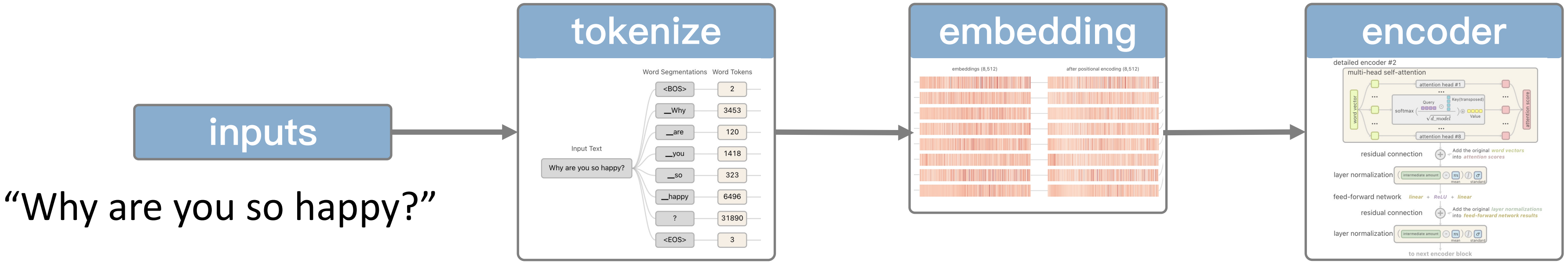

Detailed Views
Model Overview
Detailed Views
input
tokenize
embedding
encoder
input
tokenize
embedding
decoder
output
Linear & Softmax
Output Probabilities
token embedding
positional encoding
attention head
attention operation
concatenation & linear
self–attention
layer normalization
feed–forward
token embedding
positional encoding
self–attention
layer normalization
cross–attention
feed–forward
linear & softmax
attention head
attention operation
concatenation & linear
attention head
attention operation
concatenation & linear

Encoder



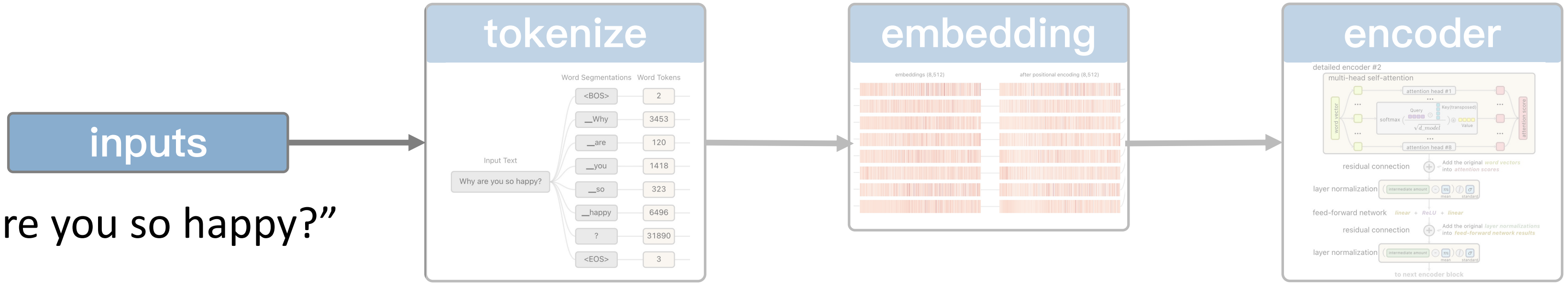
Encoder Block

Encoder



TransforLearn: Interactive Visual Tutorial for the Transformer Model

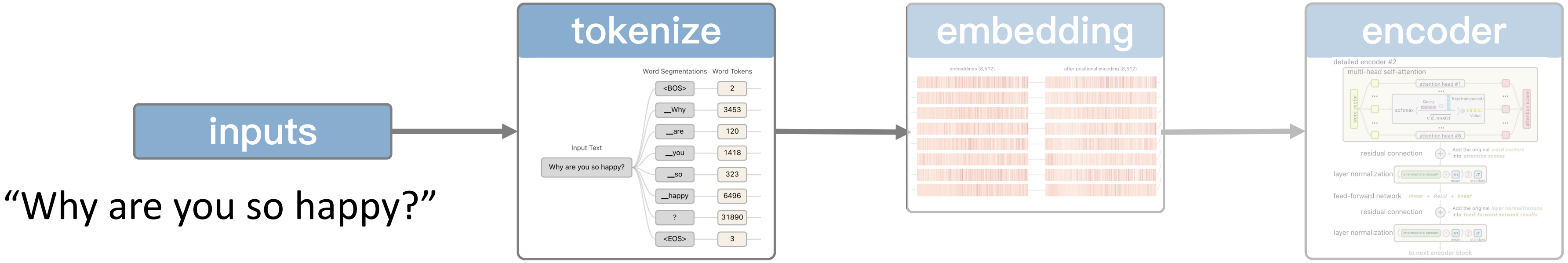
Encoder



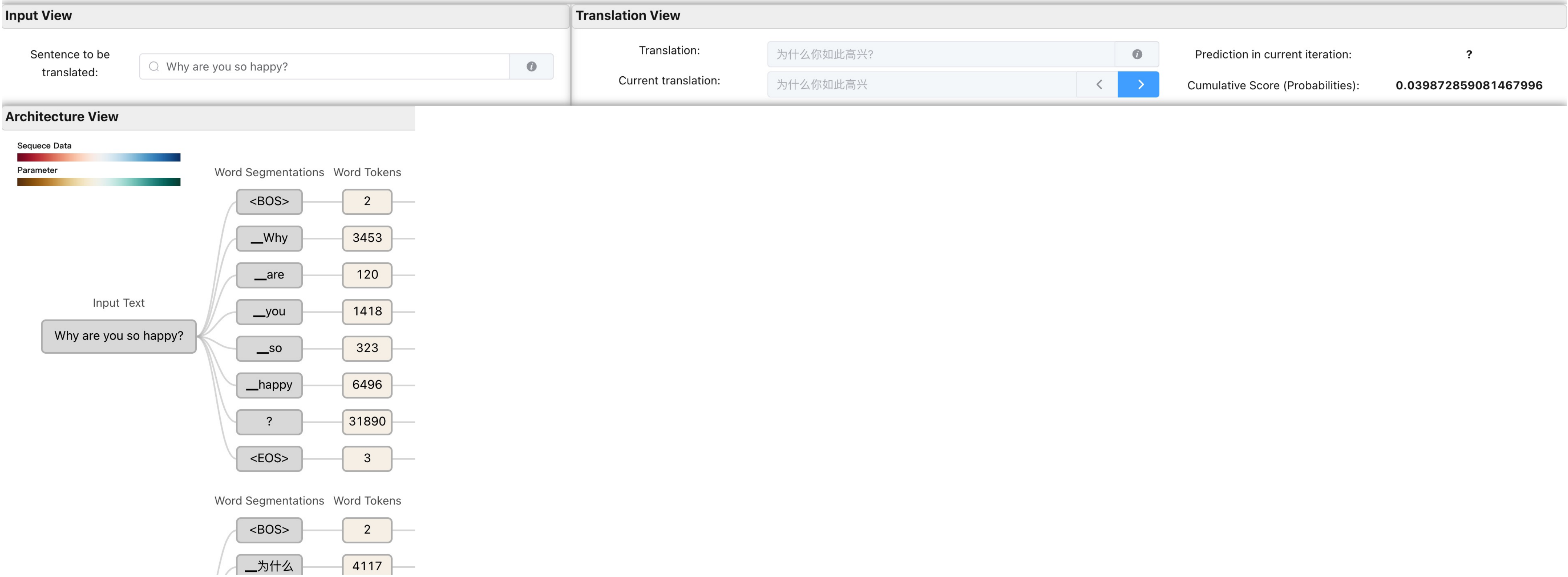
“Why are you so happy?”

TransforLearn: Interactive Visual Tutorial for the Transformer Model

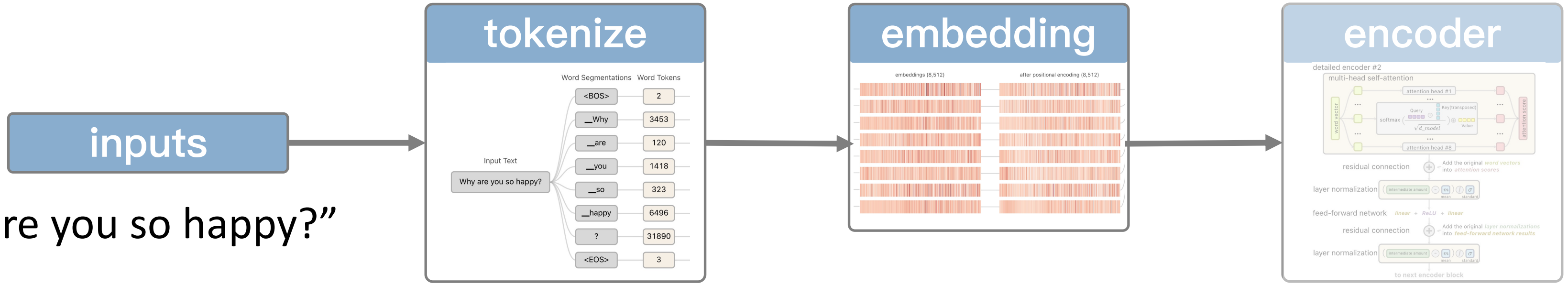
Encoder



TransforLearn: Interactive Visual Tutorial for the Transformer Model

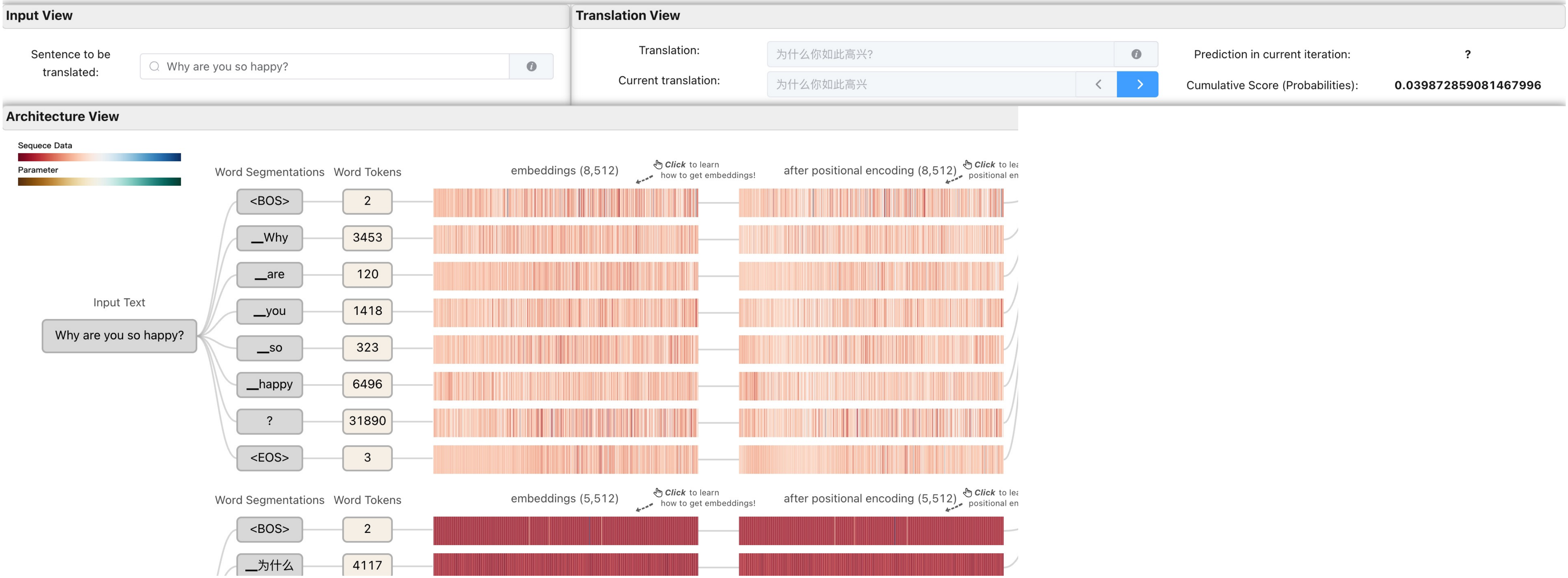


Encoder

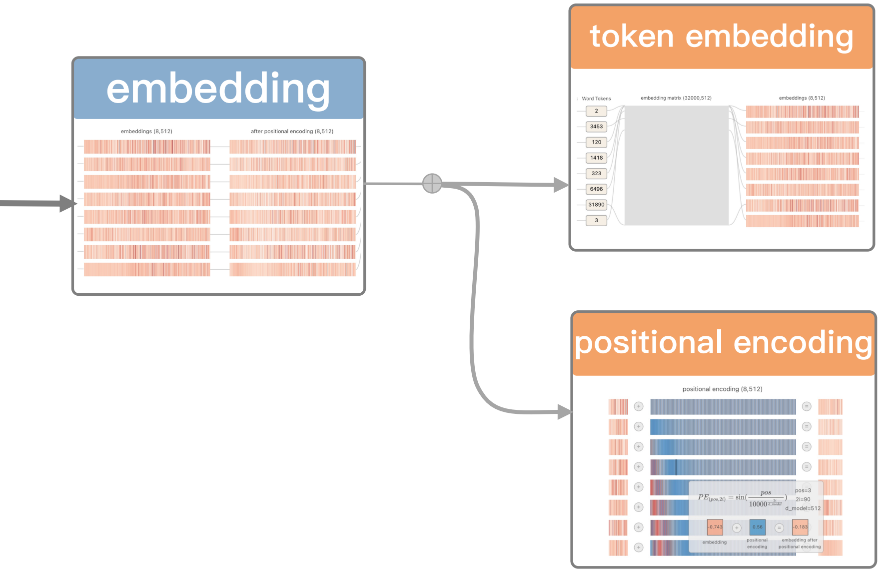


“Why are you so happy?”

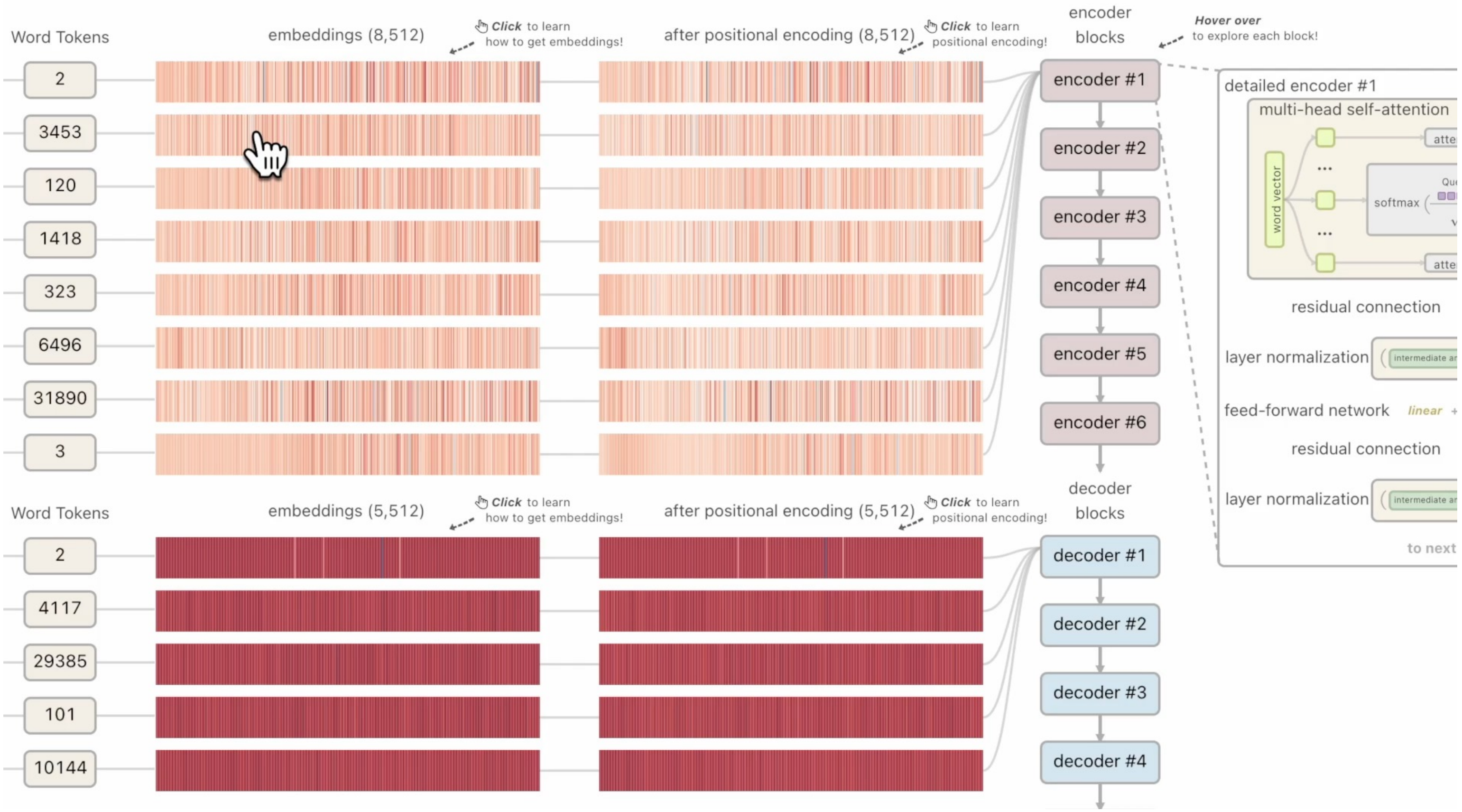
TransforLearn: Interactive Visual Tutorial for the Transformer Model



Encoder

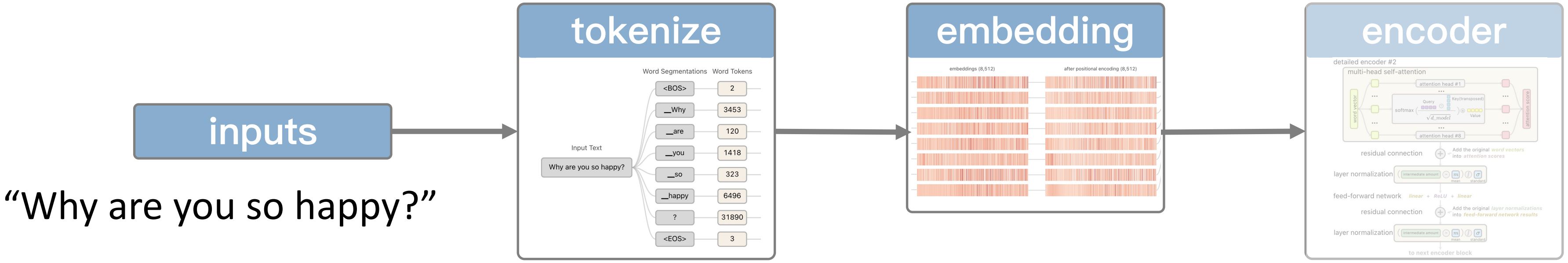


embedding
token embedding
positional encoding

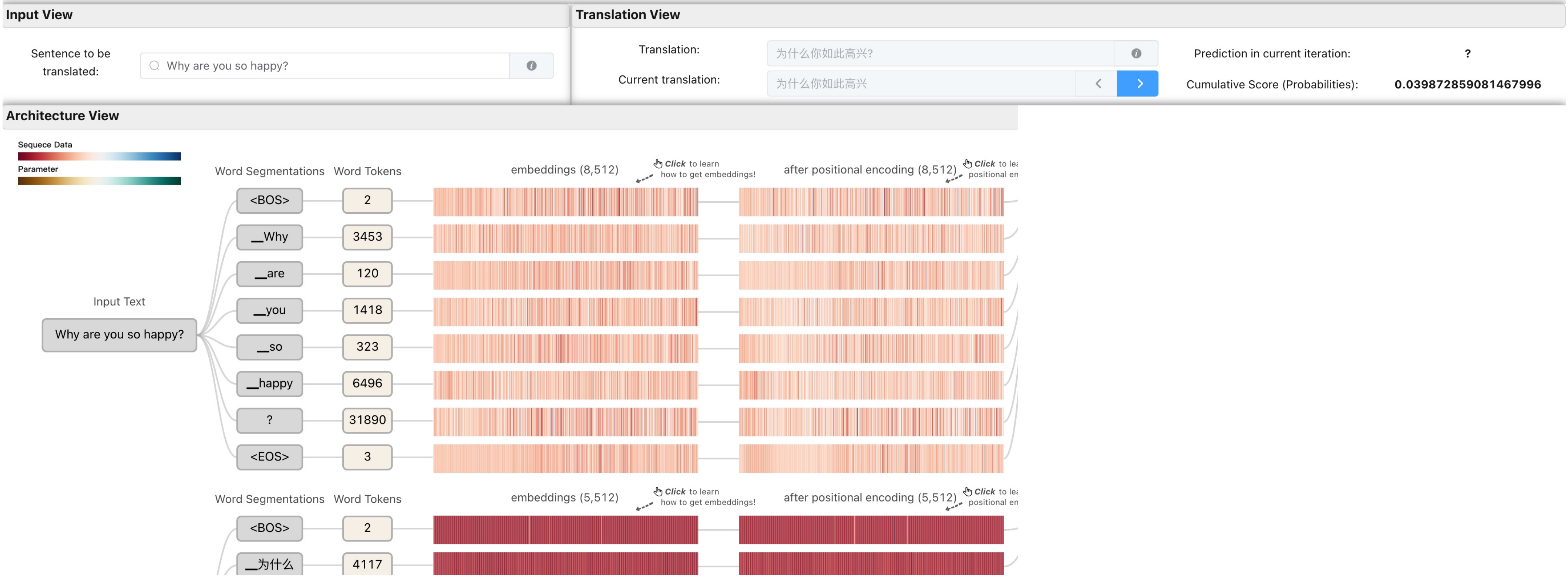


Word Tokens
embeddings (8,512)
Click to learn how to get embeddings!
after positional encoding (8,512)
Click to learn positional encoding!
encoder blocks
Hover over to explore each block!
2
3453
120
1418
323
6496
31890
3
encoder #1
encoder #2
encoder #3
encoder #4
encoder #5
encoder #6
detailed encoder #1
multi-head self-attention
word vector
softmax
residual connection
layer normalization
feed-forward network
linear
residual connection
layer normalization
to next
Word Tokens
embeddings (5,512)
Click to learn how to get embeddings!
after positional encoding (5,512)
Click to learn positional encoding!
decoder blocks
2
4117
29385
101
10144
decoder #1
decoder #2
decoder #3
decoder #4

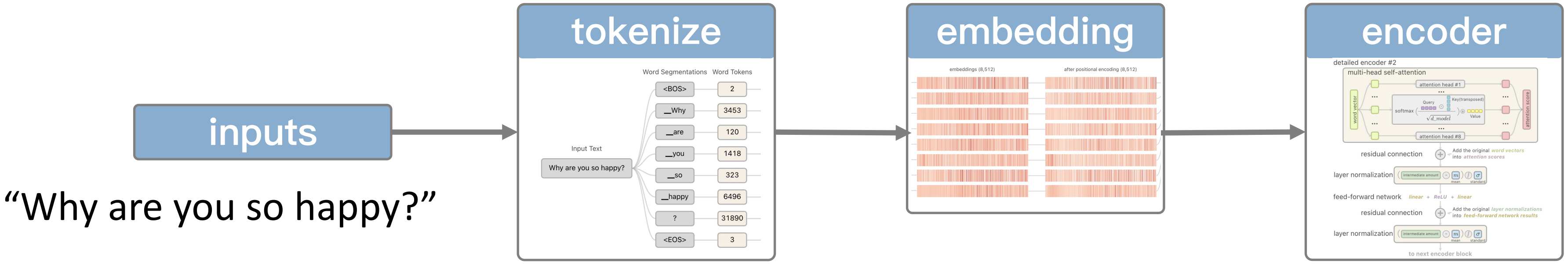
Encoder



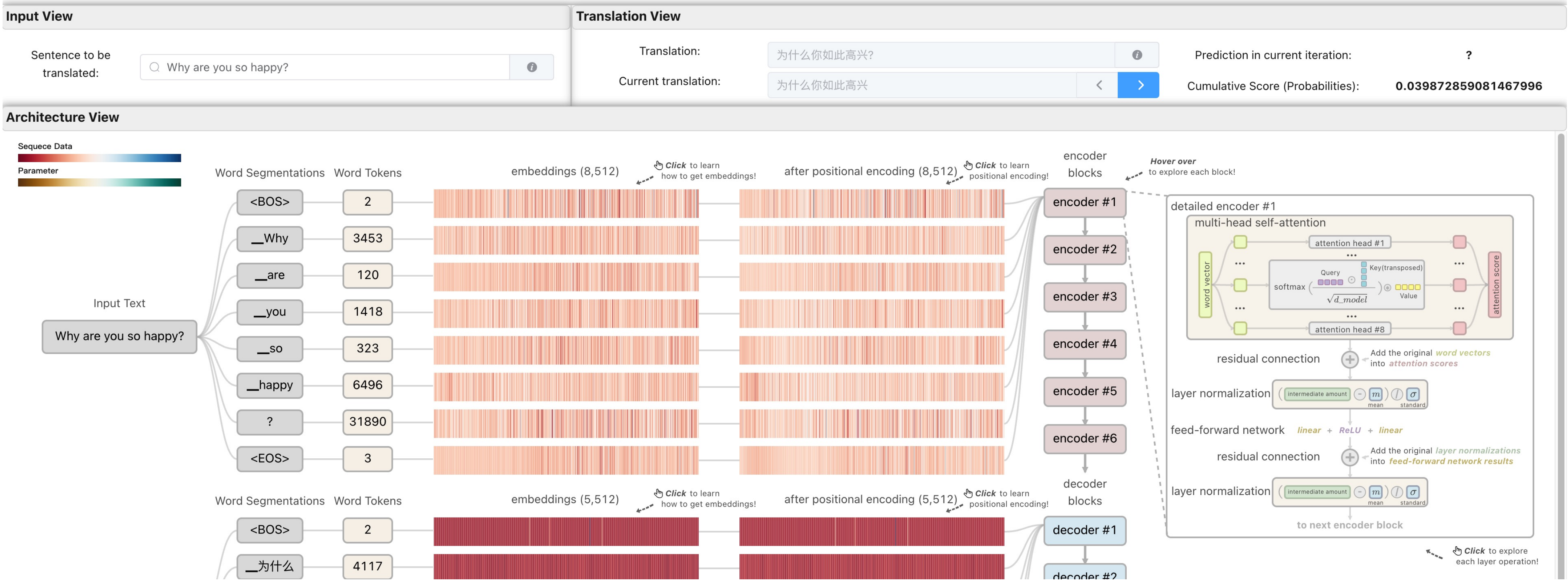
TransforLearn: Interactive Visual Tutorial for the Transformer Model



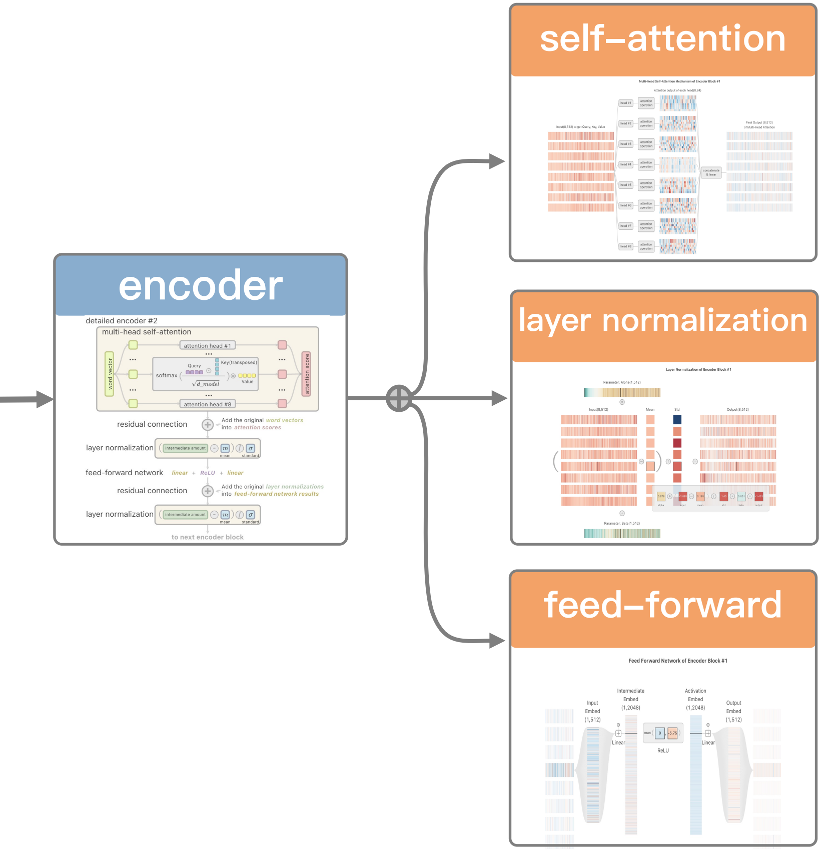
Encoder



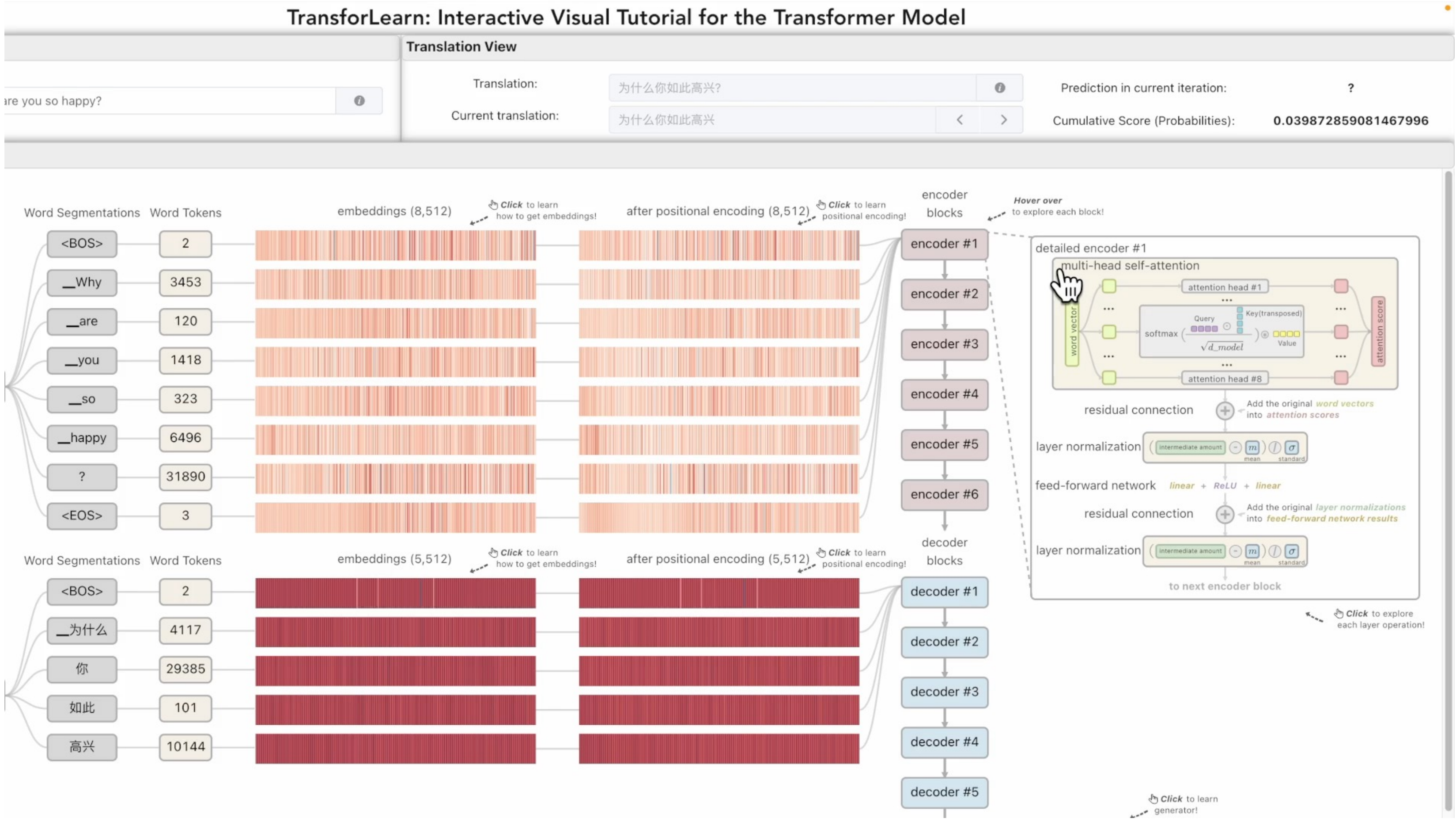
TransforLearn: Interactive Visual Tutorial for the Transformer Model



Encoder

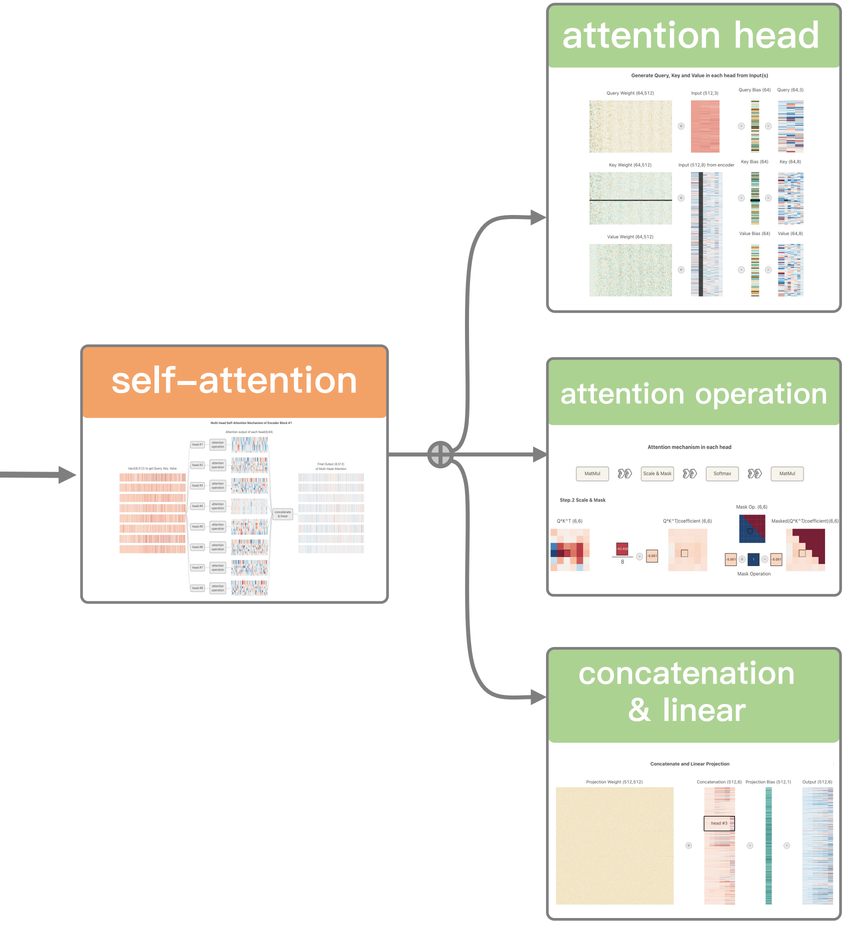


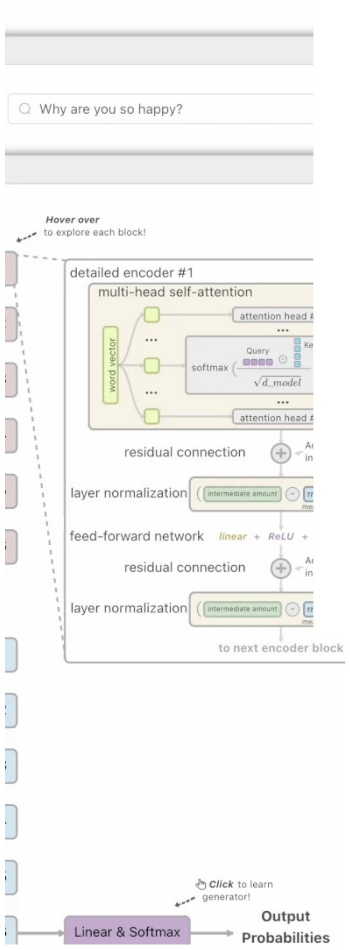
encoder
detailed encoder #2
multi-head self-attention
residual connection
layer normalization
feed-forward network
to next encoder block
self-attention
layer normalization
feed-forward

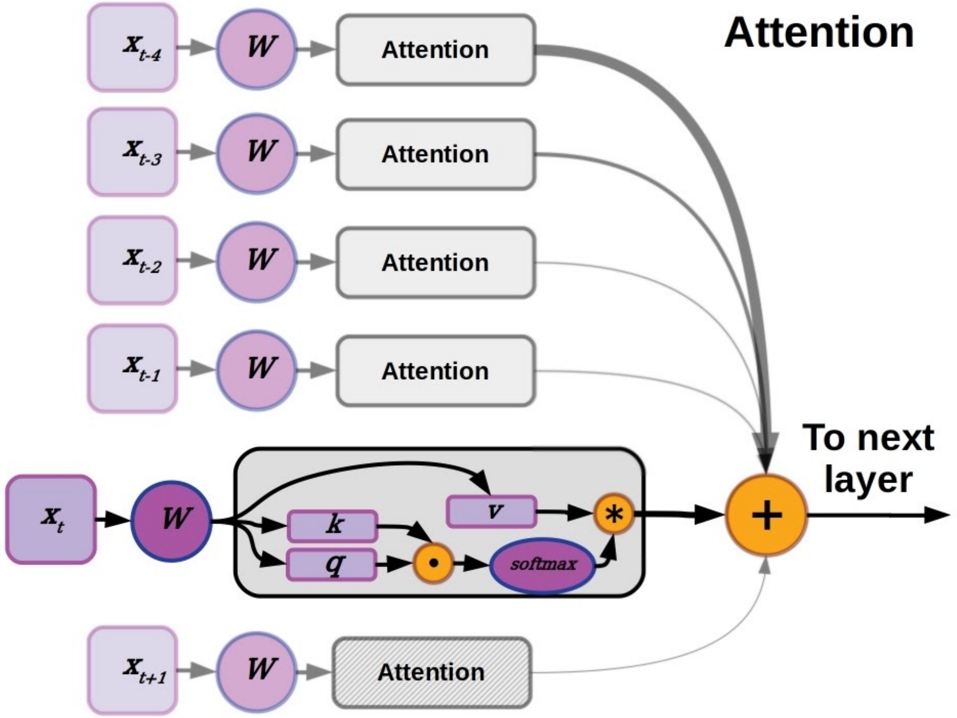


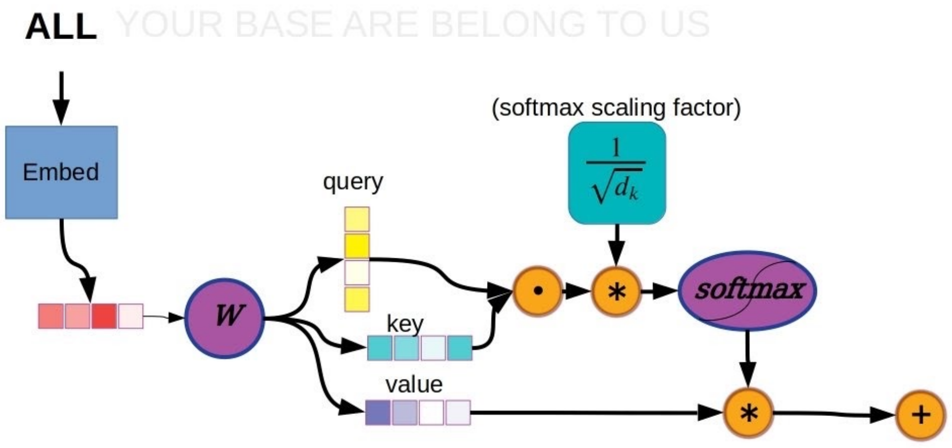
TransforLearn: Interactive Visual Tutorial for the Transformer Model
Translation View
re you so happy?
Translation:
为什么你如此高兴?
Current translation:
为什么你如此高兴
Prediction in current iteration:
?
Cumulative Score (Probabilities):
0.039872859081467996
Word Segmentations
Word Tokens
embeddings (8,512)
after positional encoding (8,512)
encoder blocks
Click to learn how to get embeddings!
Click to learn positional encoding!
Hover over to explore each block!
<BOS> 2
_Why 3453
_are 120
_you 1418
_so 323
_happy 6496
? 31890
<EOS> 3
encoder #1
encoder #2
encoder #3
encoder #4
encoder #5
encoder #6
embeddings (5,512)
after positional encoding (5,512)
decoder blocks
<BOS> 2
_为什么 4117
你 29385
如此 101
高兴 10144
decoder #1
decoder #2
decoder #3
decoder #4
decoder #5
detailed encoder #1
multi-head self-attention
attention head #1
attention head #8
softmax
Query
Key(transposed)
Value
word vector
attention score
residual connection
Add the original word vectors into attention scores
layer normalization
feed-forward network
linear + ReLU + linear
Add the original layer normalizations into feed-forward network results
to next encoder block
Click to explore each layer operation!
Click to learn generator!

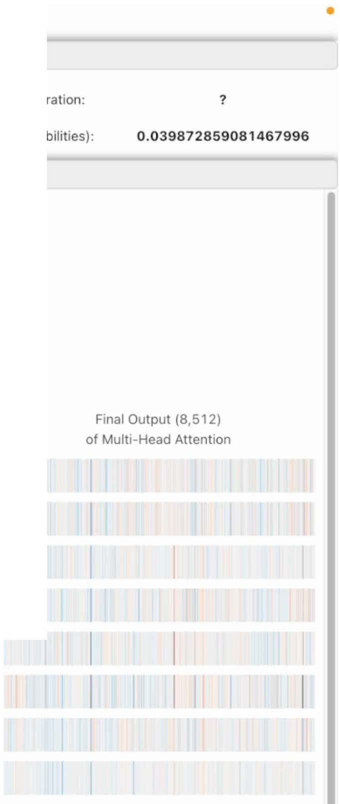
Encoder - Attention



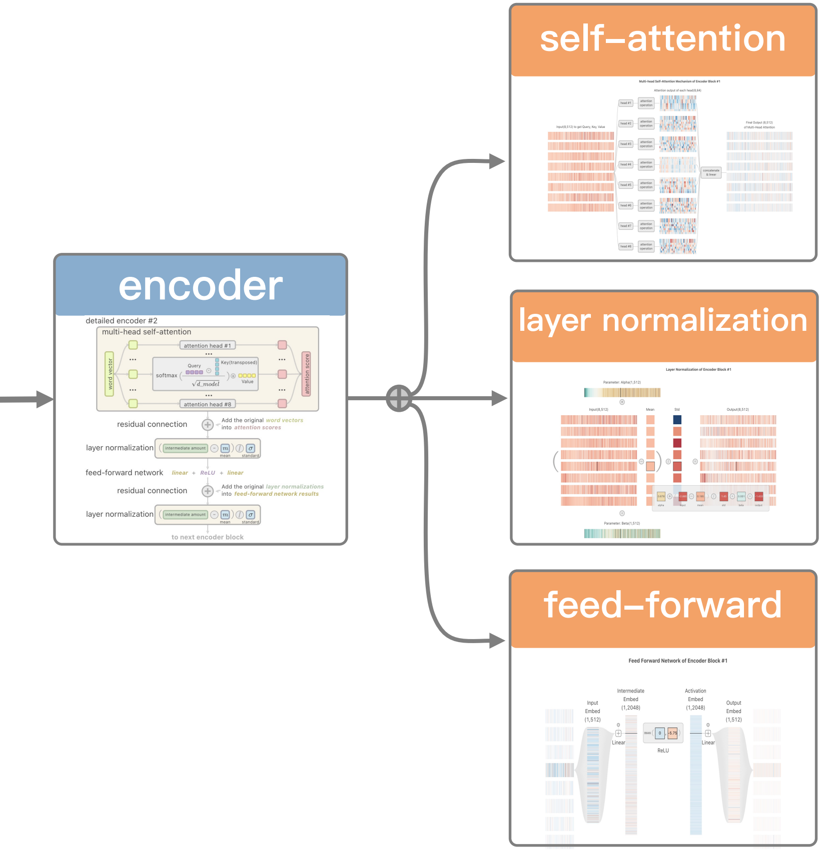


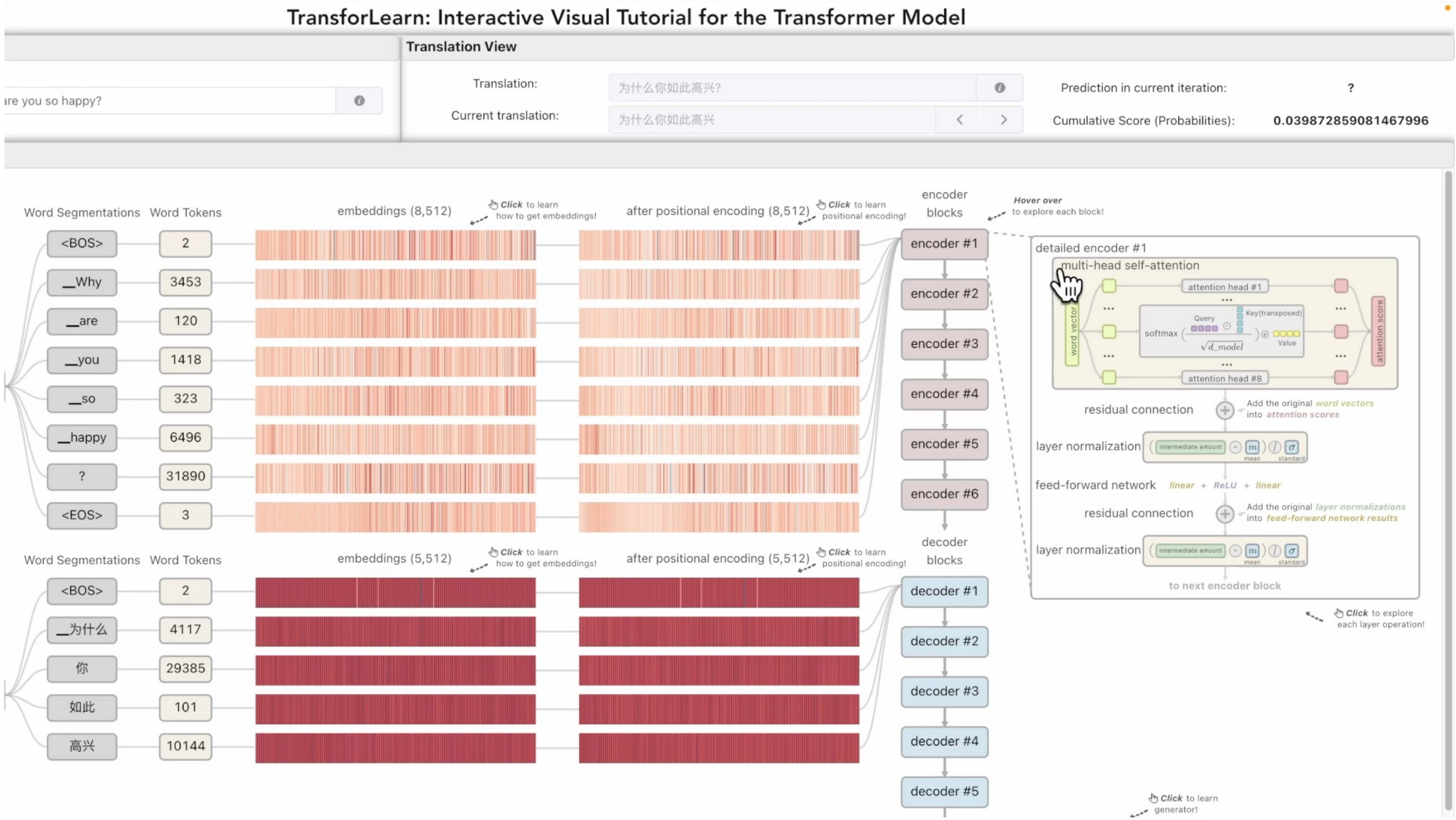




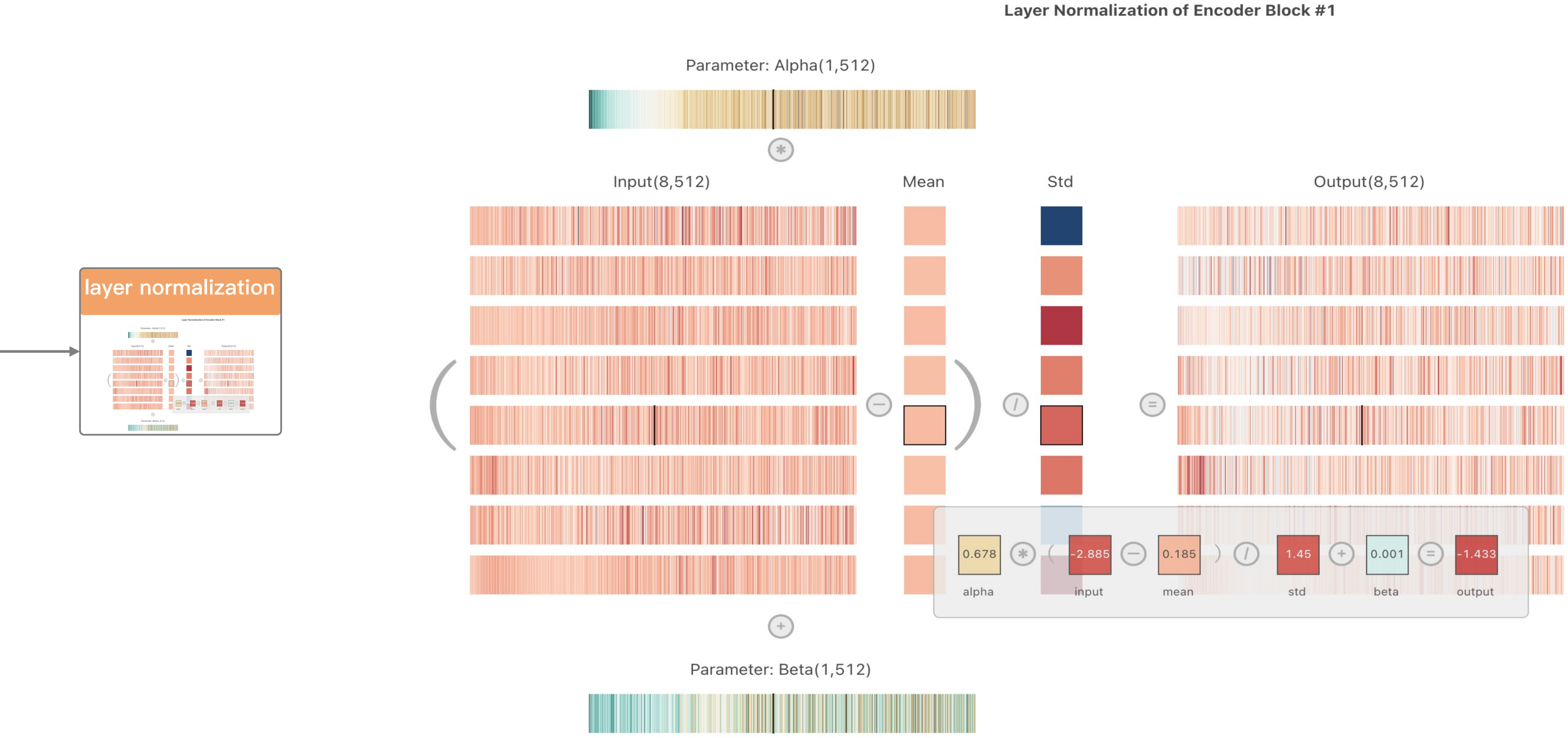


Encoder

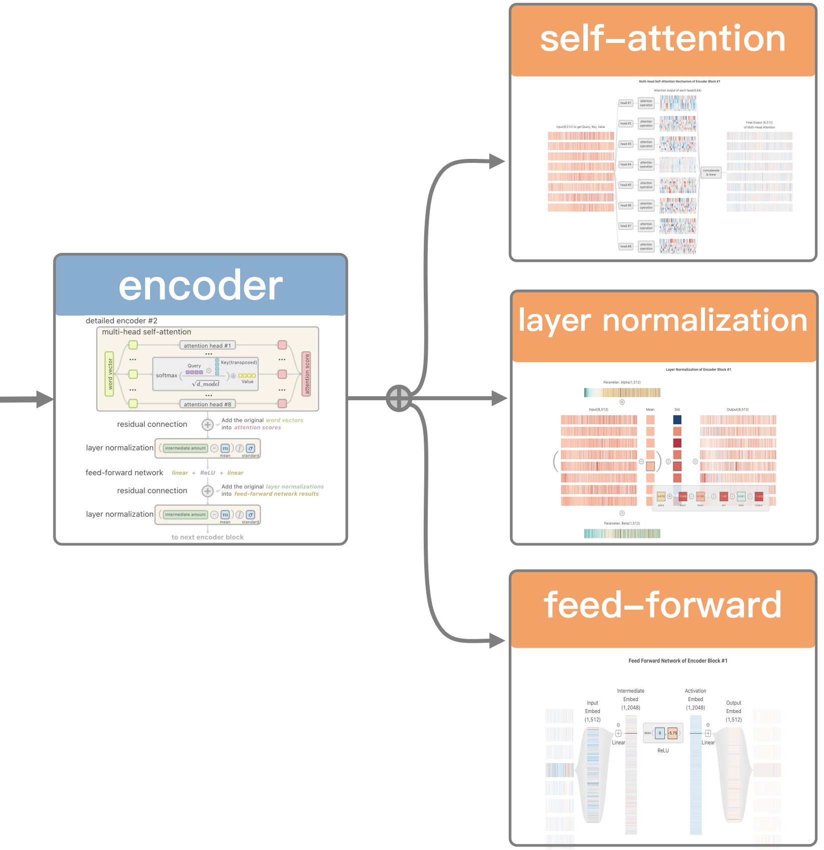

self-attention
layer normalization
feed-forward
encoder
detailed encoder #2
multi-head self-attention
residual connection
layer normalization
feed-forward network
residual connection
layer normalization
to next encoder block


TransforLearn: Interactive Visual Tutorial for the Transformer Model
Translation View
are you so happy?
Translation:
为什么你如此高兴?
Current translation:
为什么你如此高兴
Prediction in current iteration:
?
Cumulative Score (Probabilities):
0.039872859081467996
Word Segmentations
Word Tokens
embeddings (8,512)
Click to learn how to get embeddings!
after positional encoding (8,512)
Click to learn positional encoding!
encoder blocks
Hover over to explore each block!
<BOS> 2
_Why 3453
_are 120
_you 1418
_so 323
_happy 6496
? 31890
<EOS> 3
encoder #1
encoder #2
encoder #3
encoder #4
encoder #5
encoder #6
detailed encoder #1
multi-head self-attention
attention head #1
attention head #8
word vector
attention score
Query
Key(transposed)
Value
softmax
residual connection
Add the original word vectors into attention scores
layer normalization
intermediate amount
mean
standard
feed-forward network
linear + ReLU + linear
residual connection
Add the original layer normalizations into feed-forward network results
layer normalization
to next encoder block
Click to explore each layer operation!
Word Segmentations
Word Tokens
embeddings (5,512)
Click to learn how to get embeddings!
after positional encoding (5,512)
Click to learn positional encoding!
decoder blocks
<BOS> 2
_为什么 4117
你 29385
如此 101
高兴 10144
decoder #1
decoder #2
decoder #3
decoder #4
decoder #5
Click to learn generator!

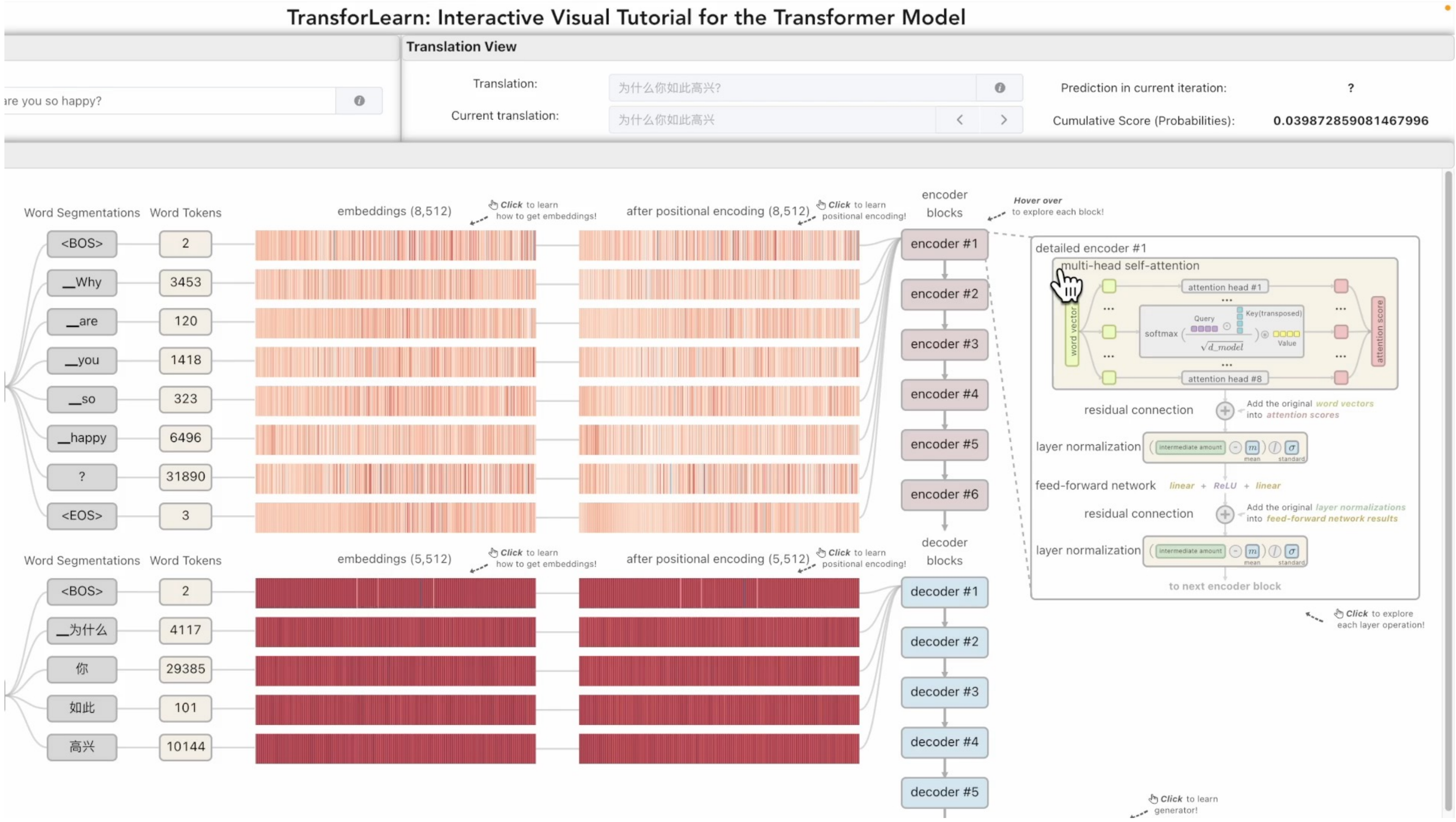
Encoder


layer normalization
Layer Normalization of Encoder Block #1
Parameter: Alpha(1,512)
Input(8,512)
Mean
Std
Output(8,512)
0.678
alpha
-2.885
input
0.185
mean
1.45
std
0.001
beta
-1.433
output
Parameter: Beta(1,512)

Encoder

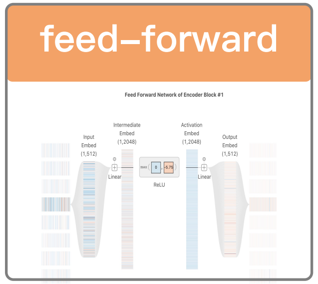


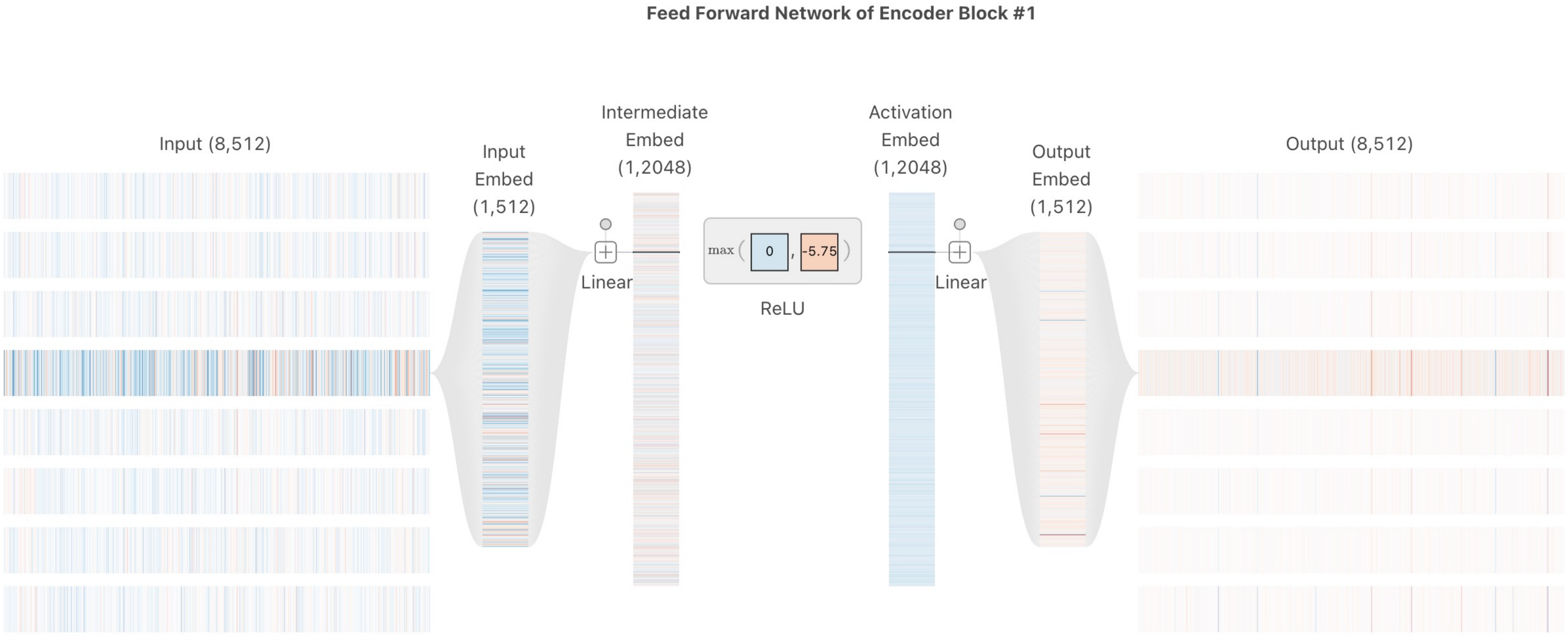
encoder
self-attention
layer normalization
feed-forward



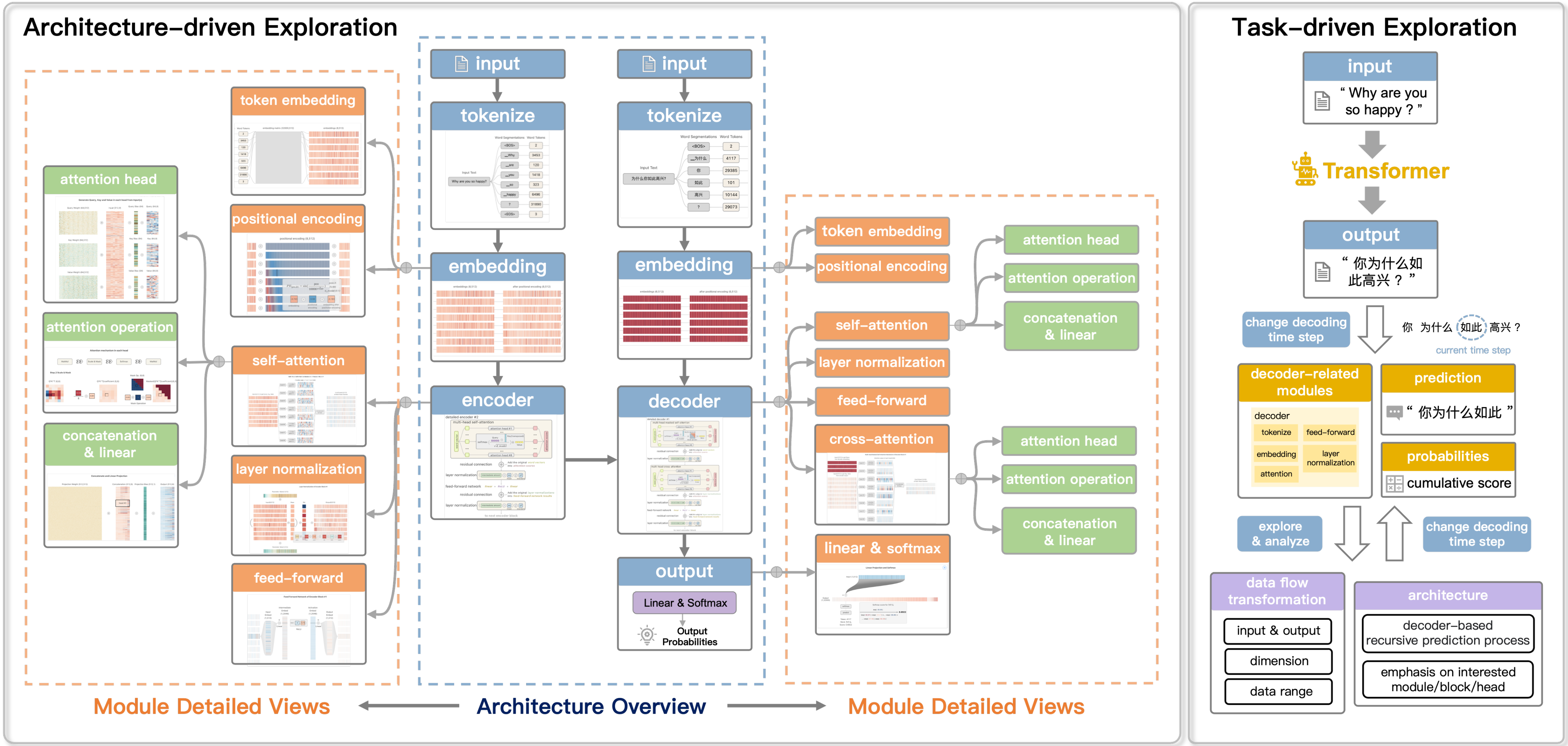
TransforLearn: Interactive Visual Tutorial for the Transformer Model
Translation View
re you so happy?
Translation:
为什么你如此高兴?
Current translation:
为什么你如此高兴
Prediction in current iteration:
?
Cumulative Score (Probabilities):
0.039872859081467996
Word Segmentations
Word Tokens
embeddings (8,512)
Click to learn how to get embeddings!
after positional encoding (8,512)
Click to learn positional encoding!
encoder blocks
Hover over to explore each block!
<BOS> 2
_Why 3453
_are 120
_you 1418
_so 323
_happy 6496
? 31890
<EOS> 3
encoder #1
encoder #2
encoder #3
encoder #4
encoder #5
encoder #6
detailed encoder #1
multi-head self-attention
attention head #1
attention head #8
word vector
attention score
residual connection
Add the original word vectors into attention scores
layer normalization
feed-forward network
linear + ReLU + linear
Add the original layer normalizations into feed-forward network results
to next encoder block
Click to explore each layer operation!
embeddings (5,512)
after positional encoding (5,512)
decoder blocks
<BOS> 2
_为什么 4117
你 29385
如此 101
高兴 10144
decoder #1
decoder #2
decoder #3
decoder #4
decoder #5
Click to learn generator!

Encoder

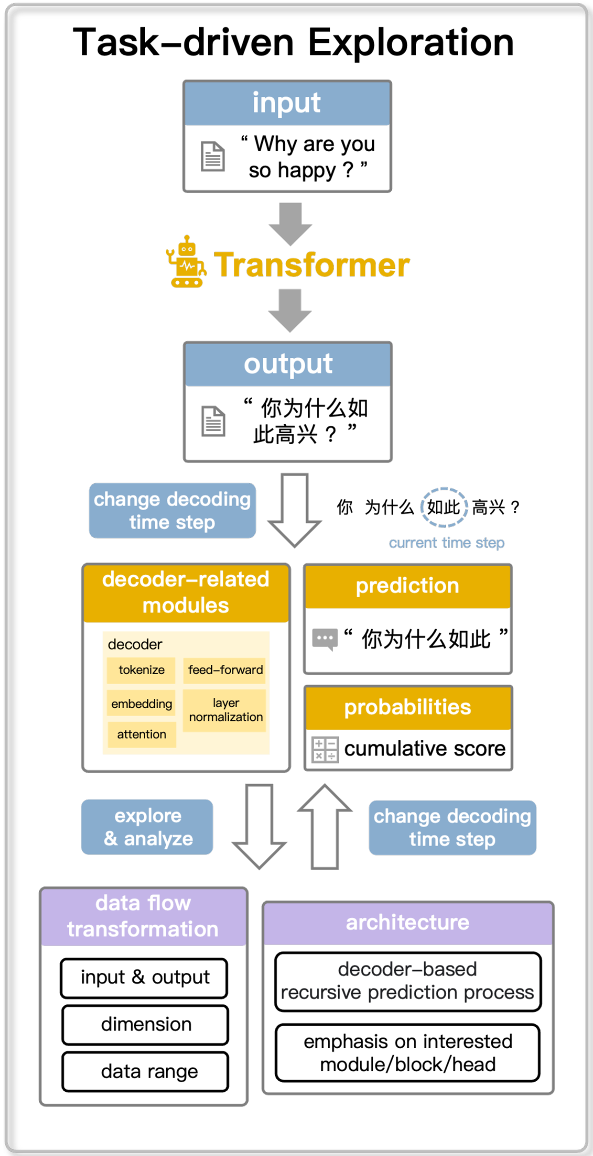

feed–forward


Feed Forward Network of Encoder Block #1
Input (8,512)
Input
Embed
(1,512)
Linear
Intermediate
Embed
(1,2048)
max (0 , -5.75)
ReLU
Activation
Embed
(1,2048)
Linear
Output
Embed
(1,512)
Output (8,512)

Visual Design - Overview



Task- driven Exploration



Explore data flow changes

- Input and output, data dimension, data range

Analyze structural features

- Decoding time step -> translation progress
- Focus on a specific module or head

Task- driven Exploration

Explore data flow changes

- Input and output, data dimension, data range

Analyze structural features

- Decoding time step -> translation progress
- Focus on a specific module or head

data flow transformation

architecture

Click to change
the decoder time step

Usage Scenario

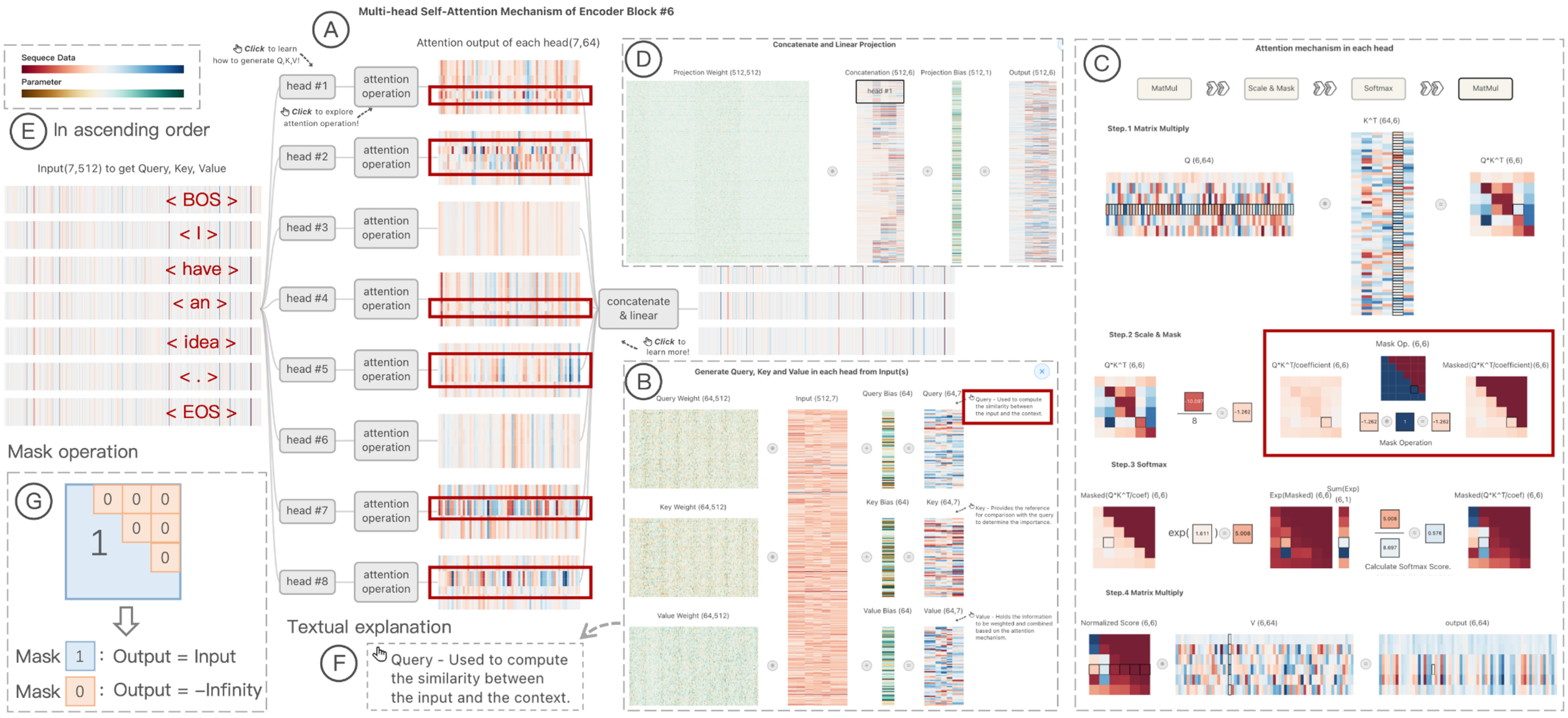
Self-study guidance for a beginner

- utilize Transformer to extract features from sequence data
- the concept and generation process of the Q, K, and V matrices
- the use of decoders for prediction

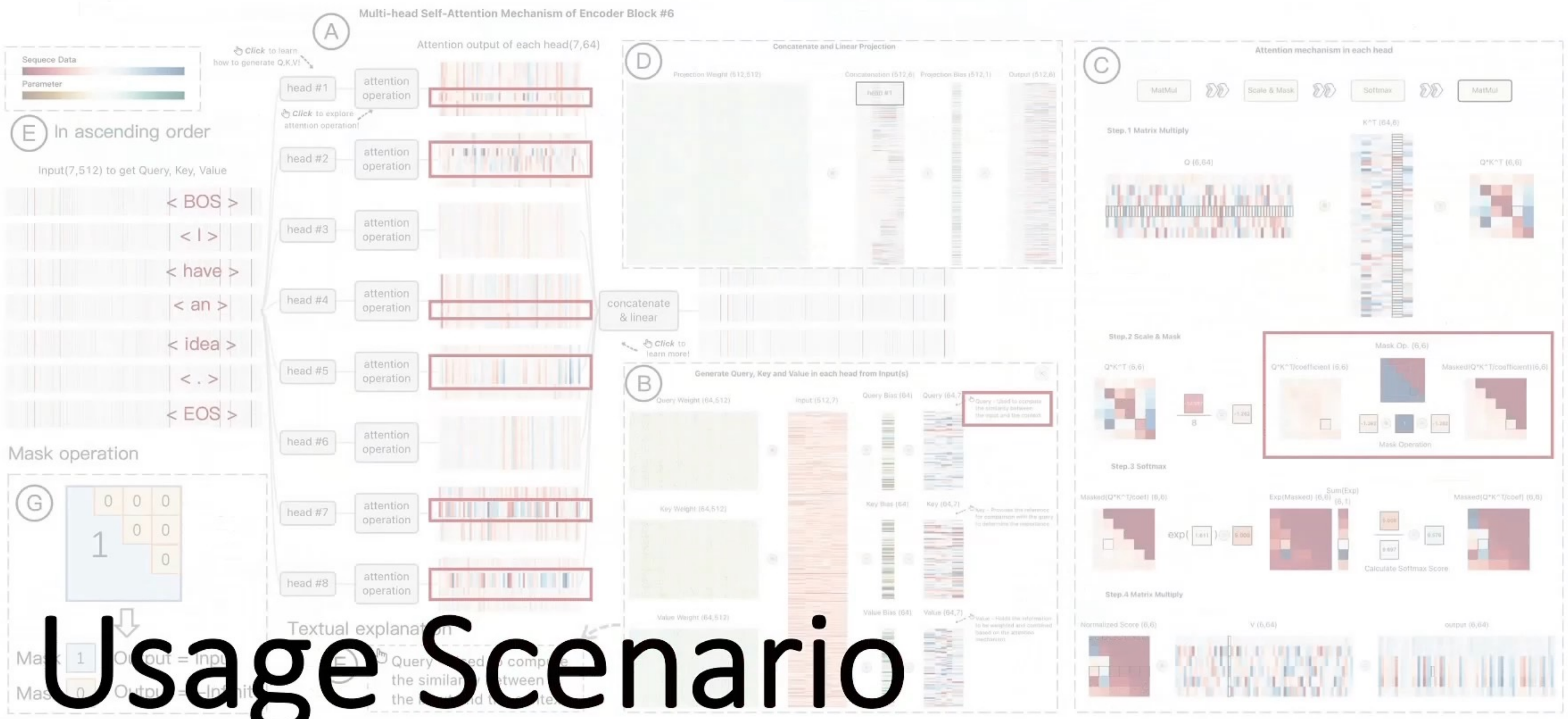
Teaching aid for lectures

- better summarize and present the teaching points
- increases the practicality and vividness of the entire teaching process

Usage Scenario



Usage Scenario



Usage Scenario

Self-study guidance for beginners

Evaluation

User-controlled Experiment

User Portrait →

System Introduction →

System Tutorial →

Observable Study →

Exit Questionnaire

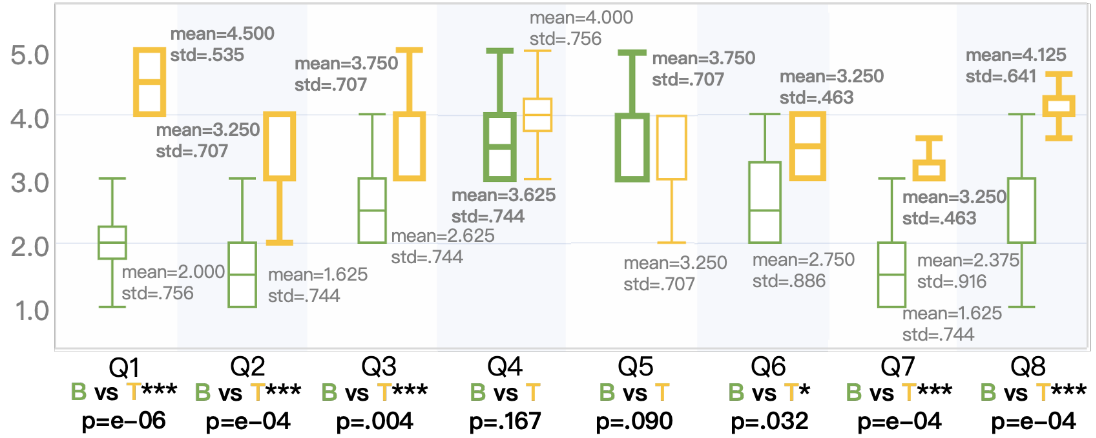
- R-1 visual summary
- R-2 interactive interface
- R-3 exploration mode
- R-4 self-directed & immersive

Level	Goal	Question
easy	G1	Q1: Components and data flow of feed-forward network.
easy	G3	Q2: Identify key words from attention matrix.
easy	G3	Q3: Final output in translation task and its derivation.
medium	G1	Q4: Differences between cross- and self-attention.
medium	G2	Q5: Add & LN significance and implementation.
medium	G1	Q6: Parallelism in Transformer.
hard	G2	Q7: Reasons for scaling before softmax.
hard	G2	Q8: Process of calculating PE & variation with position.

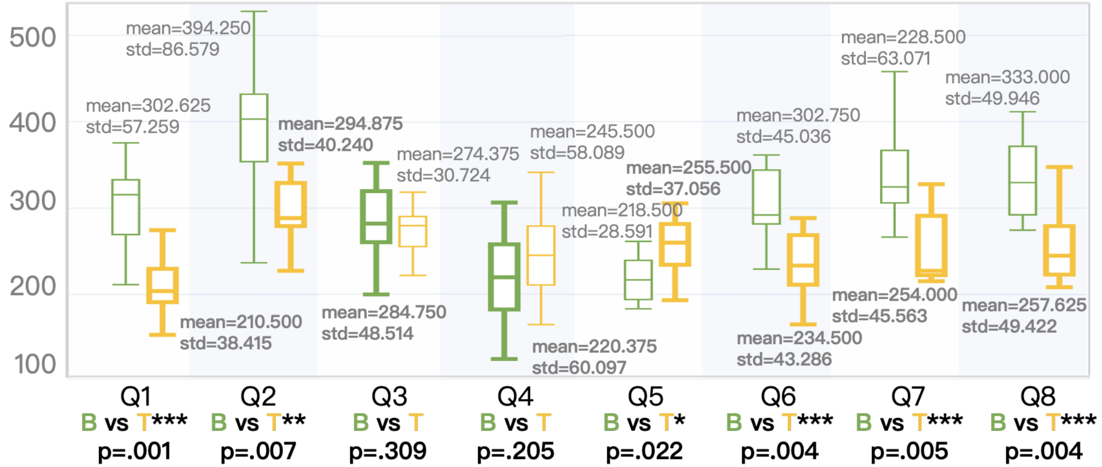
Evaluation

- improve users' understanding of structures and tasks
- bring more activity, autonomy and divergent thinking
- enhancing users' efficiency in learning through a broader coverage and enhanced interaction

Ⓐ Comparison of Questions' Answers Among Groups



Ⓑ Comparison of Answer Time Among Groups



Ⓒ Comparison of Learning Efficiency Index Among Groups

$E_{GroupX,i}$	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$	$i=8$	Mean	Std
$X = B$	0.737	1.005	0.680	0.839	0.981	0.824	0.965	0.851	0.851	0.121
$X = T$	1.421	1.702	1.631	1.402	1.263	1.385	1.381	1.542	**1.466**	0.146

Evaluation

User interviews

Implication

- Usability and effectiveness.
- Validating the knowledge for experts.

Limitation

- Different appropriate learning resources for different needs.
- Need for more instructions, animations, and comparisons.

VIS 2023

第 16 届中国 R 会议暨 2023 X-AGI 大会

TransforLearn

Interactive Visual Tutorial for the Transformer Model

Lin Gao[1], Zekai Shao[1], Ziqing Luo[1], Haibo Hu[2], Cagatay Turkay[3], Siming Chen[1,4]

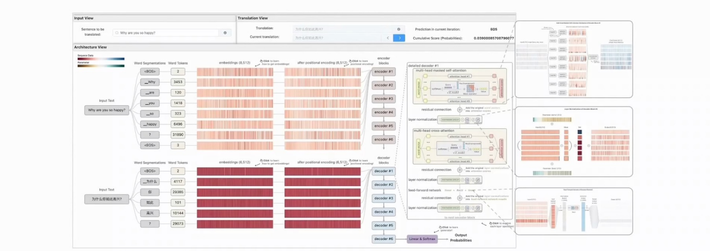
[1]School of Data Science, Fudan University

[2]School of Big Data & Software Engineering, Chongqing University

[3]Centre for Interdisciplinary Methodologies, University of Warwick

[4]Shanghai Key Laboratory of Data Science

Paper | Code | Demo



Thanks for your listening !

TransforLearn: https://trans-for-learn.github.io/

Welcome to our homepage: http://fduvis.net/

Email: leenagao0430@gmail.com