

From Shallow to Deep Language Representations

1 Basics 2 Shallow Models 3 Transformer 4 BERT

KDD19' Anchorage

Aston Zhang, Haibin Lin, Leonard Lausen, Sheng Zha, Alex Smola

www.d2l.ai gluon-nlp.mxnet.io

Context Matters: Retail Bank or River Bank?

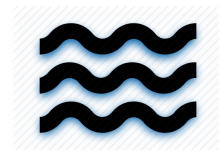
1. I jog along the **bank** of Duwamish River every day.
2. I went to the **bank** to open a savings account.

Context Matters: Retail Bank or River Bank?

1. I jog along the **bank** of Yangtze River every day.



2. I went to the **bank** to open a savings account.

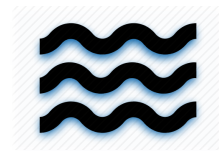


Context Matters: Retail Bank or River Bank?

1. I jog along the **bank** of Yangtze River every day.



2. I went to the **bank** to open a savings account.



With word embedding, the vector representing “**bank**” is the **same** in both sentences

Can we have representations
that depend on the **context**?

Representations

- Context-free representation
 - CBOW/Skip-gram
 - FastText
- Contextual representation
 - ELMo: Embedding from Language Model
 - **BERT: Bidirectional Embedding Representation from Transformers**

BERT

Bidirectional Embedding from
Transformers



General Language Understanding Evaluation (GLUE Benchmark)

Includes datasets for acceptability, sentiment, paraphrase, sentence similarity, natural language inference

Natural Language Inference Example:

Input_0: A man inspects the uniform of a figure in some East Asian country.

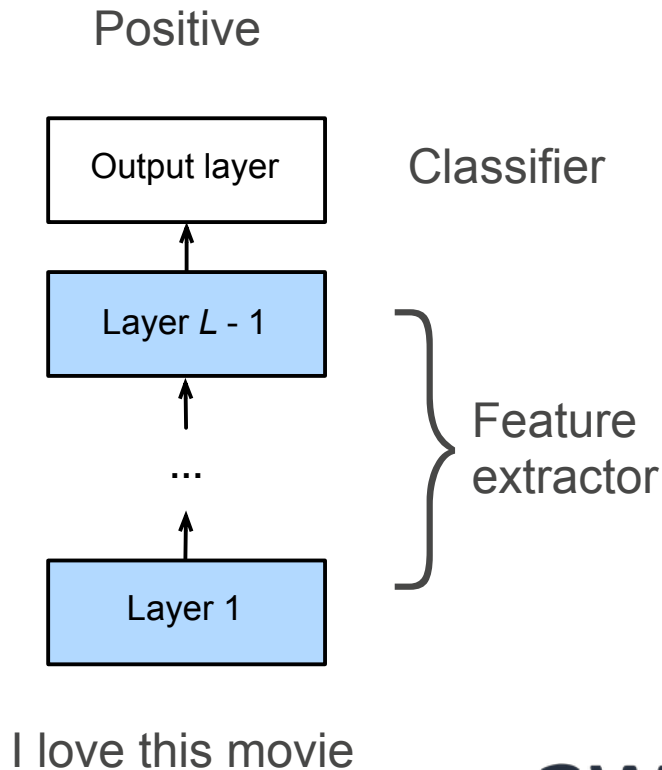
Input_1: The man is sleeping

Output: contradiction

Model	Avg Score
CBOW	58.6
BERT	80.5

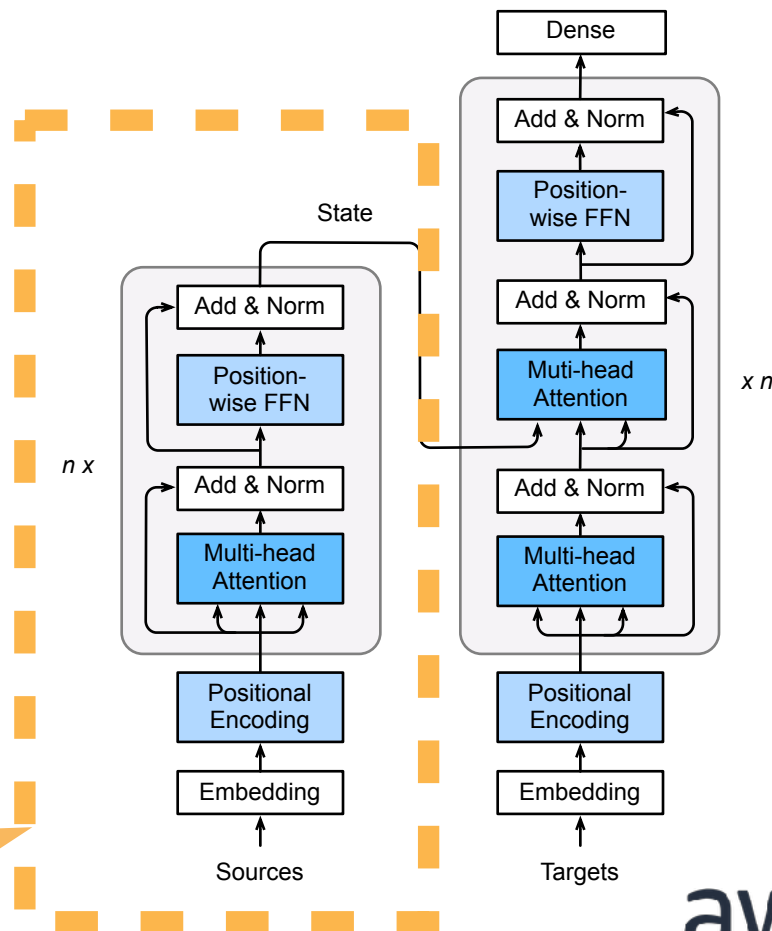
BERT

1. Pre-training: learn contextual representation on large scale corpus
2. Fine-tuning: add a simple output layer on BERT and fine-tune with the task at hand



BERT Architecture

- A (big) Transformer encoder
- BERT Base
 - # blocks = 12
 - # parameters = 110M
- BERT Large
 - # blocks = 24
 - # parameter = 340M



BERT

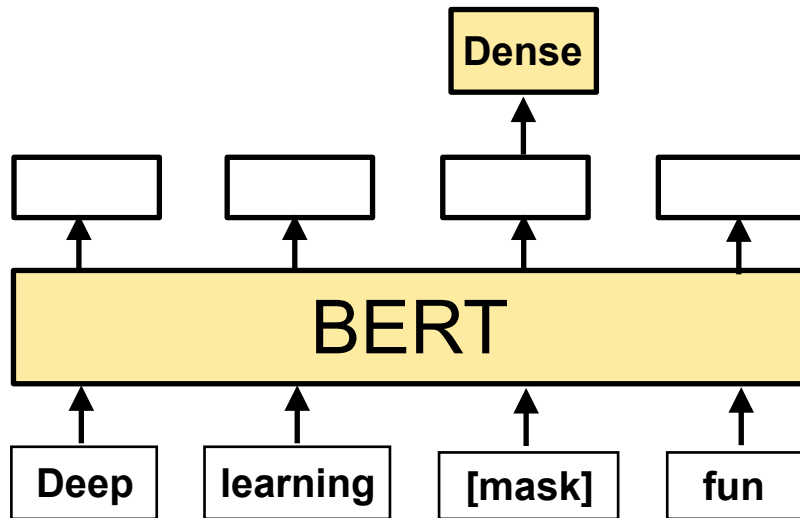
BERT Pre-training

- Pre-training tasks:
 - masked language modeling
 - next sentence prediction
- Dataset: Wikipedia and BooksCorpus (>3B words)

Pre-training Task 1: Masked Language Model

Original sentence:
Deep learning is fun.

Masked sentence:
Deep learning [mask] fun.



$$loss = -\log p(is \mid deep, learning, [mask], fun)$$

Pre-training Task 2: Next Sentence Prediction

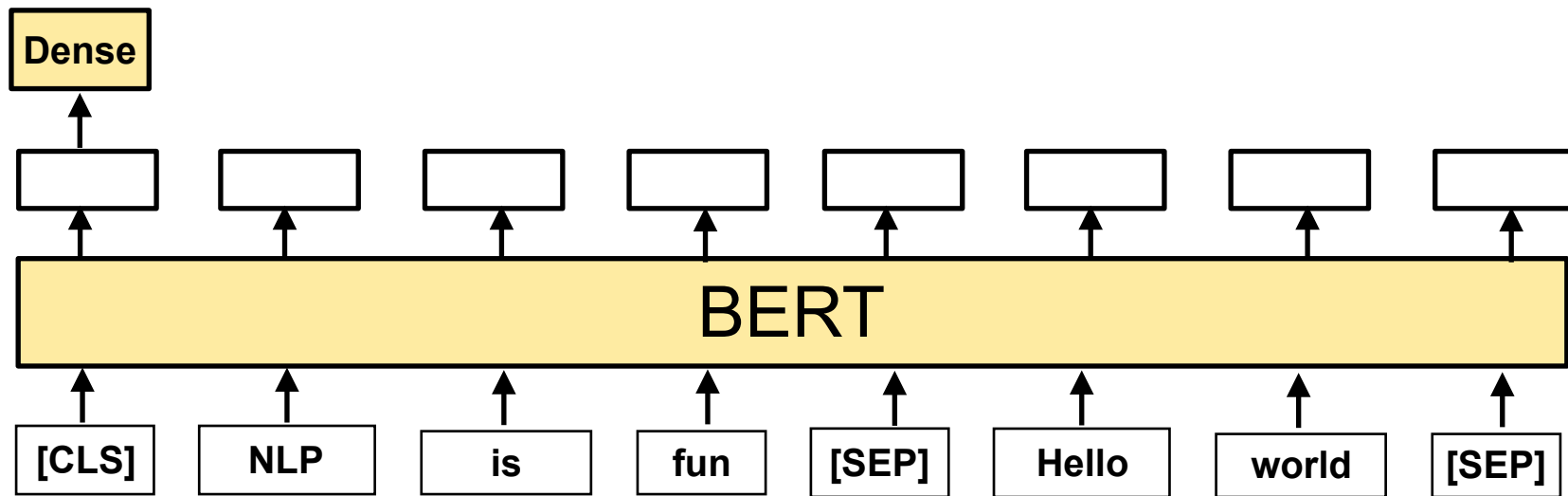
- Each example is a pair of sentences

is_next_sentence: NLP is fun. GluonNLP is awesome.

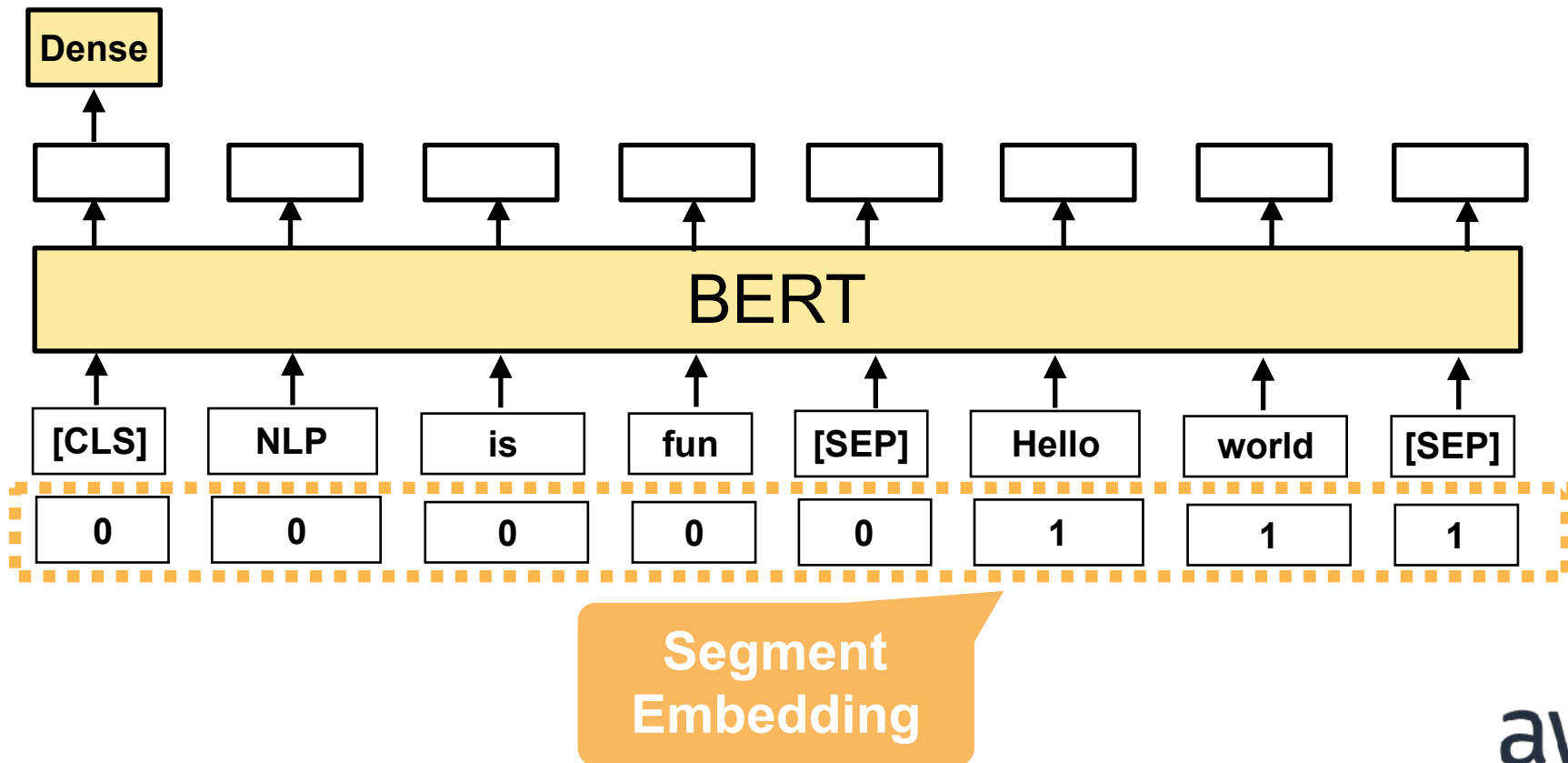
not_next_sentence: NLP is fun. Hello world.

- Sentence level binary classification

Pre-training Task 2: Next Sentence Prediction



Pre-training Task 2: Next Sentence Prediction

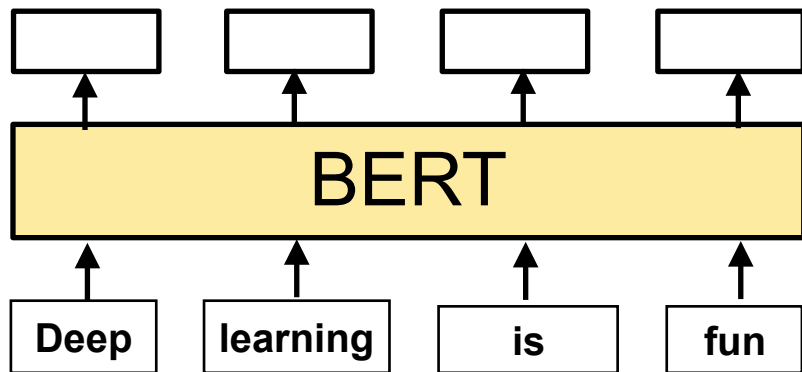


BERT Pre-training

- Pre-training tasks:
 - masked language modeling
 - next sentence prediction
- Dataset: Wikipedia and BooksCorpus (>3B words)

BERT Fine-tuning

- BERT returns a (contextual) feature vector for each token
- Different fine-tuning tasks use a different set of vectors



Fine-tuning: Sentence Classification

Input: This movie is great

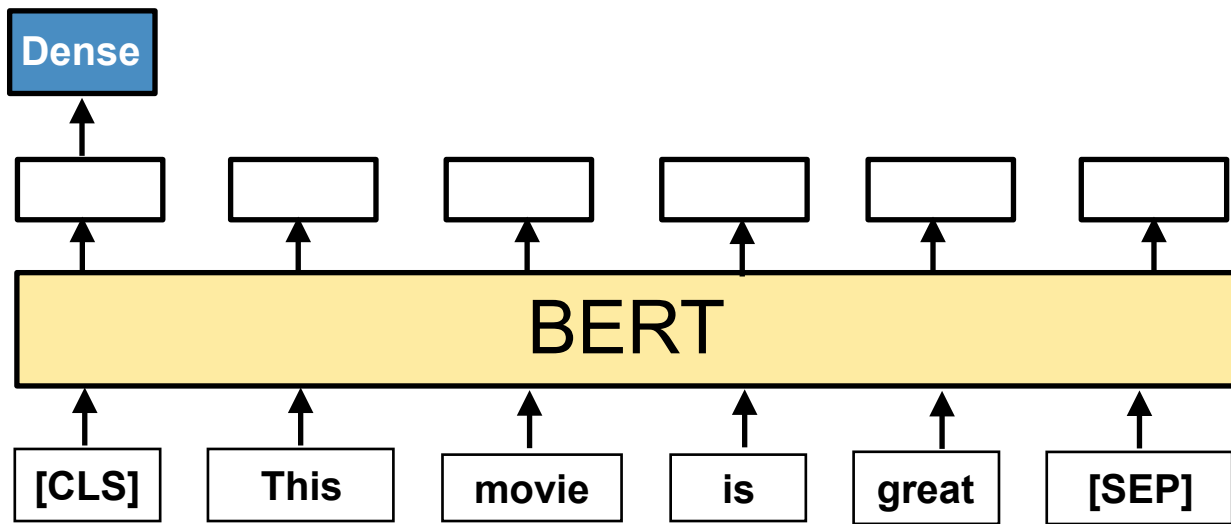
Output: positive

Fine-tuning: Sentence Classification

Input: This movie is

Output: positive

Feed the [CLS] token vector into a dense output layer.



Fine-tuning: Sentence Pair Classification

Input_0: The processor was announced in San Jose at the Forum.

Input_1: The processor was unveiled at the Forum in San Jose.

Output: is_paraphrase

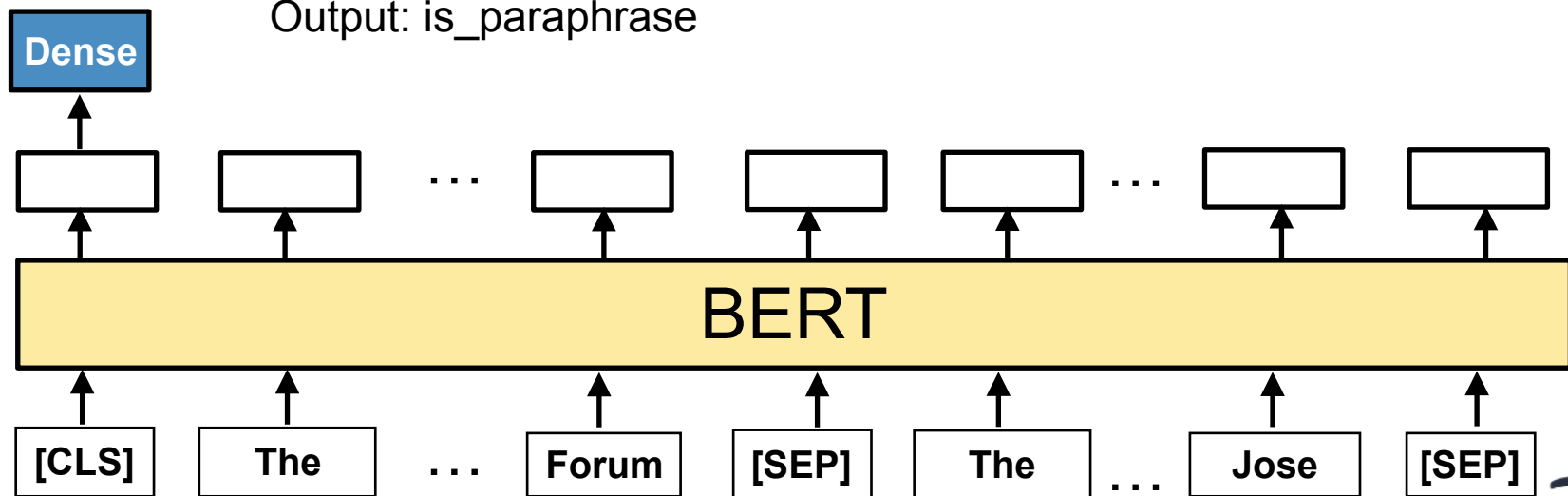
Fine-tuning: Sentence Pair Classification

- Feed the [CLS] token vector into a dense output layer.

Input_0: The processor was announced in San Jose at the Forum.

Input_1: The processor was unveiled at the Forum in San Jose.

Output: is_paraphrase



Fine-tuning: Named Entity Recognition

Input: Jim bought 3000 shares of Amazon in 2006.

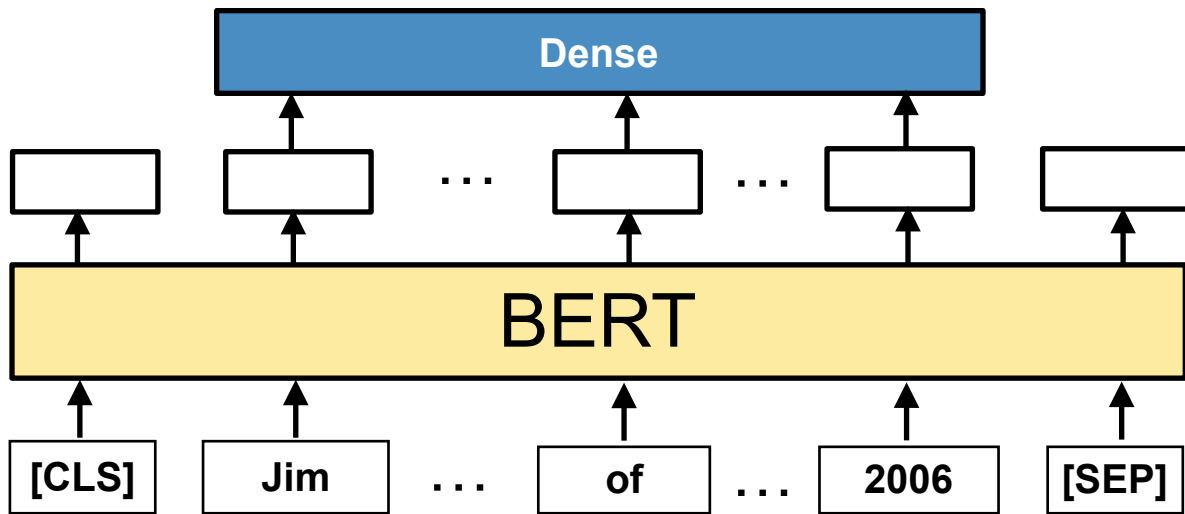
Output: [person] [organization] [time]

Fine-tuning: Named Entity Recognition

- Feed each non-special token vector into a dense output layer

Input: Jim bought 3000 shares of Amazon in 2006.

Output: [person] [organization] [time]



Fine-tuning: Question Answering

Given a question and a description text, find the answer, which is a text segment in the description

Input_0: KDD 2019 is held in Anchorage

Input_1: Where is KDD held

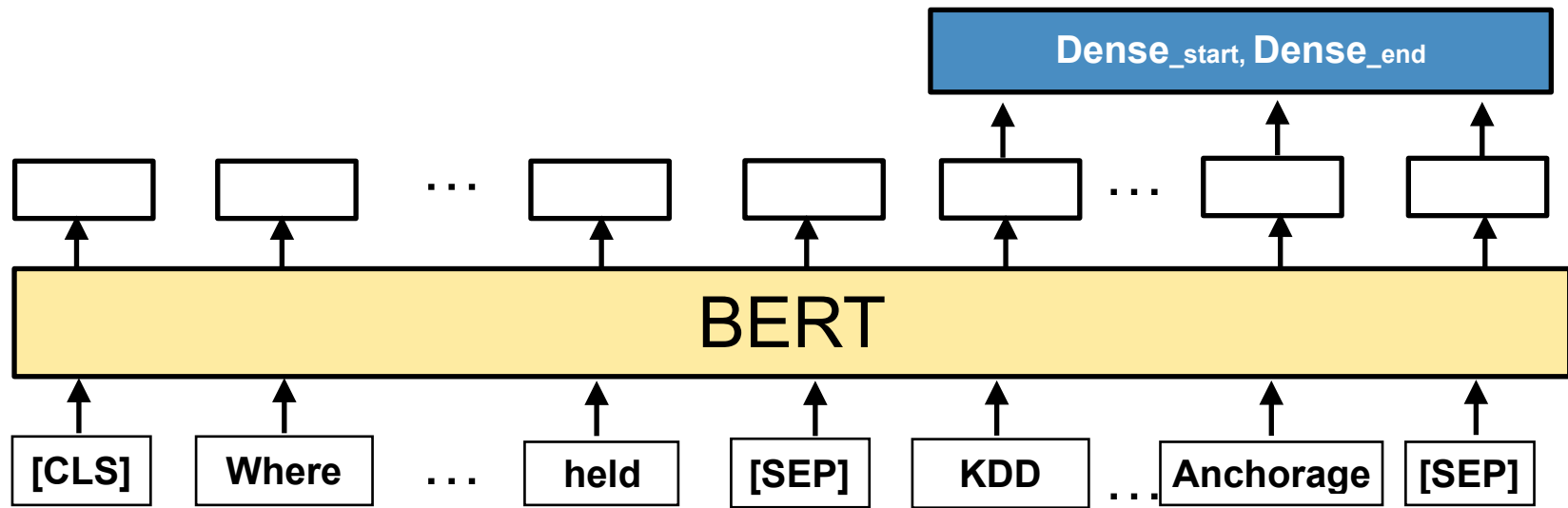
Output: Anchorage

Fine-tuning: Question Answering

Input_0: KDD 2019 is held in Anchorage

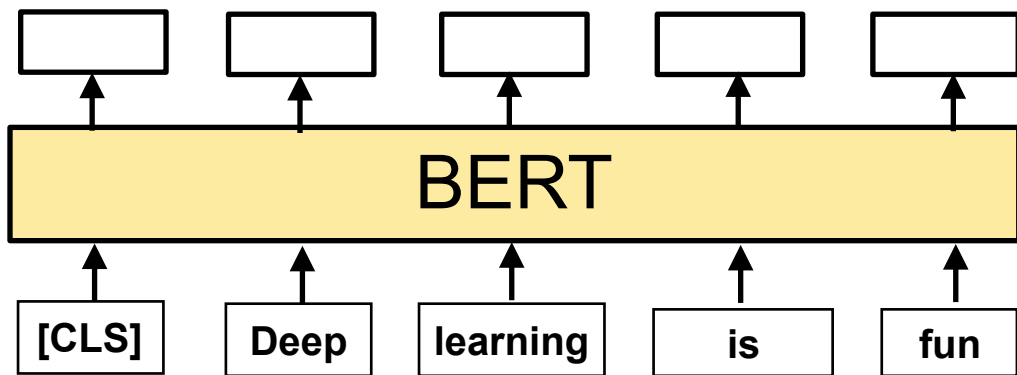
Input_1: Where is KDD held

Output: Anchorage



BERT Fine-tuning

- BERT returns a (contextual) feature vector for each token
- Different fine-tuning tasks use a different set of vectors



BERT in GluonNLP

```
from gluonnlp import model  
  
model.get_model(  
    "bert_12_768_12",  
    dataset_name="wiki_cn_cased"  
)
```

	bert_12_768_12	bert_24_1024_16
book_corpus_wiki_en_uncased	✓	✓
book_corpus_wiki_en_cased	✓	✓
openwebtext_book_corpus_wiki_en_uncased	✓	x
wiki_multilingual_uncased	✓	x
wiki_multilingual_cased	✓	x
wiki_cn_cased	✓	x
scibert_scivocab_uncased	✓	x
scibert_scivocab_cased	✓	x
scibert_basevocab_uncased	✓	x
scibert_basevocab_cased	✓	x
biobert_v1.0_pmc_cased		
biobert_v1.0_pubmed_cased		
biobert_v1.0_pubmed_pmc_cased		
biobert_v1.1_pubmed_cased		
clinicalbert_uncased	✓	x
ernie_baidu_cn_uncased	✓	x

**Available in
GluonNLP**

Notebook: BERT for Sentiment Analysis

07_bert_app/bert.ipynb