МИСИС
УНИВЕРСИТЕТ

Трансформер. Часть 2

Корнет Мария Евгеньевна
старший преподаватель кафедры
Инженерная кибернетика

BERT

BERT (Bidirectional Encoder Representations from Transformers)

Ключевая идея:

Использовать **только энкодерную часть** Transformer для получения глубоких двунаправленных контекстуальных представлений текста

Архитектурные особенности:

- **Только энкодер.**
- **Маскирование (Masked Language Model, MLM).** На этапе предобучения 15% токенов маскируются случайным образом, и модель учится их предсказывать, исходя из контекста **со всех сторон**.
- **Задача предсказания следующего предложения (NSP).**

https://arxiv.org/abs/1810.04805

BERT

- **Data:** Wikipedia (2.5B words) + BookCorpus (800M words)
- **Batch Size:** 131,072 words (1024 sequences * 128 length or 256 sequences * 512 length)
- **Training Time:** 1M steps (~40 epochs)
- **Optimizer:** AdamW, 1e-4 learning rate, linear decay (после 10000 шагов к исходному состоянию)

- dropout = 0.1 на всех слоях
- BERT-Base: 12-layer, 768-hidden, 12-head (базовая архитектура)
- BERT-Large: 24-layer, 1024-hidden, 16-head (большая архитектура)
- Trained on 4x4 or 8x8 TPU slice for 4 days
- WordPiece-токенизация (30 000 токенов)

Transformer

- **Data:** WMT 2014 English-German (4.5M пар предложений) + WMT 2014 English-French (36M пар предложений)
- **Batch Size:** 25,000 tokens ≈ 64-128 sequences × ~200-400 length
- **Training Time:**
  - English-German: 100,000 steps (~12 часов)
  - English-French: 300,000 steps (~3.5 дня)
- **Optimizer:** Adam, $\beta_1=0.9$, $\beta_2=0.98$, $\varepsilon=10^{-9}$
- **Warmup Steps:** 4,000 (линейный рост LR, затем обратный корень decay)
- **Dropout:**
  - Residual dropout = 0.1
  - Attention dropout = 0.1
  - Embedding dropout = 0.1 (только в большей модели)
- **Label Smoothing:** ε_ls = 0.1

BERT представление входа

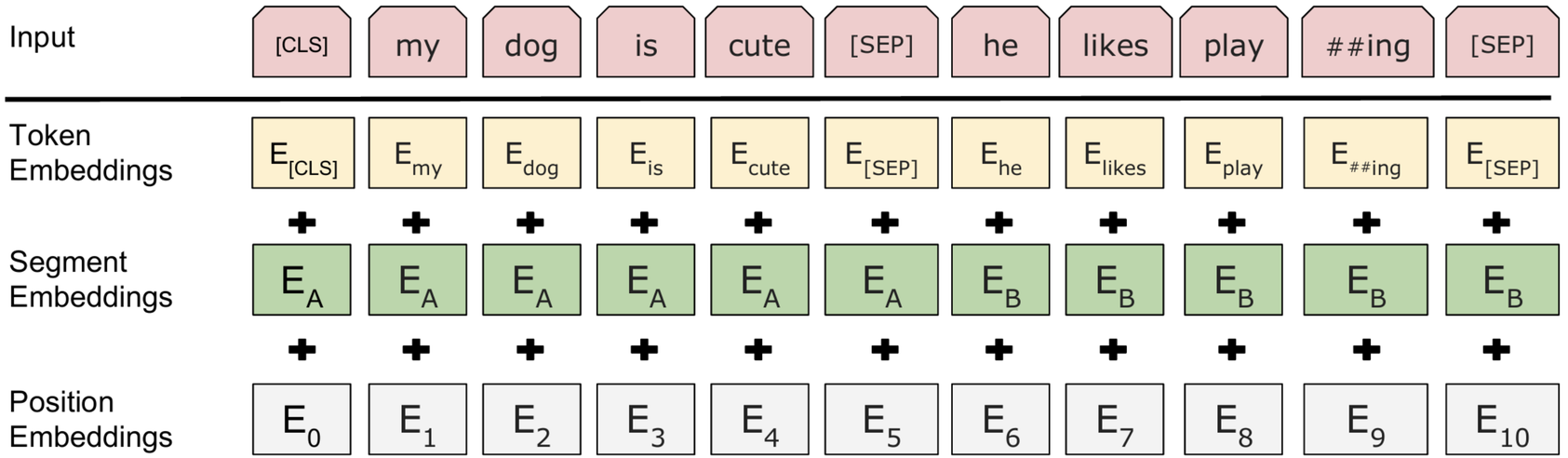


Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

https://arxiv.org/abs/1810.04805

BERT

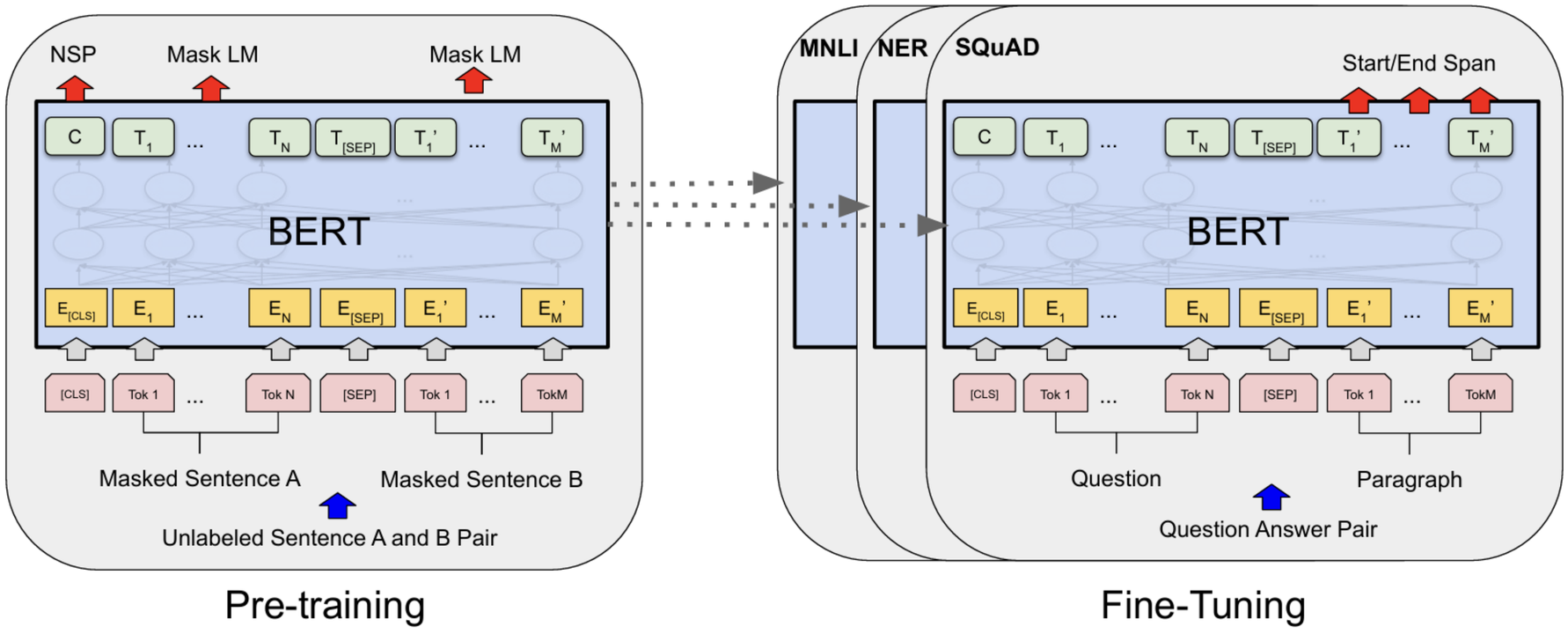
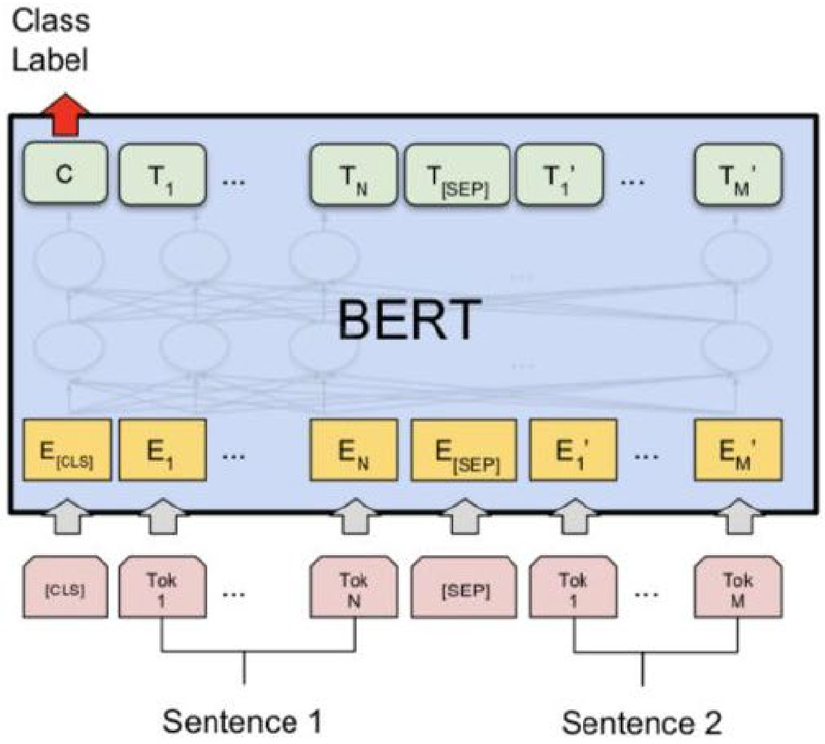


Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. `[CLS]` is a special symbol added in front of every input example, and `[SEP]` is a special separator token (e.g. separating questions/answers).

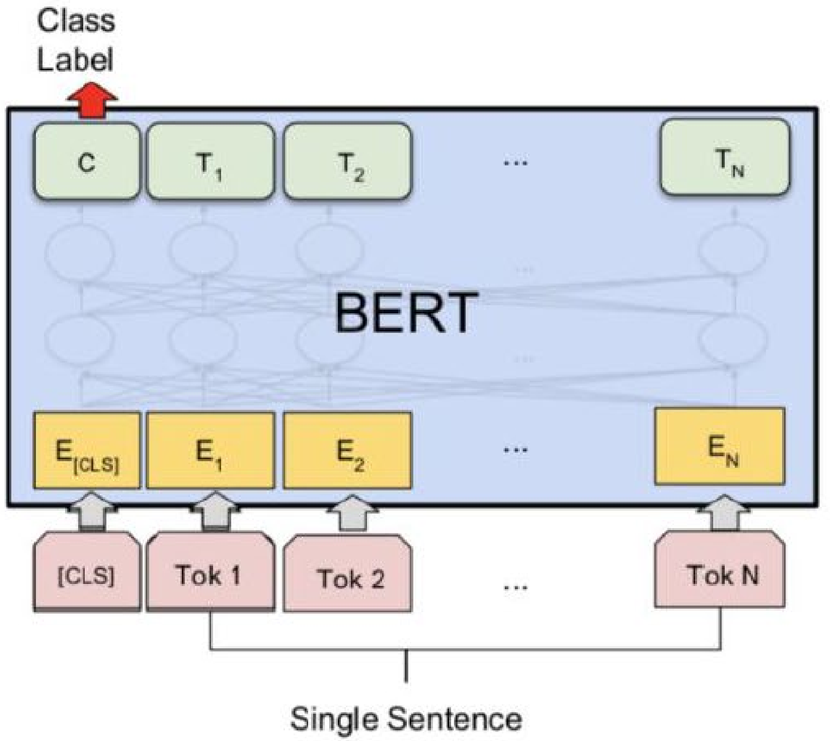
Задачи

Название задачи		Тип задачи	
MNLI	Multi-Genre Natural Language Inference	Natural Language Inference (NLI)	Отношение между посылкой и гипотезой
QQP	Quora Question Pairs	Дублирование вопросов	Являются ли два вопроса дубликатами
QNLI	Question NLI	NLI на вопрос+текст	Содержит ли текст ответ на вопрос
STS-B	Semantic Textual Similarity Benchmark	Регрессия	Оценить степень семантической близости (0–5)
MRPC	Microsoft Research Paraphrase Corpus	Классификация перефраз	Перефраз ли одно предложение другого
RTE	Recognizing Textual Entailment	Natural Language Inference (NLI)	Следует ли гипотеза из посылки
SWAG	Situations With Adversarial Generations	Завершение ситуации (multiple choice)	Выбрать логичное продолжение из 4-х вариантов

BERT

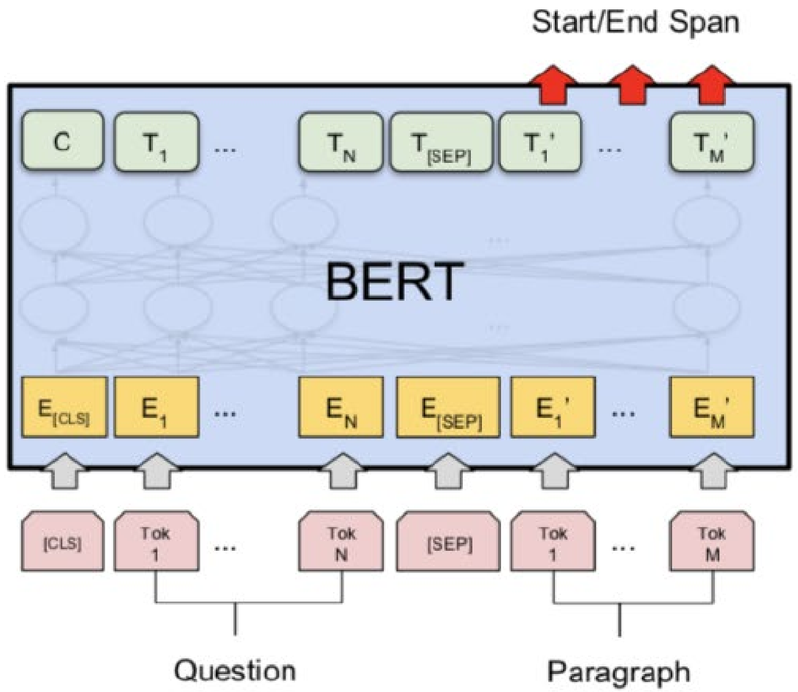


(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

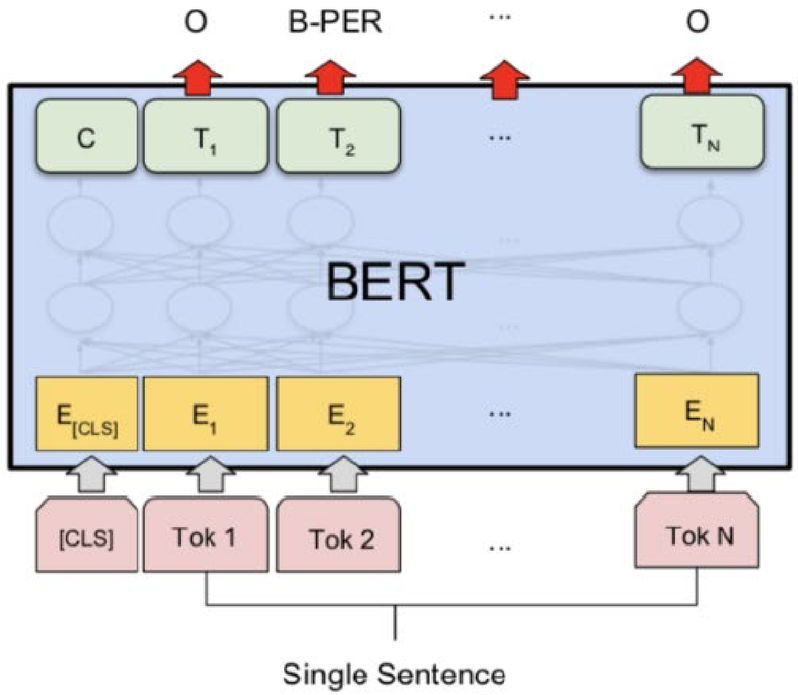


(b) Single Sentence Classification Tasks: SST-2, CoLA

BERT



(c) Question Answering Tasks:
SQuAD v1.1



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

МИСИС
УНИВЕРСИТЕТ

Спасибо
за внимание!