Introduction to Search Relevance Ranking- Session III – Knowedge Distillation

Tutorial Link: https://dlranking.github.io/dlrr/

Data source: https://huggingface.co/datasets/xglue

XGLUE: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation)

Presenters: Xue Li, Keng-hao Chang

Notebooks: Xue Li

Date: August 14th, 2022

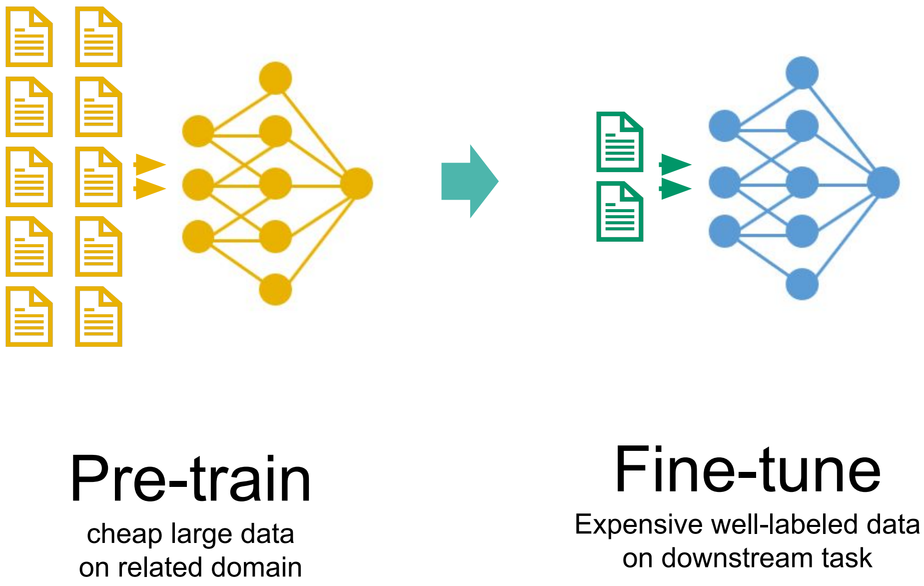
Agenda

- Knowledge distillation
- Case study in DistilBert
- Case study in Microsoft Ads for Ranking
- The colab

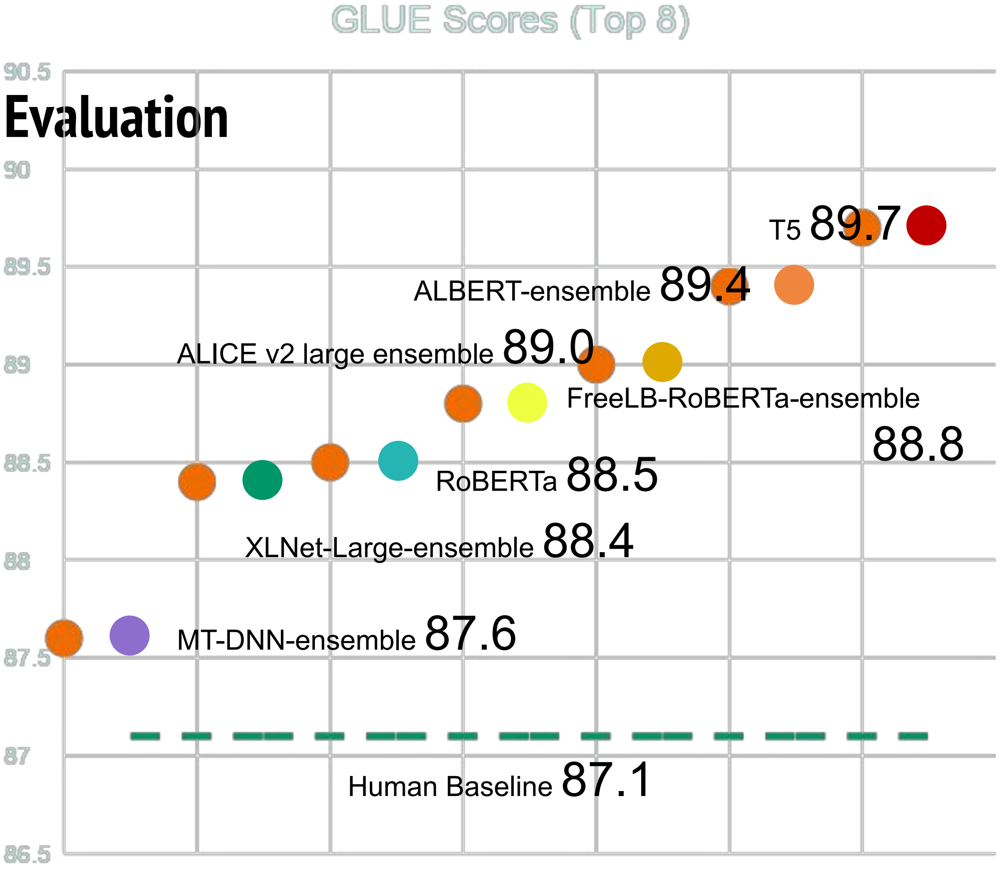
Pre-train and fine-tune

- Natural Language Processing
- Pre-train on unsupervised tasks, e.g. Language Modeling
- Fine-tune on downstream NLP tasks, e.g. Question Answering

- Large & powerful NLP models, even beat human!

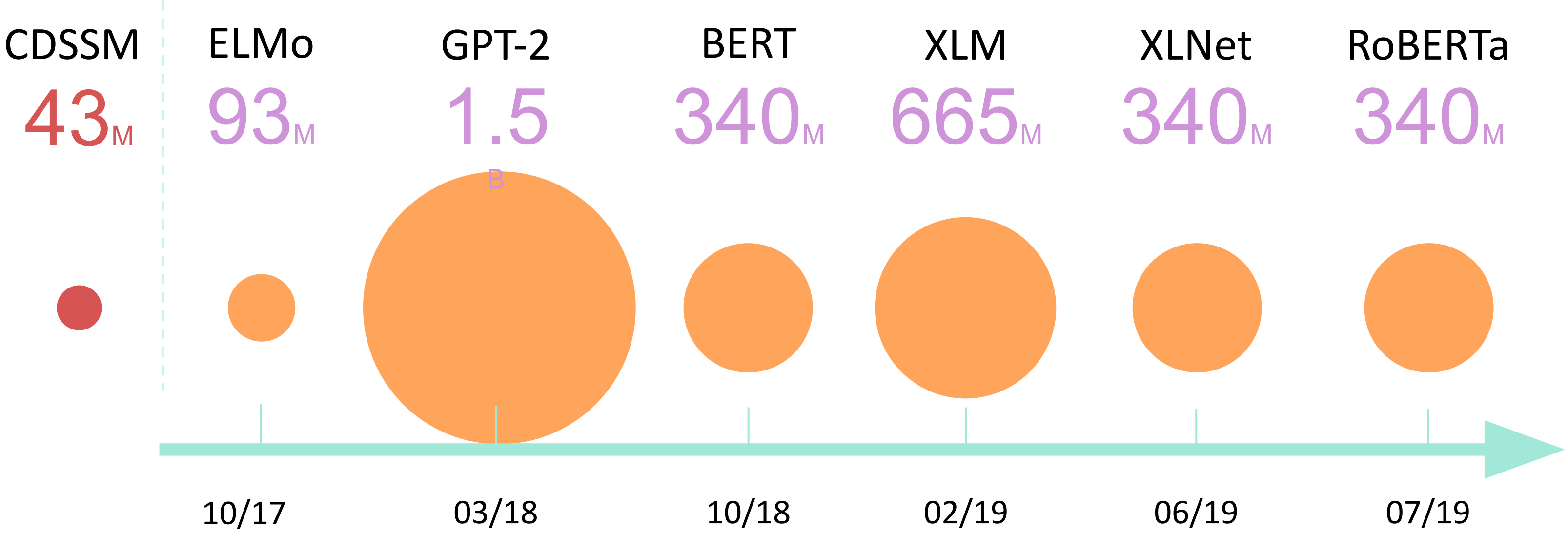


GLUE –
General Language Understanding Evaluation



https://gluebenchmark.com/leaderboard/, 2019-11-12

How large are pre-trained NLP models?



Call outs

Will cover

- Knowledge distillation
- Practices of knowledge distillation

Will not cover

- Other smaller model techniques (Albert, Electra)

Response level knowledge distillation

- Vs. Logits
- Soft label, dark knowledge, Hinton
- Analogous to label smoothing
- Limited to supervision learning

$$L_{ResD}(z_t, z_s) = \mathcal{L}_R(z_t, z_s) \ , \tag{1}$$

$$p(z_i, T) = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)} \ , \tag{2}$$

$$L_{ResD}(p(z_t, T), p(z_s, T)) = \mathcal{L}_R(p(z_t, T), p(z_s, T)) \ . \tag{3}$$

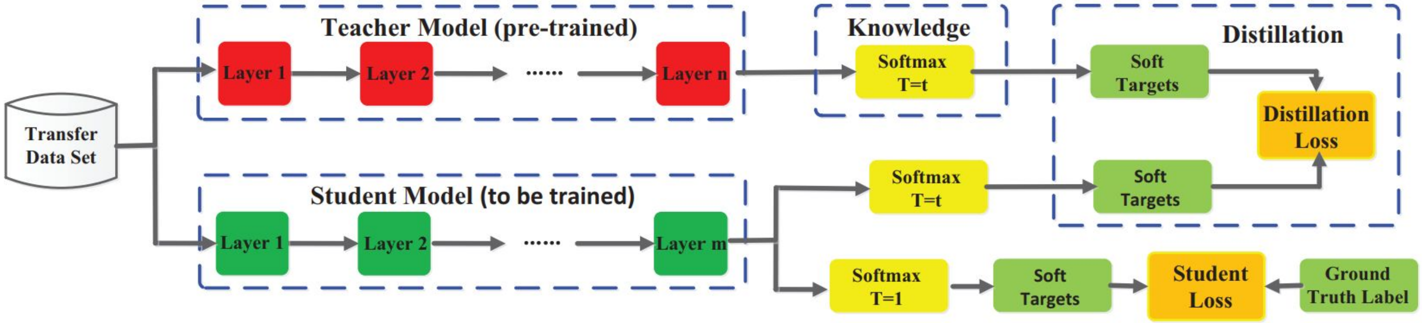
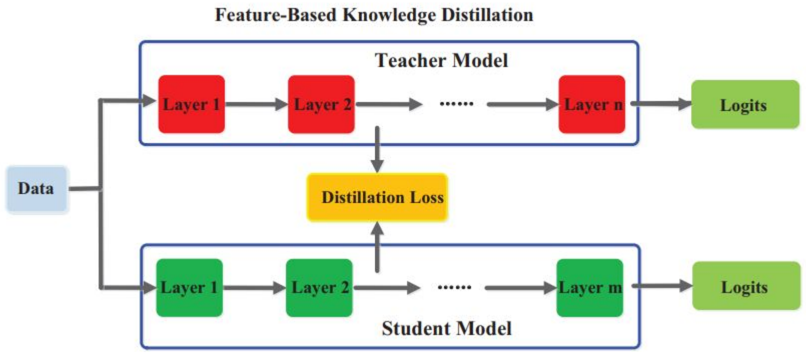


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

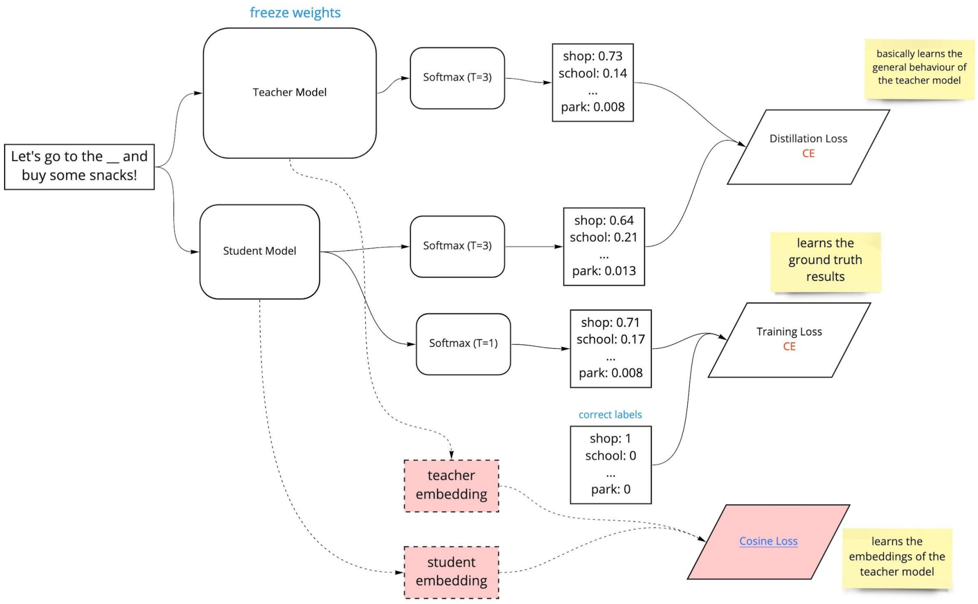
Feature-based knowledge distillation

- L2-norm distance, L1-norm distance, cross entropy loss, cosine loss

$$L_{FeaD}\left(f_t(x), f_s(x)\right) = \mathcal{L}_F\left(\Phi_t(f_t(x)), \Phi_s(f_s(x))\right) , \quad (4)$$



Case study: distillBert



- 3 losses
  - Distillation loss
  - Training loss
  - Cosine loss

- DistillBERT model retains almost 97% of the original BERT-base model's language undersetanding when evaluated on GLUE benchmarks. In addition to this, it is 40% smaller and 60% faster at inference.

Case study: Knowledge Distillation from Internal Representations

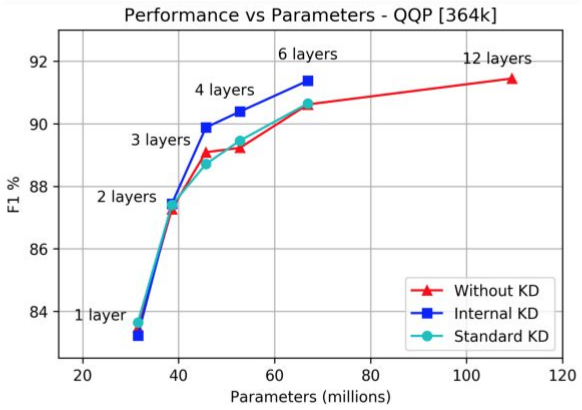


Figure 2: Performance vs. parameters trade-off. The points along the lines denote the number of layers used in BERT, which is reflected by the number of parameters in the x-axis.

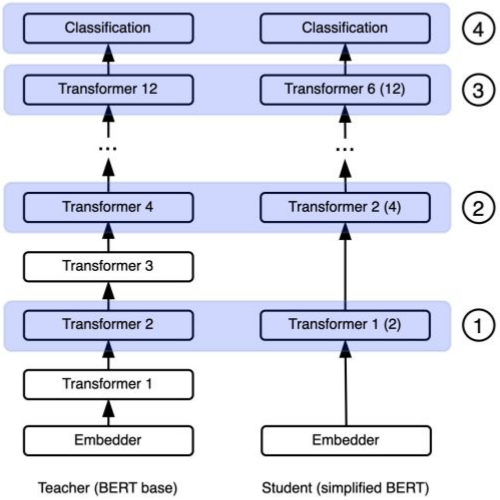


Figure 1: Knowledge distillation from internal representations. We show the internal layers that the teacher (left) distills into the student (right).

- Offline distillation
- Online distillation
    - Both teacher and student are updated simutaneously
- Self distillation

- Residual learning

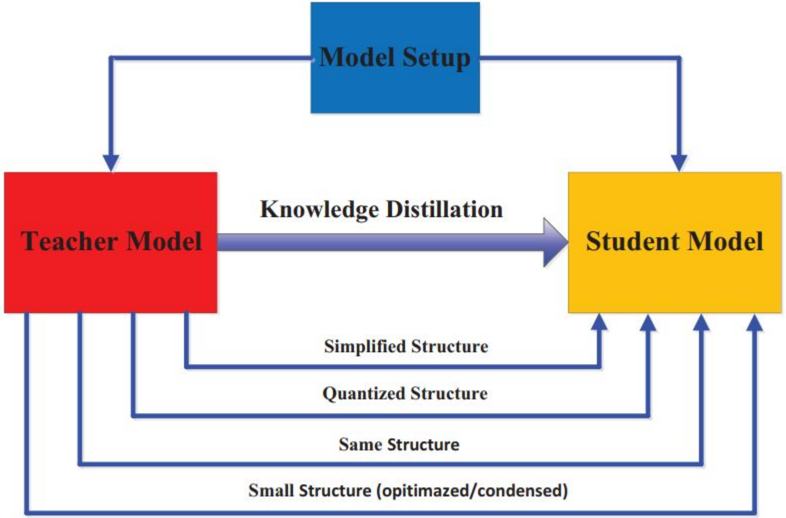


Fig. 9 Relationship of the teacher and student models.

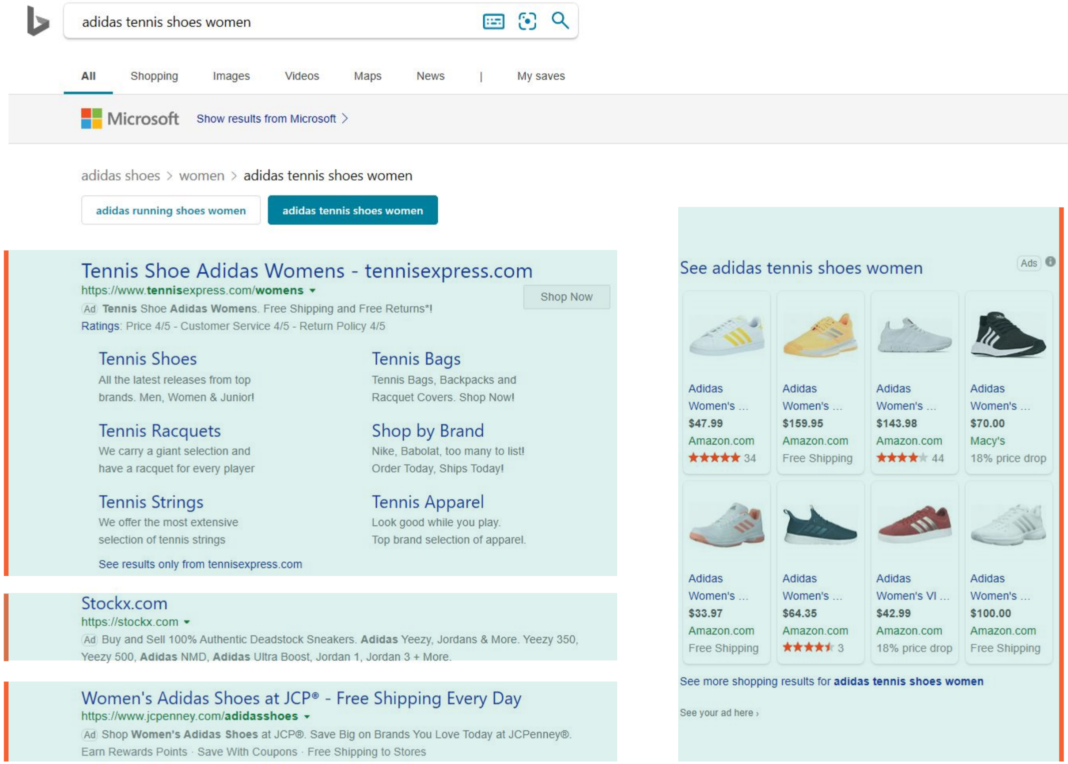
Case study: search relevance ranking at Microsoft Ads

- Point-wise relevance score
- Used as externality in ranking
  - pDefect = 1-Relevance
  - Bid * pClick – w*pDefect

Text Ads #1

Text Ads #2

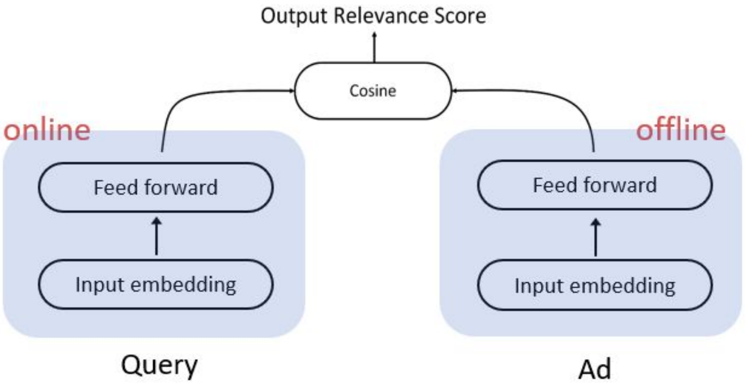
Text Ads #3

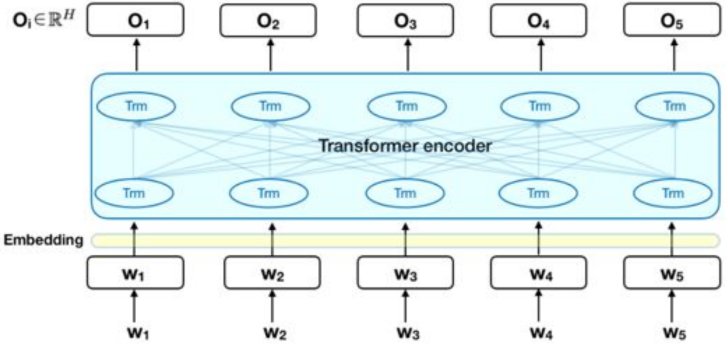


Choice of student model

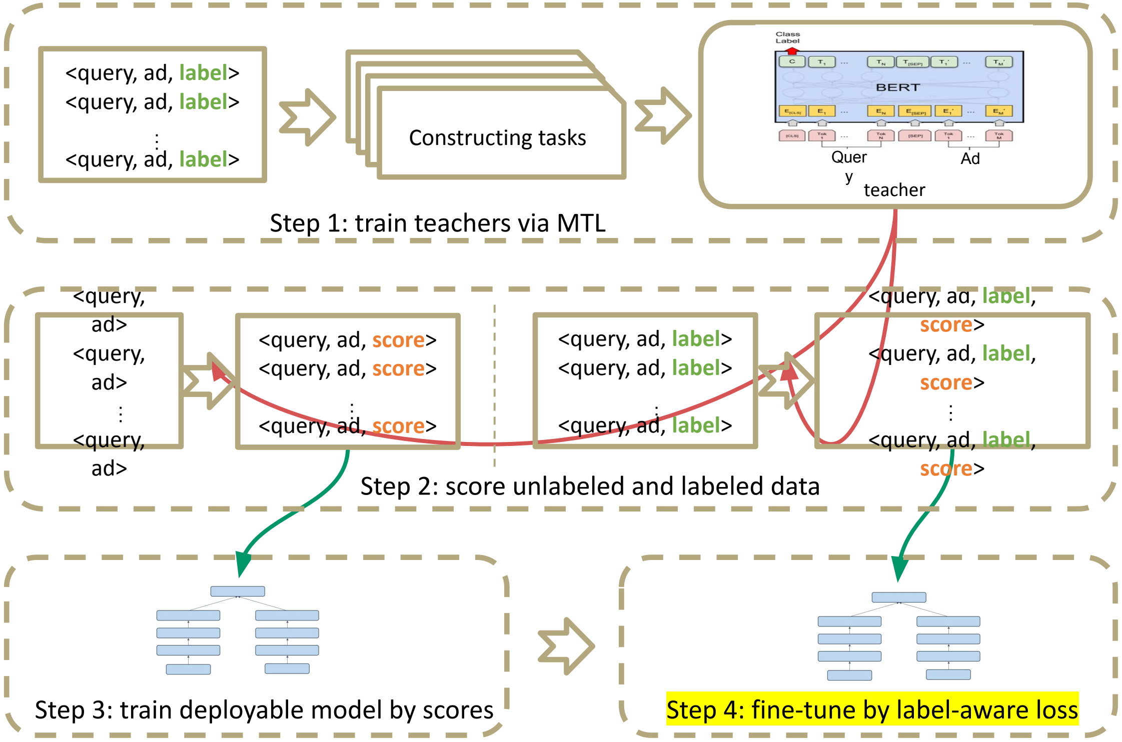
- Two tower
  - CDSSM, TwinBert
  - Doc embedding is offline computed

- BERT-like
  - cannot support fast compute, latency prohibitive.





Knowledge distillation for search Relevance



Teacher Training

- Narrow the gap between pre-trained models and target tasks

Inference Data

- Score both labeled / unlabeled data

- Train a deployable student model
- Using scored unlabeled data

Student Fine-tuning

- Fine-tune student model
- Using scored labeled data

Learning Fast Matching Models from Weak Annotations, WWW'19

Recipe of AdsBERT Distillation

**Initializatio
n**

Pre-trained BERT
340M params

Pretrain

MLM/NSP
400M Ads data

**MTL
Finetune**

8 ad tasks
40M samples

Inference

Vast amount
Proper distribution

Distillation

CDSSM keep **70%**
AUC gain

Case study: TwinBert

Table 2: ROC-AUC of TwinBERT models comparing with C-DSSM, $BERT_3$ and $BERT_{12}$ on two test sets

Model	AUC_1	AUC_2
C-DSSM	0.8713	0.8571
$BERT_3$	0.8995	0.9107
$TwinBERT_{cos}$	0.8883	0.8743
$TwinBERT_{res}$	0.9010	0.9113
$BERT_{12}$	0.9011	0.9137

Table 3: Density differences of all 4 labels by comparing top 5 results from $TwinBERT_{cos}$ and C-DSSM

bad	fair	good	excellent
-7.4%	-2.6%	1.9%	18.8%

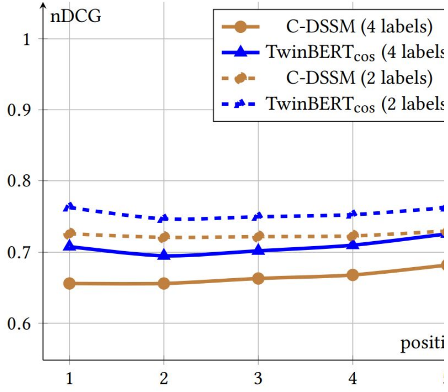


Figure 3: nDCGs of $TwinBERT_{cos}$ and C-DSSM

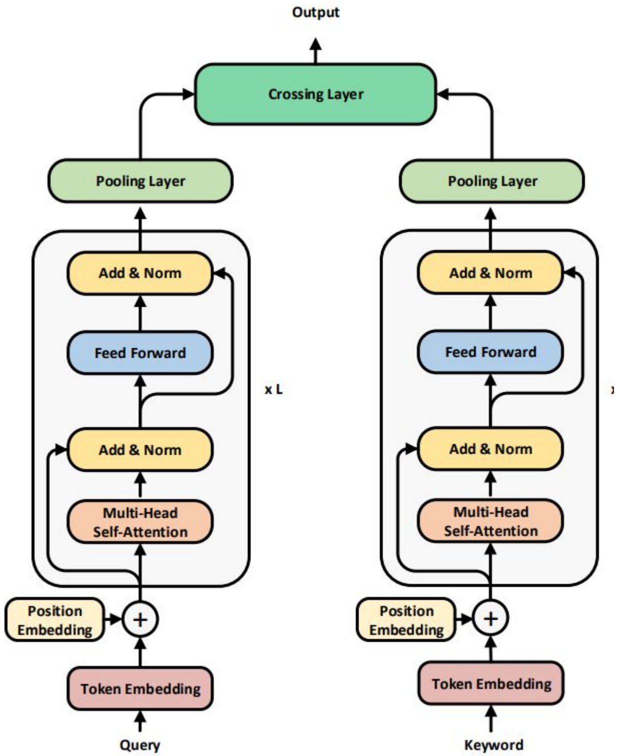


Figure 1: TwinBERT Architecture

Colab

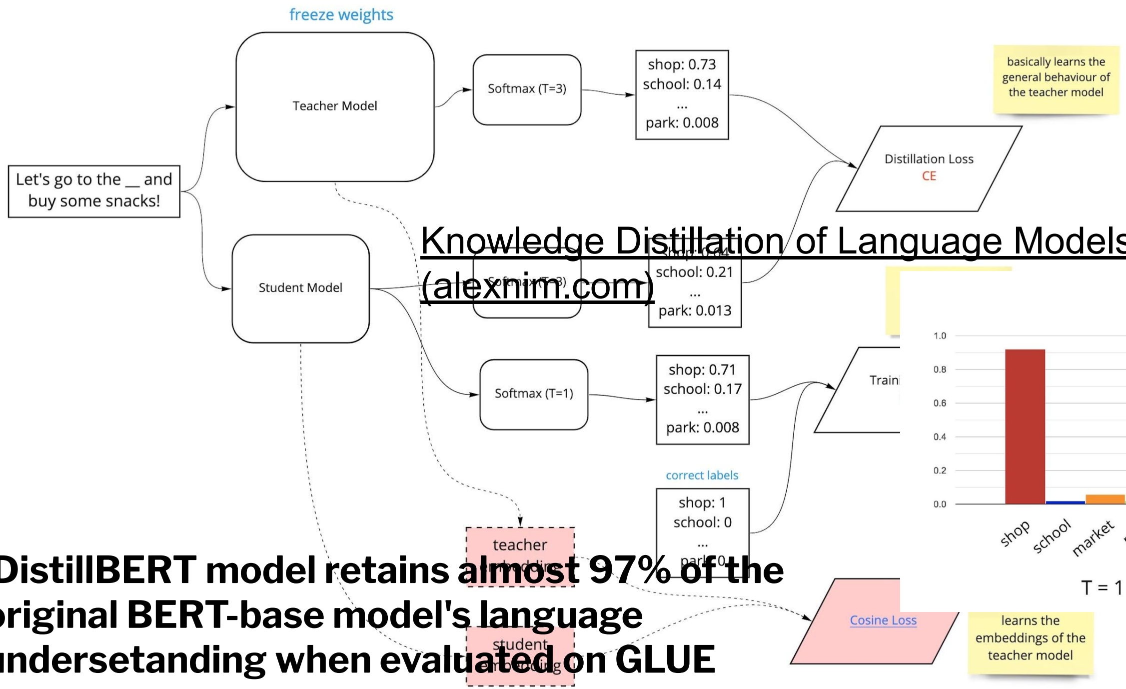
- QADSM task in xGLUE dataset, which is extracted from real Bing Ads traffic.

Appendix

- After seeing a lot of examples, the network can act as a maximum likelihood estimator, but it needs to be exposed to many examples before it can assign good probabilities, as the labels it sees are samples from a distribution with extremely high variance.

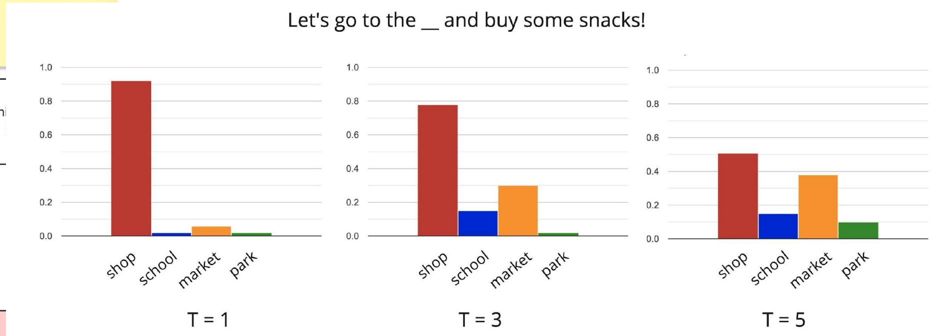
- If you've done transfer learning before, you can see where this is going.
  The big idea is:
  We can take a large pre-trained model like BERT, called the **teacher**, fine-tune it on the target task if it differs from the pre-training task, use it to predict the probabilities for our data, then use the probabilities as "soft labels" for the target model, the **student**. This way we can communicate the target distribution to the network with fewer examples!
  This also corresponds to training a student to reproduce the behavior of the teacher as accurately as possible, but with fewer parameters.

Case study: DistilBert



[7.jpg (2503×1597) (alexnim.com)]()

[Knowledge Distillation of Language Models (alexnim.com)]()



DistillBERT model retains almost 97% of the original BERT-base model's language undersetanding when evaluated on GLUE benchmarks. In addition to this, it is 40% smaller and 60% faster at inference.

Relation-based knowledge distillation

Case study: Acoustic Modeling by Amazon Alexa

- Parthasarathi and Strom (2019) leveraged student-teacher training to generate soft targets for 1 million hours of unlabeled speech data where the training dataset consisted only of 7000 hours of labeled speech.
- The teacher model produced a probability distribution over all the output classes. The student model also produced a probability distribution over the output classes given the same feature vector and the objective function optimized the cross-entropy loss between these two distributions. Here, knowledge distillation helped simplify the generation of target labels on a large corpus of speech data.

Separability enables fast retrieval via ANN