Generalization

(or lack thereof)

Changyan Wang and Ben Dodge

Learning and Evaluating
General Linguistic Intelligence

Dani Yogatama, Cyprien de Masson d'Autume, Jerome Connor,
Tomas Kocisky, Mike Chrzanowski, Lingpeng Kong,
Angeliki Lazaridou, Wang Ling, Lei Yu, Chris Dyer, Phil Blunsom
DeepMind, London, United Kingdom
{dyogatama,cyprien,jeromeconnor,tkocisky,chrzanowskim}@google.com
{lingpenk,angeliki,lingwang,leiyu,cdyer,pblunsom}@google.com

Scientific groundwork

BAM! Born-Again Multi-Task Networks for Natural Language Understanding

Kevin Clark[†] **Minh-Thang Luong**[‡] **Urvashi Khandelwal**[†]
Christopher D. Manning[†] **Quoc V. Le**[‡]
[†]Computer Science Department, Stanford University
[‡]Google Brain
{kevclark,urvashik,manning}@cs.stanford.edu
{thangluong,qvl}@google.com

Better techniques

What is general linguistic intelligence?

Criteria from the paper:

(i) deal with the full complexity of natural language across a variety of tasks

(ii) effectively store and reuse representations, combinatorial modules, and previously acquired linguistic knowledge to avoid *catastrophic forgetting*

(iii) adapt to new linguistic tasks in new environments with little experience

And why do we **care**?

Where are we now?

Codelength evaluation

Datasets/metrics do not focus on *generalization* or *abstraction*

Models are not evaluated on all datasets from a given task

MRQA 2019

BERT

Unsupervised pretraining enables transfer to many tasks

Multi-task training provides a pathway to general intelligence

BAM!
GLUE

Paper Outline

- ➢ New evaluation metric
- ➢ Tasks & datasets
- ➢ Models
- ➢ Five interesting questions

New evaluation metric

Codelength aims to measure the number of task-specific training examples needed to reach high performance

$$\ell(\mathcal{A}) = \log_2 |\mathcal{Y}| - \sum_{i=2}^{N} \log_2 p(y_i \mid x_i; \hat{\mathbf{W}}_{\mathcal{A}_{i-1}})$$

Dataset

Number of classes

Parameters trained on examples 1 through *i*–1

Interpretation (Blier & Ollivier, 2018)

Alice has all (x, y) pairs and Bob only has the x. Alice wants to send y to Bob.

Proposition 1 (Shannon–Huffman code)**.** *Suppose that Alice and Bob have agreed in advance on a model* p*, and both know the inputs* $x_{1:n}$*. Then there exists a code to transmit the labels* $y_{1:n}$ *losslessly with codelength (up to at most one bit on the whole sequence)*

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i) \tag{2.1}$$

We are not concerned with how to do this in practice.

Uniform Encoding

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i)$$

Don't use any deep nets at all, use a uniform model (*K* is number of classes)

$$p(y_i|x_i) = \frac{1}{K} \longrightarrow \ell(A) = N \log_2 K$$

This is the same as if Alice just sent all the labels to Bob with no model.

Two-Part Encoding

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i)$$

1. Alice trains a deep net and sends the parameters θ to Bob
2. Alice uses the deep net to transmit the labels more efficiently

$$\ell(A) = \ell(\theta) - \sum_{i=1}^{N} \log_2 p_\theta(y_i|x_i)$$

Bits needed to
transmit parameters
(too large)

Online/Prequential Code

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i)$$

1. Alice sends one label
2. Both Alice and Bob train on the label
3. Alice uses resulting deep net to send the next label

$$\ell(A) = \log_2 K - \sum_{i=2}^{N} \log_2 p_{\theta_{A_{i-1}}}(y_i|x_i)$$

Bits for first example

More about codelength

- Chaitin's hypothesis: "*comprehension is compression*"
- Expensive to compute for every training example, so split into subsets

$$\ell(\mathcal{A}) = |\mathcal{S}_1| \log_2 |\mathcal{Y}| - \sum_{i=2}^{M} \log_2 p(y_{\mathcal{S}_i} \mid x_{\mathcal{S}_i}; \hat{\mathbf{W}}_{\mathcal{S}_{i-1}})$$

- How to do for span selection tasks?

Main Tasks

READING COMPREHENSION

★ SQuAD 1.1

- questions constructed from Wikipedia passages
- 90k train / 10k val

TriviaQA

- trivia questions & answers, evidence from the web
- 76k train / 300 val

QuAC

- information-seeking dialogue, reponse spans from Wikipedia
- 80k train / 7k val

NATURAL LANGUAGE INFERENCE

★ MNLI

- multi-genre entailment
- 400k train / 20k test

SNLI

- 550k train / 10k test

TriviaQA (Joshi et al. 2017)

- QA pairs collected from 14 trivia websites
- Evidence filtered from Bing, Wikipedia
- Only documents which contain answer
- Multiple training examples per QA pair

Question: The Dodecanese Campaign of WWII that was an attempt by the Allied forces to capture islands in the Aegean Sea was the inspiration for which acclaimed 1961 commando film?
Answer: The Guns of Navarone
Excerpt: The Dodecanese Campaign of World War II was an attempt by Allied forces to capture the Italian-held Dodecanese islands in the Aegean Sea following the surrender of Italy in September 1943, and use them as bases against the German-controlled Balkans. The failed campaign, and in particular the Battle of Leros, inspired the 1957 novel **The Guns of Navarone** and the successful 1961 movie of the same name.

QuAC (Choi et al. 2018)

Figure 1: An example dialog about a Wikipedia section. The student, who does not see the section text, asks questions. The teacher provides a response in the form of a text span (or No answer), optionally yes or no (Yes / No), and encouragement about continuing a line of questioning (should, $\hookrightarrow$, could $\bar{\hookrightarrow}$, or should not $\not\hookrightarrow$ ask a follow-up question).

- Entire context as evidence
- Pros / cons of collection method?

Section: Daffy Duck, Origin & History

STUDENT: **What is the origin of Daffy Duck?**
TEACHER: $\hookrightarrow$ first appeared in Porky's Duck Hunt
STUDENT: **What was he like in that episode?**
TEACHER: $\hookrightarrow$ assertive, unrestrained, combative
STUDENT: **Was he the star?**
TEACHER: $\bar{\hookrightarrow}$ No, barely more than an unnamed bit player in this short
STUDENT: **Who was the star?**
TEACHER: $\not\hookrightarrow$ No answer
STUDENT: **Did he change a lot from that first episode in future episodes?**
TEACHER: $\hookrightarrow$ Yes, the only aspects of the character that have remained consistent (...) are his voice characterization by Mel Blanc
STUDENT: **How has he changed?**
TEACHER: $\hookrightarrow$ Daffy was less anthropomorphic
STUDENT: **In what other ways did he change?**
TEACHER: $\hookrightarrow$ Daffy's slobbery, exaggerated lisp (...) is barely noticeable in the early cartoons.
STUDENT: **Why did they add the lisp?**
TEACHER: $\hookrightarrow$ One often-repeated "official" story is that it was modeled after producer Leon Schlesinger's tendency to lisp.
STUDENT: **Is there an "unofficial" story?**
TEACHER: $\hookrightarrow$ Yes, Mel Blanc (...) contradicts that conventional belief
...

Other Tasks

SEMANTIC ROLE LABELING

FitzGerald et al. (2018)

- SRL as span-prediction
- 200k train / 25k test

In 1950 Alan M. Turing ***published*** "Computing machinery and intelligence" in Mind, in which he ***proposed*** that machines could be ***tested*** for intelligence ***using*** questions and answers.

Predicate		Question	Answer
published	1	Who published something?	Alan M. Turing
	2	What was published?	"Computing Machinery and Intelligence"
	3	When was something published?	In 1950

RELATION EXTRACTION

Levy et al. (2017)

- slot-filling as Q&A
- 900k train / 5k test

Relation	Question Template
$educated\_at(x, y)$	Where did x graduate from? In which university did x study? What is x's alma mater?
$occupation(x, y)$	What did x do for a living? What is x's job? What is the profession of x?
$spouse(x, y)$	Who is x's spouse? Who did x marry? Who is x married to?

Models

TRANSFORMER

BERT$_{BASE}$

- default vocabulary
- 110M parameters

RNN

ELMo + LSTM + BiDAF

- character-based
- 100M ELMo parameters

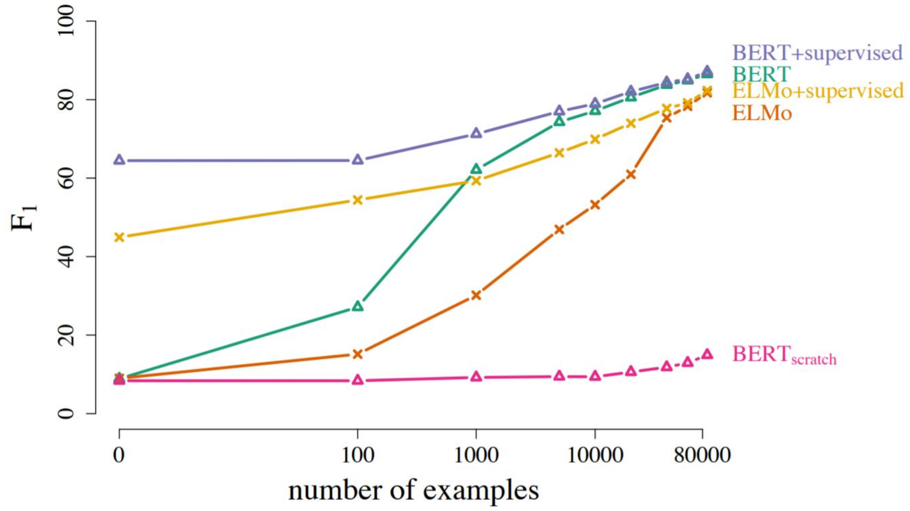
Experiments

1. How much in-domain training data is needed to obtain good performance?
2. Can pretraining on other datasets and tasks improve performance?
3. Do these models generalize to other datasets from the same task?
4. How fast do these models forget their previously acquired linguistic knowledge?
5. How does curriculum affect performance and how do we design this curriculum?

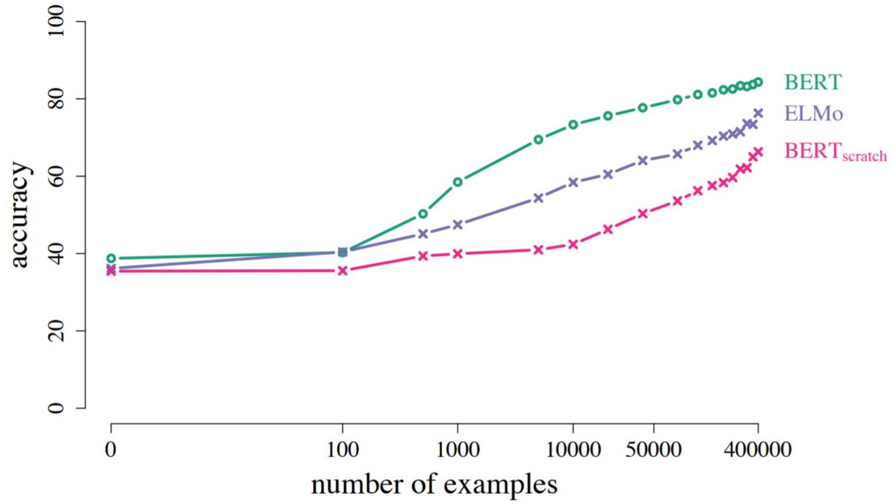
1

How much **in-domain training data** is needed to obtain good performance?

READING COMPREHENSION
(SQuAD 1.1)



NATURAL LANGUAGE INFERENCE
(MNLI)



Models need about **40,000** training examples

Codelengths?

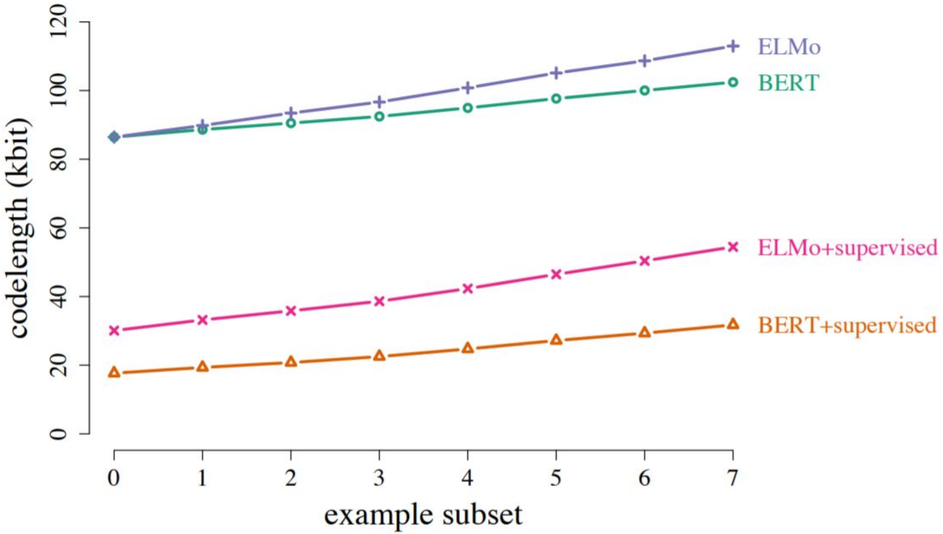
	SQuAD 1.1	MNLI
BERT	102.42 kbits	89.25 kbits
ELMo	112.96 kbits	132.17 kbits

2

Can **pretraining on other datasets and tasks** improve performance?

Pretrain on all supervised tasks (SRL, RE, MNLI, SNLI, TriviaQA, QuAC), then train on SQuAD.

Model	EM (↑)	F_1 (↑)	codelength (↓)
BERT	78.5	86.5	102.4
BERT + supervised	79.4	87.1	31.7
ELMo	72.1	81.8	113.0
ELMo + supervised	72.8	82.3	54.5



3

Do these models **generalize** to other datasets from the **same task**?

Evaluate best SQuAD model on other tasks.

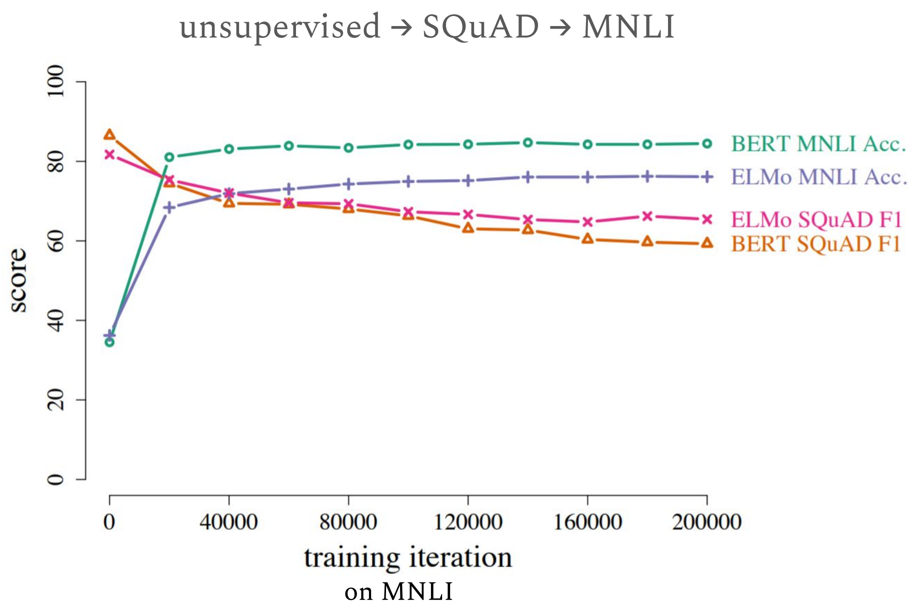
	SQuAD	Trivia	QuAC	QA-SRL	QA-ZRE
BERT	86.5 (78.5)	35.6 (13.4)	56.2 (43.9)	77.5 (65.0)	55.3 (40.0)
ELMo	81.8 (72.2)	32.9 (12.6)	45.0 (34.5)	68.7 (52.3)	60.2 (42.0)

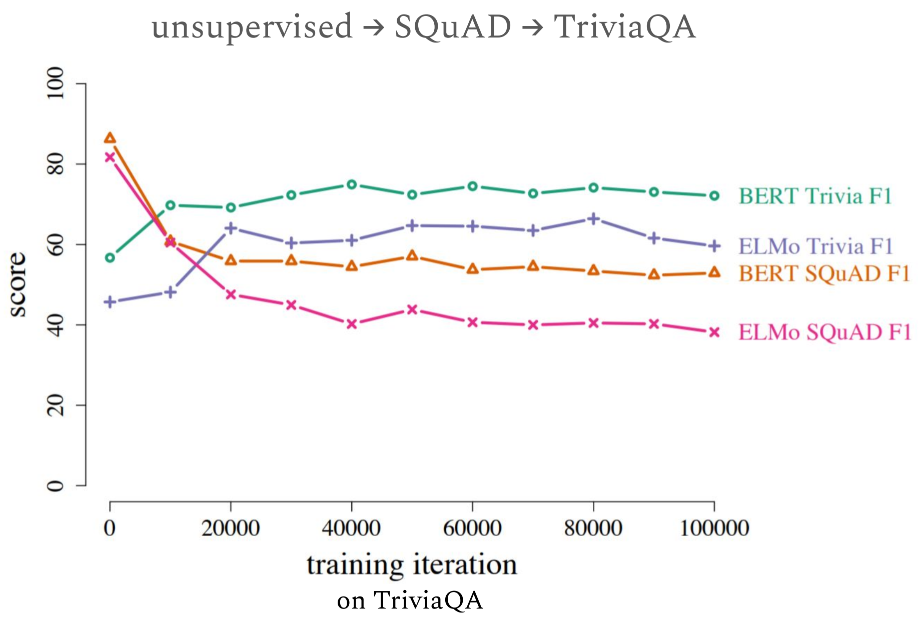
Table 2: F_1 (exact match) scores of the best BERT and ELMo models trained on SQuAD and evaluated on other question answering datasets.

4

How fast do these models forget their **previously acquired linguistic knowledge**?

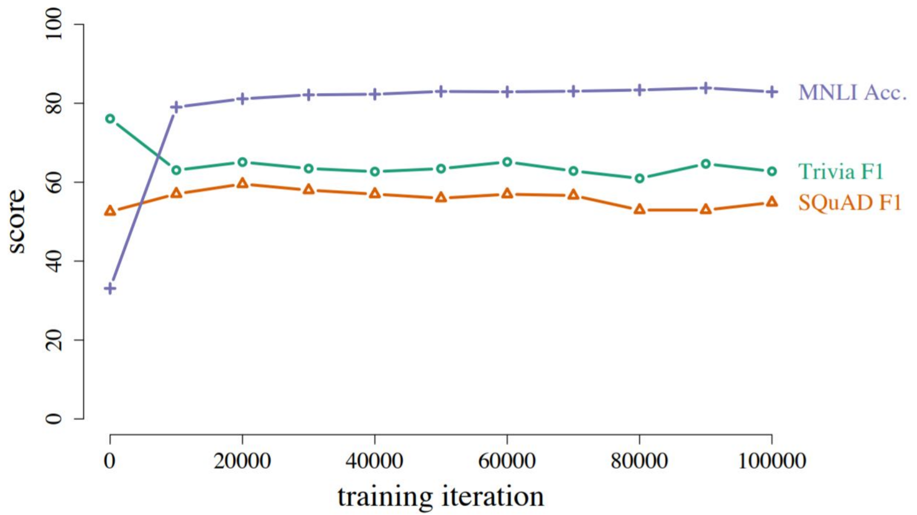
Train on one dataset at a time (“continual learning”).





Train on one dataset at a time (“continual learning”).

unsupervised → SQuAD → TriviaQA→ MNLI



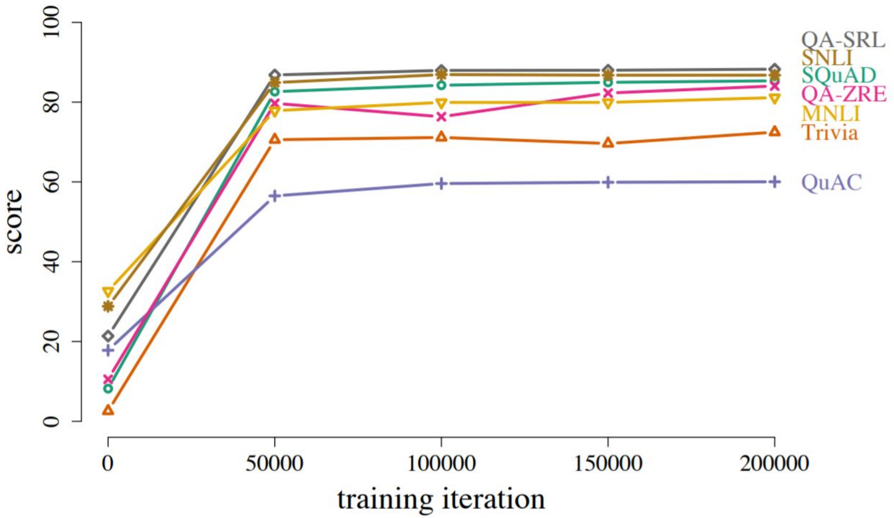
on MNLI

5

How does **curriculum** affect performance and how do we design this curriculum?

Train on all datasets at the same time (“random training curriculum” / “mixed curriculum”).

	SQuAD	Trivia	QuAC	QA-SRL	QA-ZRE	MNLI	SNLI
BERT	85.4	72.5	60.0	85.0	88.2	81.1	88.0
ELMo	78.3	57.1	54.3	67.3	88.5	69.1	77.9



Key Takeaways

- Current models **solve datasets**, **not tasks**. They need significant in-domain training data to attain good performance.
- Ability to **rapidly generalize** can and should be evaluated both *across* datasets and *within* datasets (using codelength, for example).
- Poor generalization is partly due to **task-specific components**, so we might look for ways to unify tasks (text-to-text framework, for example).
- Continual training does not work, as **models forget earlier training**. Only mixed training curricula lead to good multi-task models.

BAM! Born-Again Multi-Task Networks for Natural Language Understanding

Kevin Clark[†] **Minh-Thang Luong**[‡] **Urvashi Khandelwal**[†]
Christopher D. Manning[†] **Quoc V. Le**[‡]
[†]Computer Science Department, Stanford University
[‡]Google Brain
{kevclark,urvashik,manning}@cs.stanford.edu
{thangluong,qvl}@google.com

Outline

- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

Review of Distillation

- Train a teacher model
- Replace gold-label with teacher probability predictions
- E.g. from large model (teacher) to small model

Standard Training Objective

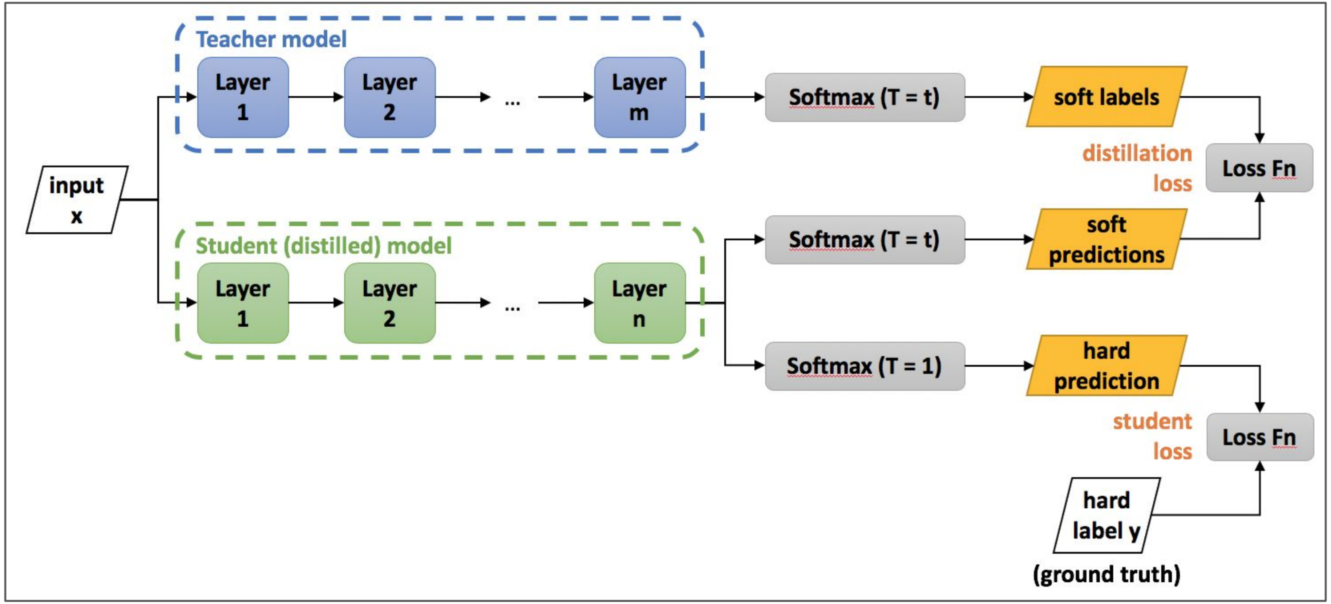
$$\mathcal{L}(\theta) = \sum_{x_i, y_i \in \mathcal{D}} l(y_i, f(x_i, \theta))$$

Distillation Training Objective

$$\mathcal{L}(\theta) = \sum_{x_i, y_i \in \mathcal{D}} l(f(x_i, \theta'), f(x_i, \theta))$$

Teacher's Parameters

Student's Parameters



Standard Training Objective

$$\mathcal{L}(\theta) = \sum_{x_i, y_i \in \mathcal{D}} l(y_i, f(x_i, \theta))$$

Teacher's Parameters

Distillation Training Objective

$$\mathcal{L}(\theta) = \sum_{x_i, y_i \in \mathcal{D}} l(f(x_i, \theta'), f(x_i, \theta))$$

Student's Parameters

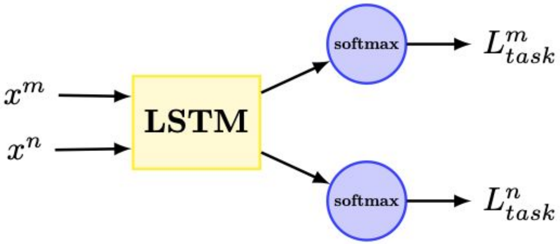
Outline

- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

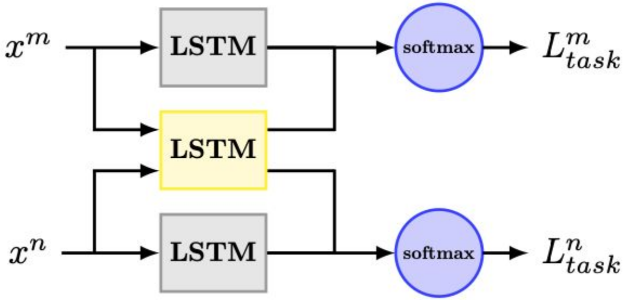
Background / Previous Work

- Multi-Task NLP Models
  - Design architecture to share only helpful information (Ruder et al. 2019)
  - BAM is orthogonal
- Distillation
  - Distillation is used in NLP from large -> small models (Kim and Rush 2016)
  - **Born-again models** (Furlanello et al. 2018)
    - Large -> Large (same size)
  - Distill single-language-pair translation models into a multi-language model (Tan et al 2019)
- Multi-task BERT: **MT-DNN** (Liu et al. 2019)

Multi-Task NLP: Architecture Changes



(a) Fully Shared Model (FS-MTL)



(b) Shared-Private Model (SP-MTL)

Background / Previous Work

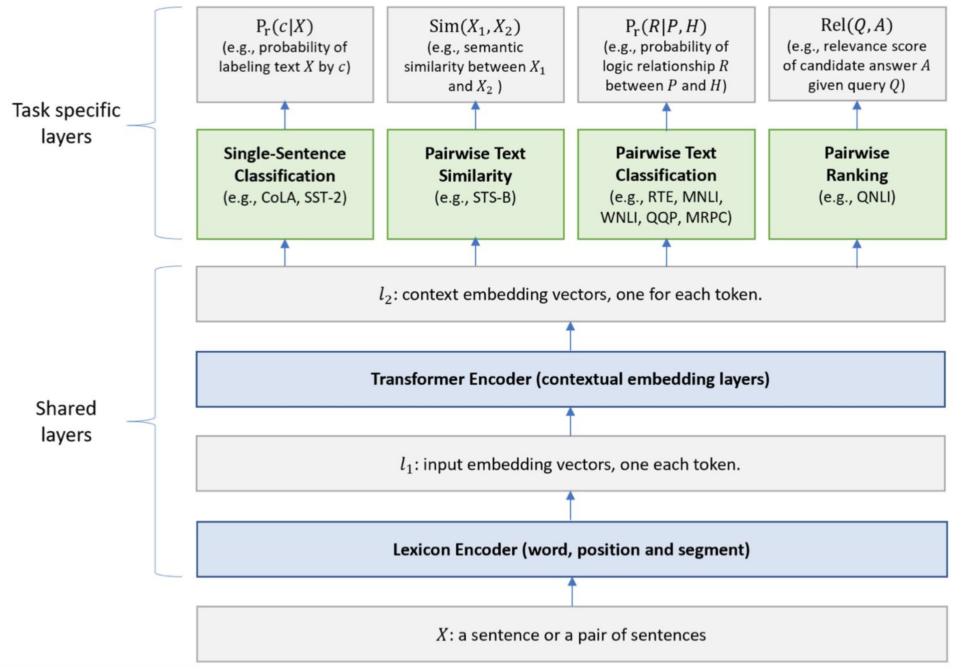
- **Born-again network** (Furlanello et al, 2018)
  - Variant of distillation
  - Teacher, student have same architecture
  - Surprisingly, student does better than teacher!

Network	Teacher	BAN
DenseNet-112-33	18.25	**16.95**
DenseNet-90-60	17.69	**16.69**
DenseNet-80-80	17.16	**16.36**
DenseNet-80-120	16.87	**16.00**

Test Error on CIFAR-100

Background / Previous Work

- Multi-task BERT: **MT-DNN** (Liu et al. 2019)
  - Mixed curriculum



Model	GLUE score
BERT-Base (Devlin et al., 2019)	78.5
BERT-Large (Devlin et al., 2019)	80.5
BERT on STILTs (Phang et al., 2018)	82.0
MT-DNN (Liu et al., 2019b)	82.2

Outline

- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

Context

- Catastrophic forgetting
- Is single-task fine-tuning necessary? **Mixed curriculum**?
  - Yogatama et al: Mixed curriculum does okay!
  - Performance lags
    - SQuAD: 86.5 -> 85.4
    - MNLI: 84.6 -> 81.1
  - MT-DNN: Mixed curriculum yields **stronger** performance!
- Can mixed curriculum beat fine-tuned models?
  - BAM: Yes, using some tricks
    - Distill many teachers into a single multi-task model
- Note: BAM doesn't resolve continual learning

Outline

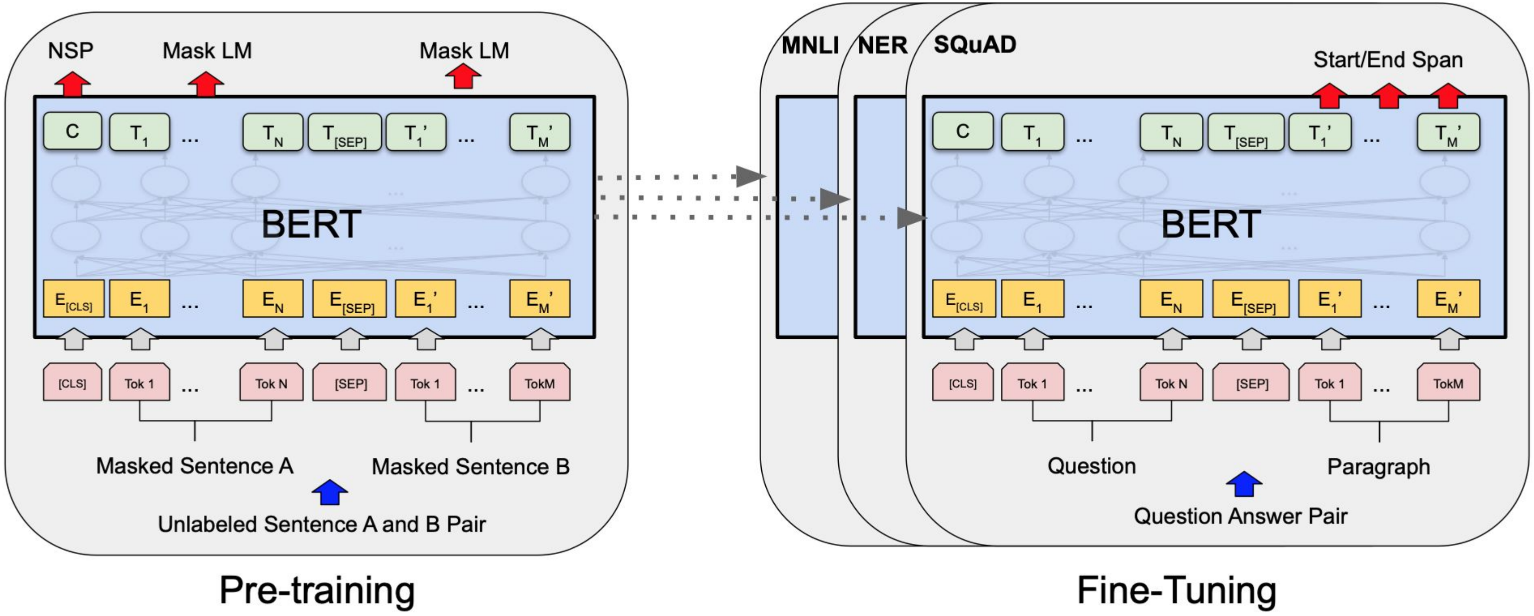
- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

BAM: General Approach

- **Multi-task BERT, with single-task teachers**
  - Train many single-task models, use as teachers for multi-task model
- Main tricks:
  - Many teachers, one per task
  - Born-again: same architecture
  - Teacher annealing
  - Task sampling
  - Different learning rate per layer

Review: Single-Task BERT

- Pre-train using language modeling
- Fine-tune a **different** BERT model for each single task



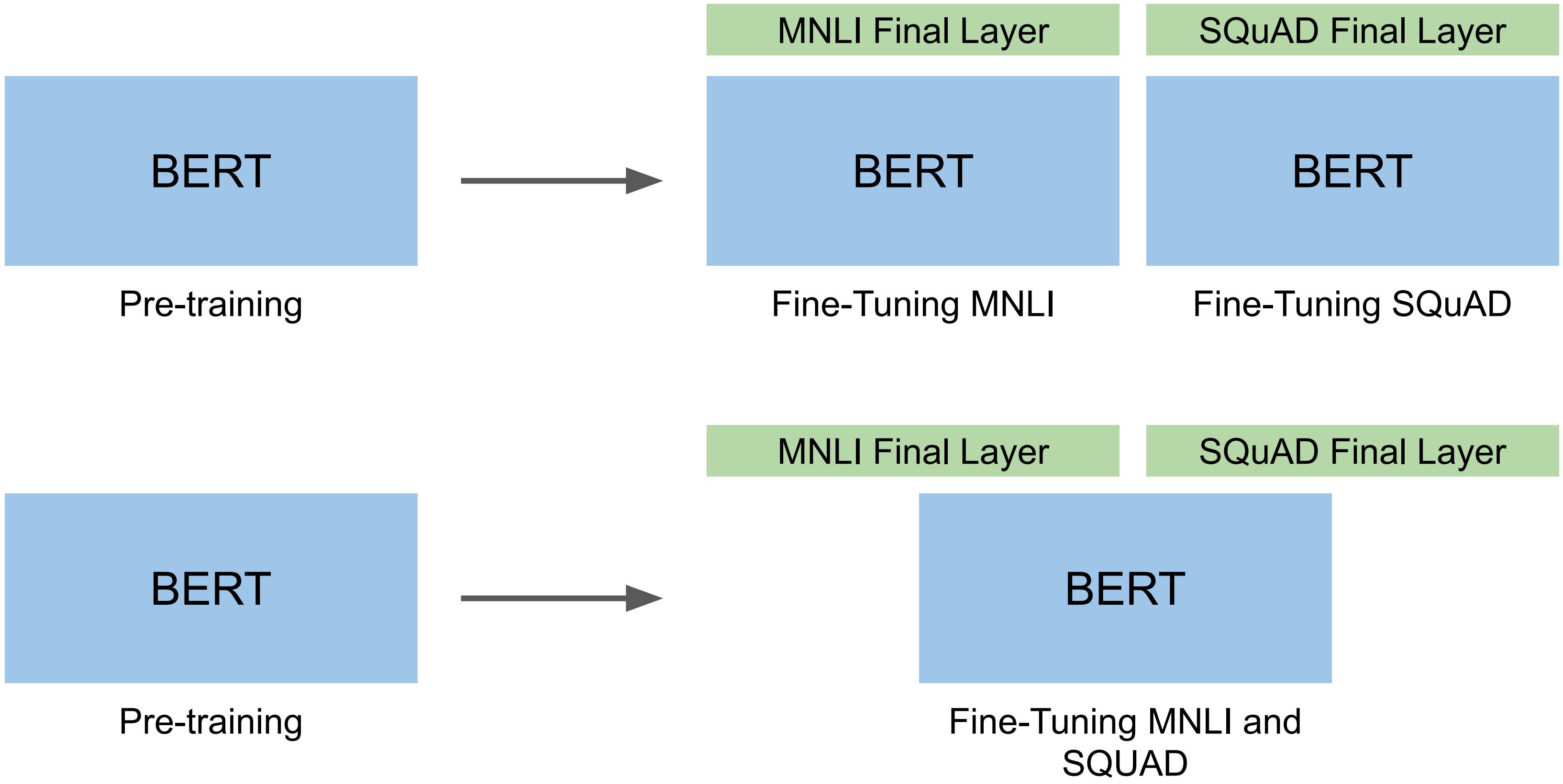
Review: Single-Task BERT

- Pre-train using language modeling
- Fine-tune a different BERT model for each single task
  - Add a new final layer on top of the pre-trained network
  - For classification tasks, use softmax
    - softmax(W c)
  - For regression tasks, normalize labels and use sigmoid activation
    - sigmoid(w^T c)

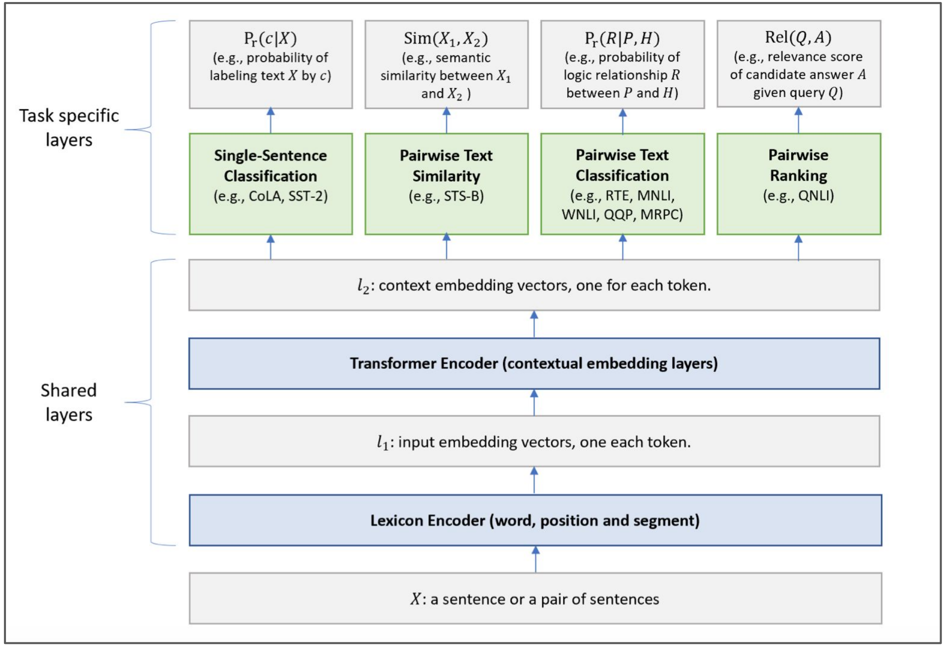
Multi-task BERT in BAM

- Same architecture as standard BERT
- For multi-task model, **only change final layer**
  - All other parameters shared between tasks!
- **Mixed curriculum**
  - Different tasks are mixed
  - Each minibatch contains multiple tasks
- Training objective
  - Either use standard gold-label training
  - Or use distillation (using a BERT teacher)
    - Born-again, since teacher has same architecture as student
    - Clarify: Single vs Multi-task teacher

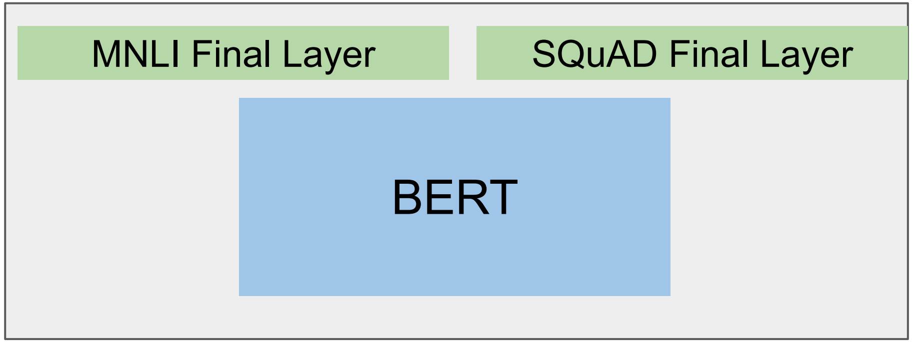
Single vs. Multi-task BERT



BAM vs. MT-DNN



$P_r(c|X)$
(e.g., probability of labeling text X by c)
$\text{Sim}(X_1, X_2)$
(e.g., semantic similarity between X_1 and X_2)
$P_r(R|P,H)$
(e.g., probability of logic relationship R between P and H)
$\text{Rel}(Q,A)$
(e.g., relevance score of candidate answer A given query Q)
Task specific layers
Single-Sentence Classification (e.g., CoLA, SST-2)
Pairwise Text Similarity (e.g., STS-B)
Pairwise Text Classification (e.g., RTE, MNLI, WNLI, QQP, MRPC)
Pairwise Ranking (e.g., QNLI)
l_2: context embedding vectors, one for each token.
Transformer Encoder (contextual embedding layers)
Shared layers
l_1: input embedding vectors, one each token.
Lexicon Encoder (word, position and segment)
X: a sentence or a pair of sentences



MNLI Final Layer
SQuAD Final Layer
BERT

Outline

- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

Teacher Annealing

- Standard distillation: Either train on teacher outputs or gold label
- Teacher annealing: Mix teacher outputs and gold label
- Gradually increase lambda to 1 (use gold labels more over time)

$$l(y_i, f(x_i, \theta)) \qquad l(f(x_i, \theta'), f(x_i, \theta))$$

$$l(\lambda y_i + (1 - \lambda) f(x_i, \theta'), f(x_i, \theta))$$

Other Tricks

- Task Sampling (Bowman et al 2018)
  - Sample an example from a task proportionally to the ¾ root of the size of dataset for that task (slightly downweight examples from large datasets)
  - $|\mathcal{D}_\tau|^{0.75}$
- Layerwise-learning-rate (Howard and Ruder 2018)
  - Different learning rate for each layer: BASE_LR * α^d
  - Layers closest to input get lower learning rate
  - $\alpha = 0.9$ for multi-task models

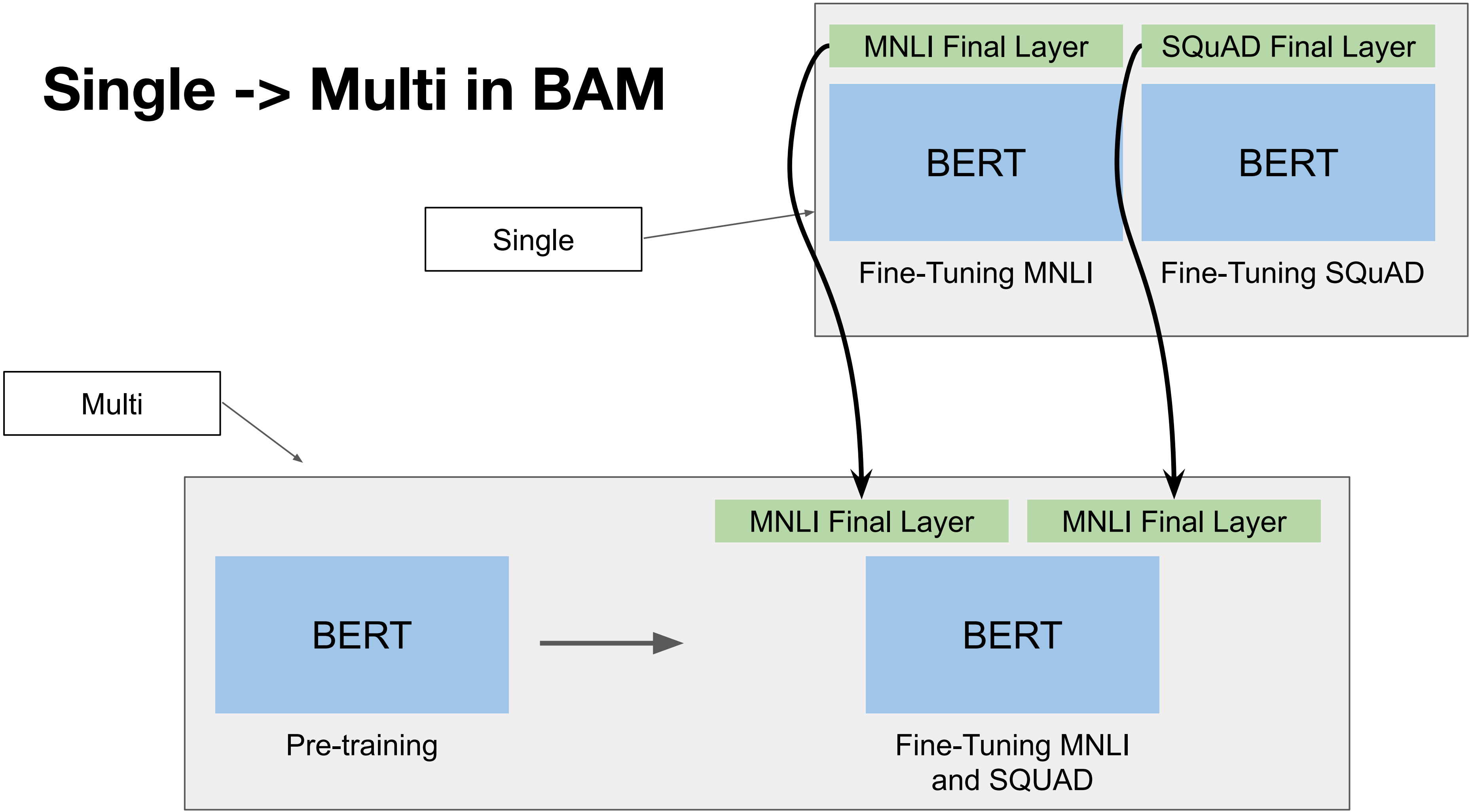
Outline

- Review of Distillation
- Background / Previous Work
- Context
- BAM's General Approach
- Tricks
- Experiments / Results

Experimental Setup

- Evaluate on GLUE
  - Collection of tasks including question answering, sentiment analysis, and textual entailment
- Compare various versions of BERT
  - **Single** (standard BERT, single-task fine-tuning)
  - **Multi** (mixed curriculum, gold labels)
  - **Single** -> **Single** (standard BERT, single-task fine-tuning, teachers are single-task learners)
  - **Single** -> **Multi** (mixed curriculum, teachers are single-task learners)
  - **Multi** -> **Multi** (mixed curriculum, teachers are multi-task)
  - **Single** -> **Multi** -> **Single** -> **Multi** (multiple rounds of distillation)

Single -> Multi in BAM



Review: GLUE

- Single-sentence tasks
  - CoLA (Is this sentence grammatical?)
  - SST-2 (Sentiment analysis: Is this sentence positive or negative?)
- Similarity and Paraphrase Tasks
  - MRPC, QQP, STS-B (Are these sentences semantically equivalent?)
- Inference Tasks
  - MNLI, QNLI, RTE, WNLI
  - (What is the relationship between these sentences? Entailment, contradiction, or neutral?)

Results

Model	**Avg.**	**CoLA**[a] $|\mathcal{D}| = 8.5k$	**SST-2**[b] 67k	**MRPC**[c] 3.7k	**STS-B**[d] 5.8k	**QQP**[e] 364k	**MNLI**[f] 393k	**QNLI**[g] 108k	**RTE**[h] 2.5k
Single	84.0	60.6	93.2	88.0	90.0	91.3	86.6	92.3	70.4
Multi	85.5	60.3	93.3	88.0	89.8	91.4	86.5	92.2	82.1
Single→Single	84.3	**61.7****	93.2	**88.7***	90.0	91.4	**86.8****	**92.5*****	70.0
Multi→Multi	85.6	60.9	93.5	88.1	89.8	**91.5***	86.7	92.3	82.0
Single→Multi	**86.0*****	**61.8****	**93.6***	**89.3****	89.7	**91.6***	**87.0*****	**92.5*****	**82.8***

Dataset references: [a]Warstadt et al. (2018) [b]Socher et al. (2013) [c]Dolan and Brockett (2005) [d]Cer et al. (2017) [e]Iyer et al. (2017) [f]Williams et al. (2018) [g]constructed from SQuAD (Rajpurkar et al., 2016) [h]Giampiccolo et al. (2007)

Table 1: Comparison of methods on the GLUE dev set. *, **, and *** indicate statistically significant ($p < .05$, $p < .01$, and $p < .001$) improvements over both Single and Multi according to bootstrap hypothesis tests.[3]

Model	Avg.	CoLA[a] $\lvert\mathcal{D}\rvert$ = 8.5k	SST-2[b] 67k	MRPC[c] 3.7k	STS-B[d] 5.8k	QQP[e] 364k	MNLI[f] 393k	QNLI[g] 108k	RTE[h] 2.5k
Single	84.0	60.6	93.2	88.0	90.0	91.3	86.6	92.3	70.4
Multi	85.5	60.3	93.3	88.0	89.8	91.4	86.5	92.2	82.1
Single→Single	84.3	**61.7****	93.2	**88.7***	90.0	91.4	**86.8****	**92.5*****	70.0
Multi→Multi	85.6	60.9	93.5	88.1	89.8	**91.5***	86.7	92.3	82.0
Single→Multi	**86.0*****	**61.8****	**93.6***	**89.3****	89.7	**91.6***	**87.0*****	**92.5*****	**82.8***

Trained Tasks	RTE score
RTE	70.0
RTE + MNLI	83.4
RTE + QQP + CoLA + SST	75.1
All GLUE	82.8

Table 5: Which tasks help RTE? Pairwise differences are statistically significant ($p < .01$) according to Mann-Whitney U tests.[3]

Results

Model	GLUE score
BERT-Base (Devlin et al., 2019)	78.5
BERT-Large (Devlin et al., 2019)	80.5
BERT on STILTs (Phang et al., 2018)	82.0
MT-DNN (Liu et al., 2019b)	82.2
Span-Extractive BERT on STILTs (Keskar et al., 2019)	82.3
Snorkel MeTaL ensemble (Hancock et al., 2019)	83.2
MT-DNN$_{KD}$* (Liu et al., 2019a)	83.7
BERT-Large + BAM **(ours)**	82.3

Table 2: Comparison of test set results. *MT-DNN$_{KD}$ is distilled from a diverse ensemble of models.

Ablation

Model	Avg. Score
Single→Multi	86.0
No layer-wise LRs	-0.3
No task sampling	-0.4
No teacher annealing: $\lambda = 0$	-0.5
No teacher annealing: $\lambda = 0.5$	-0.3

Table 4: Ablation Study. Differences from Single→Multi are statistically significant ($p < .001$) according to Mann-Whitney U tests.[3]

Model	Avg.
Single	84.0
Multi	85.5
Single→Single	84.3
Multi→Multi	85.6
Single→Multi	**86.0****

Conclusion and Caveats

- Multi-task training can perform **better** than single-task training!
- Tricks are important!
  - Teacher annealing, layer-wise learning rate, task sampling
- Single-task fine-tuning isn't necessary?
- BAM doesn't solve continual learning: need mixed curriculum

Criteria from the paper:

(i) deal with the full complexity of natural language across a variety of tasks

(ii) effectively store and reuse representations, combinatorial modules, and previously acquired linguistic knowledge to avoid *catastrophic forgetting*

(iii) adapt to new linguistic tasks in new environments with little experience