



UNIVERSITY OF  
**ILLINOIS**  
URBANA-CHAMPAIGN

# Exploring and Predicting Transferability across NLP Tasks

by Tu Vu, Tong Wang, Tsendsuren Munkhdalai, Alessandro Sordoni, Adam Trischler,  
Andrew Mattarella-Micke, Subhransu Maji, Mohit Iyyer

ECE 594 paper presentation

Po-Hao Wu

Mar 3, 2022



# Outline

---

Introduction  
Method  
Conclusion  
Discussion



# Introduction

index	sentence1	sentence2	label
0	I stuck a	United Nations official Ekeus heads for Baghdad	1
		NP B-NP B-ORG I-NP I-ORG NN I-NP O NNP B-NP B-PER VBZ B-VP O IN B-PP O NNP B-NP B-LOC	
4	When T	George was particu	0
5	George got the tick	George was particu	0

```

{"data": [{"title": "", "paragraphs":
{"question": "what country was eliz
"answer_start": 3477}, {"text":
"answer_start": 4079}, {"text":
"answer_start": 8735}, {"text":

```

```

925059f5e841a7ec2d0d8b9",
[{"text": "new york city",
{"text": "new york city",
{"text": "new york city",
, {"text": "new york

```

## Goal of this paper

- **Study of the transferability between 33 NLP tasks**
  - Text classification
  - Question answering
  - Sequence labeling
- **Transfer learning is more beneficial when source tasks differ substantially from the target task.**
- **Using task embeddings to predict the most transferable source tasks**



# Method

## Pipeline

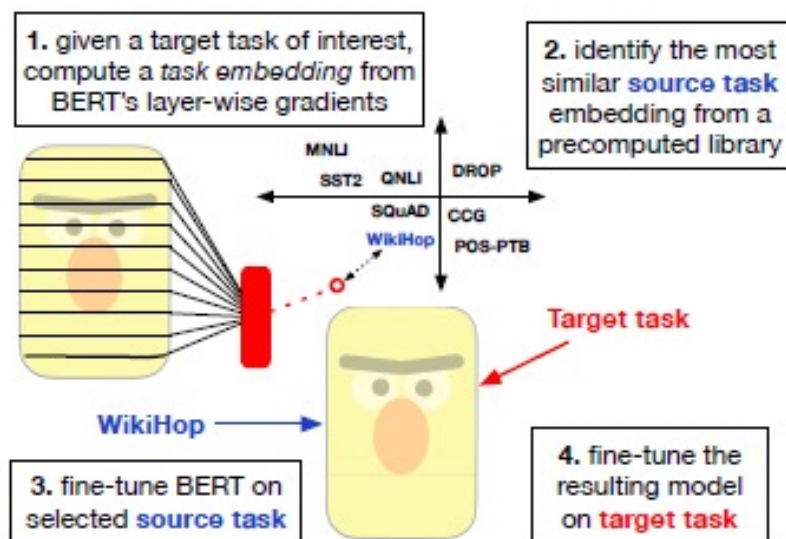


Figure 1: A demonstration of our task embedding pipeline. Given a target task, we first compute its task embedding and then identify the most similar source task embedding (in this example, WikiHop) from a pre-computed library via cosine similarity. Finally, we perform intermediate fine-tuning of BERT on the selected source task before fine-tuning on the target task.<sup>1</sup>

# Datasets


Task	Train	Task type	Domain	
<i>text classification/regression (CR)</i>				
SNLI (Bowman et al., 2015)	570K	NLI	misc.	
MNLI (Williams et al., 2018)	393K	NLI	misc.	
QQP (Iyer et al., 2017)	364K	paraphrase identification	social QA	
QNLI (Wang et al., 2019b)	105K	QA-NLI	Wikipedia	
SST-2 (Socher et al., 2013)	67K	sentiment analysis	movie reviews	
SciTail (Khot et al., 2018)	27K	NLI	science QA	
CoLA (Warstadt et al., 2019)	8.5K	grammatical acceptability	misc.	
STS-B (Cer et al., 2017)	7K	semantic similarity	misc.	
MRPC (Dolan and Brockett, 2005)	3.7K	paraphrase identification	news	
RTE (Dagan et al., 2005, et seq.)	2.5K	NLI	news, Wikipedia	
WNLI (Levesque, 2011)	634	coreference NLI	fiction books	43K
<i>question answering (QA)</i>				
SQuAD-2 (Rajpurkar et al., 2018)	162K	QA	Wikipedia, crowd	d Steedman, 2007)
NewsQA (Trischler et al., 2017)	120K	QA	news, crowd	a)
HotpotQA (Yang et al., 2018)	113K	multi-hop QA	Wikipedia, crowd	19a)
SQuAD-1 (Rajpurkar et al., 2016)	108K	QA	Wikipedia, crowd	019a)
DuoRC-p (Saha et al., 2018)	100K	paraphrased QA	Wikipedia/IMDB, crowd	l., 1993)
DuoRC-s (Saha et al., 2018)	86K	paraphrased QA	Wikipedia/IMDB, crowd	:t al., 2011)
DROP (Dua et al., 2019)	77K	multi-hop quantitative reasoning	Wikipedia, crowd	and De Meulder, 2003)
WikiHop (Welbl et al., 2018)	51K	multi-hop QA	Wikipedia, KB	al., 2014)
BoolQ (Clark et al., 2019)	16K	natural yes/no QA	Wikipedia, web queries	erg, 2016)
ComQA (Abujabal et al., 2019)	11K	factoid QA w/ paraphrases	snippets, WikiAnswers	ig and Buchholz, 2000)
CQ (Bao et al., 2016)	2K	knowledge-based QA	snippets, web queries/KB	9K
<i>sequence labeling (SL)</i>				
ST (Bjerva et al., 2016)	43K	semantic tagging	Groningen Meaning Bank	
CCG (Hockenmaier and Steedman, 2007)	40K	CCG supertagging	Penn Treebank	
Parent (Liu et al., 2019a)	40K	syntactic tagging	Penn Treebank	
GParent (Liu et al., 2019a)	40K	syntactic tagging	Penn Treebank	
GGParent (Liu et al., 2019a)	40K	syntactic tagging	Penn Treebank	
POS-PTB (Marcus et al., 1993)	38K	part-of-speech tagging	Penn Treebank	
GED (Yannakoudakis et al., 2011)	29K	grammatical error detection	misc.	
NER (Tjong Kim Sang and De Meulder, 2003)	14K	named entity recognition	news	
POS-EWT (Silveira et al., 2014)	13K	part-of-speech tagging	Web Treebank	
Conj (Ficler and Goldberg, 2016)	13K	conjunct identification	Penn Treebank	
Chunk (Tjong Kim Sang and Buchholz, 2000)	9K	syntactic chunking	Penn Treebank	

Table 4: Datasets used in our experiments and their characteristics, grouped by task class and sorted by training dataset size.



## Experimental setup

$$D = \{(x^i, y^i)\}_{i=1}^n$$

 [CLS]  $w_1^1$   $w_2^1$  ...  $w_{L_1}^1$  [SEP]  $w_1^2$   $w_2^2$  ...  $w_{L_2}^2$

- each task is solved by applying a classification layer over either the final [CLS] token representation (for CR) or the entire sequence of final layer token representations (for QA or SL).
- fine-tunes all CR and QA tasks for three epochs, and SL tasks for six epochs

## relative transfer gain

$$g_{s \rightarrow t} = \frac{p_{s \rightarrow t} - p_t}{p_t}$$

- In-class transfer
- Out-of-class transfer

## Summary of this findings:

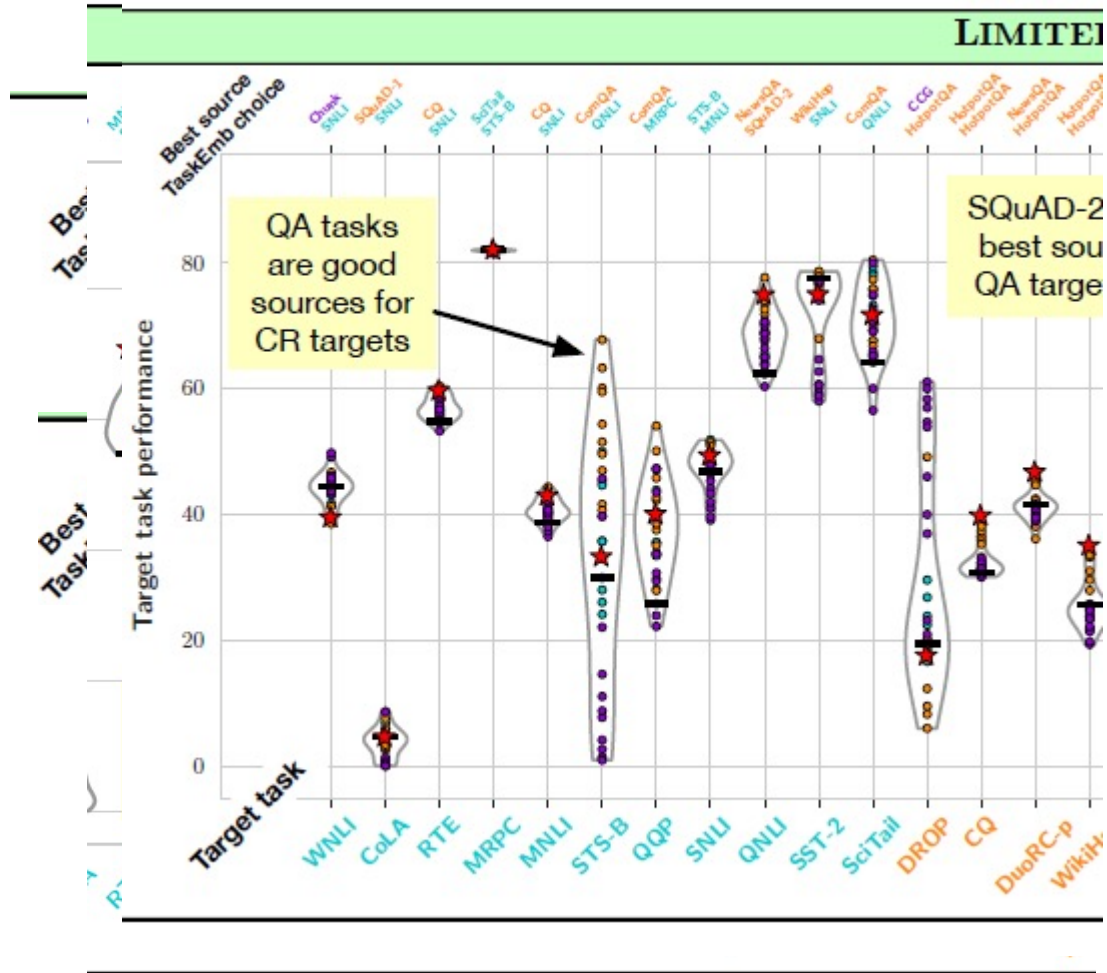
- transfer gains are possible even when the source dataset is small.
- Out-of-class transfer succeeds in many cases, some of which are unintuitive.
- Factors other than source dataset size, such as the similarity between source and target tasks, matter more in low-data regimes.

FULL → FULL			
↓src,tgt→	CR	QA	SL
CR	6.3 (11)	3.4 (10)	0.3 (10)
QA	3.2 (10)	9.5 (11)	0.3 (9)
SL	5.3 (8)	2.5 (10)	0.5 (11)
FULL → LIMITED			
	CR	QA	SL
CR	56.9 (11)	36.8 (10)	2.0 (10)
QA	44.3 (11)	63.3 (11)	5.3 (11)
SL	45.6 (11)	39.2 (6)	20.9 (11)
LIMITED → LIMITED			
	CR	QA	SL
CR	23.7 (11)	7.3 (11)	1.1 (11)
QA	37.3 (11)	49.3 (11)	4.2 (11)
SL	29.3 (10)	30.0 (8)	10.2 (11)

Table 2: A summary of our transfer results for each combination of the three task classes in the three data regimes. Each cell represents the relative gain of the *best* source task in the source class (row) for a given target task, averaged across all of target tasks in the target class (column). In parentheses, we additionally report the number of target tasks (out of 11) for which at least one source task results in a positive transfer gain. The diagonal cells indicate in-class transfer.



## In-class & Out-of-class transfer



- Large source datasets are not always best for data-constrained target tasks
- the similarity between the source and target tasks matters more for data-constrained targets
- QA tasks is domain similarity (e.g., SQuAD and several other datasets were all built from Wikipedia)

## Task embedding methods

### TEXTEMB

- Computed by pooling BERT's representations across an entire dataset
- Captures properties of the text and domain.
- Final task embedding is  $\sum_{x \in D} \frac{hx}{|D|}$

## Task embedding methods

### TASKEMB

- correlation between the fine-tuning loss function and the parameters of BERT
- Encodes more information about the type of knowledge and reasoning required to solve the task
- create representations of tasks derived from the Fisher information matrix  
=> which of the model parameters are most useful for the task and provides a rich source of knowledge about the task

$$F_{\theta} = \mathbb{E}_{x,y \sim P_{\theta}(x,y)} \nabla_{\theta} \log P_{\theta}(y|x) \nabla_{\theta} \log P_{\theta}(y|x)^T$$

## Task embedding evaluation

### Evaluation metrics

- (1) the average rank  $\rho$  of the source task with the highest absolute transfer gain
- (2) Normalized Discounted Cumulative Gain(NDCG),  
a common information retrieval measure that evaluates the quality of the entire ranking

$$\text{NDCG}_p = \frac{\text{DCG}_p(R_{\text{pred}})}{\text{DCG}_p(R_{\text{true}})}$$

$$\text{DCG}_p(R) = \sum_{i=1}^p \frac{2^{\text{rel}_i} - 1}{\log_2(i + 1)}$$

## Source task selection experiments

Method	FULL → FULL				FULL → LIMITED				LIMITED → LIMITED			
	<i>in-class (10)</i>		<i>all-class (32)</i>		<i>in-class (10)</i>		<i>all-class (32)</i>		<i>in-class (10)</i>		<i>all-class (32)</i>	
	$\rho$	NDCG	$\rho$	NDCG	$\rho$	NDCG	$\rho$	NDCG	$\rho$	NDCG	$\rho$	NDCG
<i>classification / regression</i>												
DATA SIZE	3.6	80.4	8.5	74.7	3.8	62.9	9.8	54.6	-	-	-	-
CURVEGRAD	5.5	68.6	17.8	64.9	6.4	45.2	18.8	35.0	5.9	50.8	13.3	42.4
TEXT EMB	5.2	76.4	13.1	71.3	3.5	60.3	8.6	52.4	4.8	61.4	13.2	43.9
TASK EMB	2.8	82.3	6.2	76.7	3.4	68.2	8.2	60.9	4.2	62.6	11.6	44.8
TEXT+TASK	2.6	83.3	5.6	78.0	3.3	69.5	8.2	62.0	4.2	62.7	11.4	44.8
<i>question answering</i>												
DATA SIZE	3.2	84.4	13.8	63.5	2.3	77.0	13.6	40.2	-	-	-	-
CURVEGRAD	8.3	64.8	15.7	55.0	8.2	49.1	16.7	32.8	6.8	53.4	15.3	40.1
TEXT EMB	4.1	81.1	6.8	79.7	2.7	77.6	4.1	77.0	4.1	65.6	7.6	66.5
TASK EMB	3.2	84.5	6.5	81.6	2.5	78.0	4.0	79.0	3.6	67.1	7.5	68.5
TEXT+TASK	3.2	85.9	5.4	82.5	2.2	81.2	3.6	82.0	3.6	66.5	7.0	69.6
<i>sequence labeling</i>												
DATA SIZE	7.9	90.5	19.2	91.6	4.3	63.2	20.3	34.0	-	-	-	-
CURVEGRAD	5.6	92.6	14.6	92.8	8.0	40.7	17.9	30.8	7.0	53.2	18.6	40.8
TEXT EMB	3.7	95.0	10.4	95.3	3.9	65.1	8.5	61.1	5.0	67.2	10.1	63.8
TASK EMB	3.4	95.7	9.6	95.2	2.7	80.5	4.4	76.3	2.5	82.1	5.5	76.9
TEXT+TASK	3.3	96.0	9.6	95.2	2.7	80.3	4.2	78.4	2.5	82.5	5.3	76.9

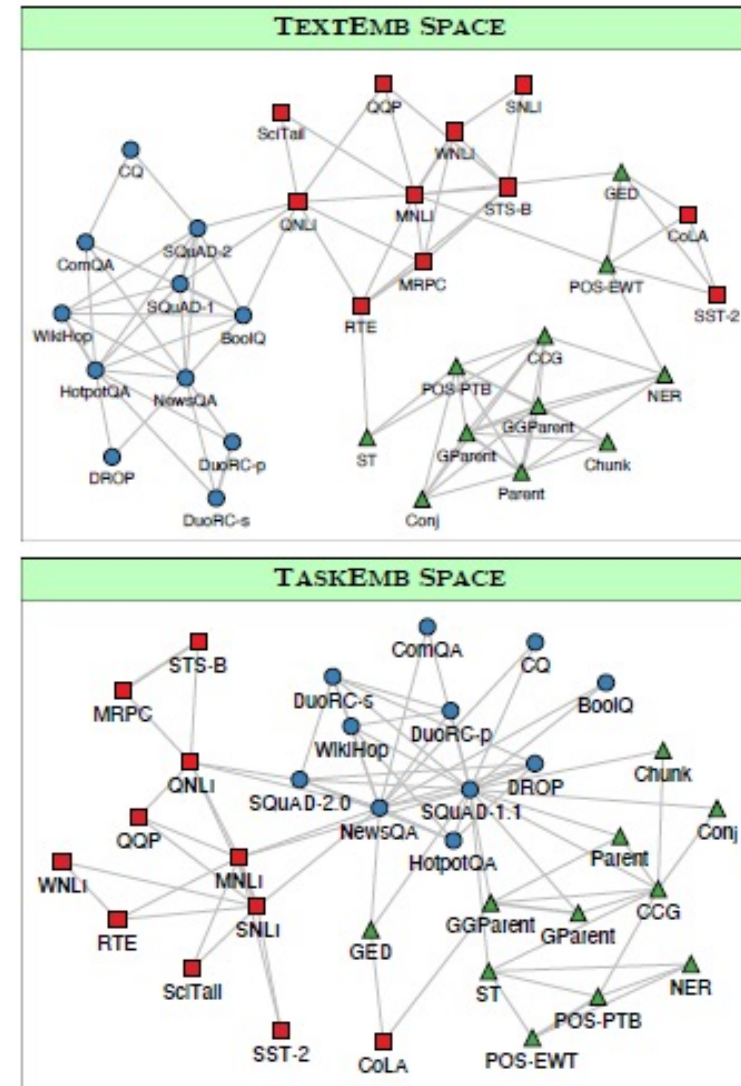
Table 3: To evaluate our embedding methods, we measure the average rank ( $\rho$ ) that they assign to the best source task (i.e., the one that results in the largest transfer gain) across target tasks, as well as the average NDCG measure of the overall ranking’s quality. In parentheses, we show the number of source tasks in each setting. Combining the complementary signals in TASK EMB and TEXT EMB consistently decreases  $\rho$  (lower is better) and increases NDCG across all settings, and both methods in isolation generally perform better than the baseline methods.



## Task embedding spaces

- the different task spaces in the FULL→FULL regime using the Fruchterman-Reingold force-directed placement algorithm (Fruchterman and Reingold, 1991).
- The task space of TEXTEMB shows that datasets with similar sources are near one another
- TASKEMB captures domain information to some extent, and it also encodes task similarity

$$f(t_1, t_2) = \frac{1}{r_{\rightarrow t_2}(t_1)} + \frac{1}{r_{\rightarrow t_1}(t_2)}$$





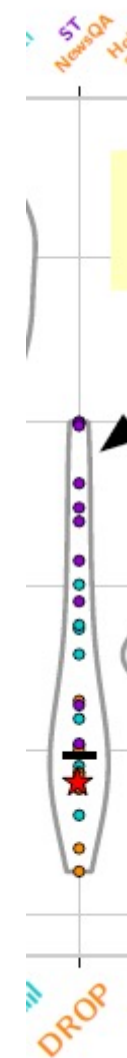
# Conclusion

## Highlight

- Transfer learning on a large-scale empirical study of the transferability between 33 NLP tasks performs well, especially when target training data is limited
- task embeddings allow us to predict source tasks that will likely improve target task performance.
- data size, the similarity between the source and target tasks, domains, and task complexity are crucial for effective transfer

## Limitation & Future work

- Selected epochs are different among three classes
- some of the results are not intuitive , such as using part-of-speech tagging as a source task for DROP results
- methods clearly do not capture all of the factors that influence task transferability





# Discussion