CogTaskonomy: Cognitively Inspired Task Taxonomy Is Beneficial to Transfer Learning in NLP

Yifei Luo[†] **, Minghui Xu**[†] and **Deyi Xiong**[*]
College of Intelligence and Computing, Tianjin University, Tianjin, China
{yfluo,xuminghui,dyxiong}@tju.edu.cn

Abstract

Is there a principle to guide transfer learning across tasks in natural language processing (NLP)? Taxonomy (Zamir et al., 2018) finds that a structure exists among visual tasks, as a principle underlying transfer learning for them. In this paper, we propose a cognitively inspired framework, CogTaskonomy, to learn taxonomy for NLP tasks. The framework consists of Cognitive Representation Analytics (CRA) and Cognitive-Neural Mapping (CNM). The former employs Representational Similarity Analysis, which is commonly used in computational neuroscience to find a correlation between brain-activity measurement and computational modeling, to estimate task similarity with task-specific sentence representations. The latter learns to detect task relations by projecting neural representations from NLP models to cognitive signals (i.e., fMRI voxels). Experiments on 12 NLP tasks, where BERT/TinyBERT are used as the underlying models for transfer learning, demonstrate that the proposed CogTaskonomy is able to guide transfer learning, achieving performance competitive to the Analytic Hierarchy Process (Saaty, 1987) used in visual Taskonomy (Zamir et al., 2018) but without requiring exhaustive pairwise $O(m^2)$ task transferring. Analyses further discover that CNM is capable of learning model-agnostic task taxonomy. The source code is available at https://github.com/tjunlp-lab/CogTaskonomy.git.

1 Introduction

Transfer learning (TL) has attracted extensive research interests in natural language processing with a wide range of forms, e.g., TL from pretrained language models (PLM) to downstream tasks (Devlin et al., 2018; Radford et al., 2018), from a task with rich labeled data to a task with low resource (Chu and Wang, 2018; Yu et al., 2021), from high-resource languages to low-resource languages (Gu et al., 2018; Ko et al., 2021), etc.[1] A high-level concept or question on cross-task transfer learning is how these involved tasks are related to each other. Is sentiment analysis related to paraphrasing? Is textual entailment more related to question answering than named entity recognition? All these sub-questions resolve themselves into whether a structure exists among NLP tasks. Such task taxonomy is of notable values to transfer learning in NLP in that it has the potential to guide TL and reduce redundancies across tasks (Zamir et al., 2018).

In this paper, partially inspired by the task taxonomy in visual tasks (Zamir et al., 2018), we study the hierarchical task structure for NLP tasks. But significantly different from the visual Taskonomy (Zamir et al., 2018), we construct NLP taskonomy from a cognitively inspired perspective.

Cognitively inspired NLP is the intersection of NLP and cognitive neuroscience that aims at uncovering cognitive processes in the brain, including cognition in language comprehension. With the increasing availability of cognitively annotated data, on the one hand, cognitive processing signals (e.g., eye-tracking, EEG, fMRI) have been explored to enhance neural models for a wide range of NLP tasks (Barrett and Søgaard, 2015; Bingel et al., 2016; Hollenstein and Zhang, 2019; Hollenstein et al., 2019a). On the other hand, representations learned in NLP models are used to predict brain activation patterns recorded in cognitive processing data (Mitchell et al., 2008; Pereira et al., 2018; Hale et al., 2018; Hollenstein et al., 2019b). These studies on the bidirectional association between the two areas demonstrate that information underlying cognitive processing data is closely related to tasks and representations in NLP. Hence we want to know whether it is feasible to isolate task repre-

[†]Equal contribution.
[*]Corresponding author.

[1]In this paper, we focus on cross-task transfer learning in the same language.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 904 - 920

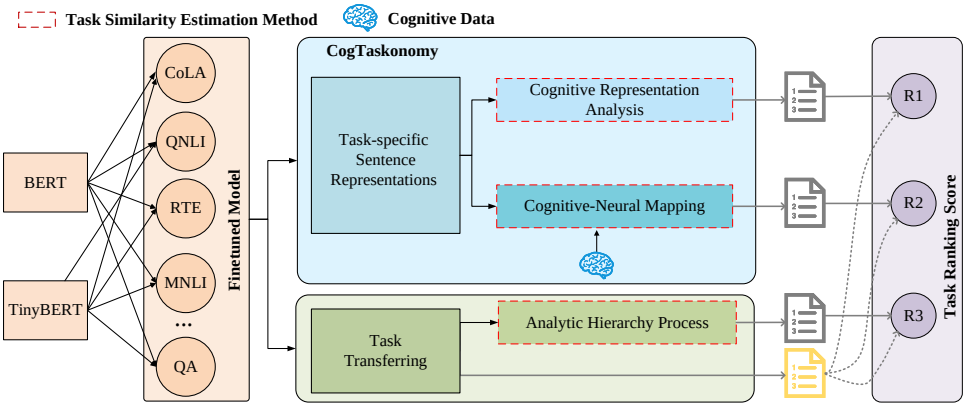


Figure 1: Illustration of CogTaskonomy. Pretrained language models are fine-tuned on downstream tasks to obtain task-specific sentence representations, which are then fed into CRA, CNM for estimating task similarity. We can obtain the most similar task for each target task by the corresponding task similarity estimation method and rank it according to the oracle task ranking obtained according to transfer learning performance. R is the task ranking score which is averaged over all target tasks.

sentations from cognitive processing data and use them to learn task taxonomy in NLP.

To examine this hypothesis, we propose CogTaskonomy, a **Cog**nitively Inspired **Task** Tax**onomy** framework, as illustrated in Figure 1, to learn a task structure for NLP tasks. CogTaskonomy consists of two main cognitively inspired components: Cognitive Representation Analytics (CRA) and Cognitive-Neural Mapping (CNM). CRA extracts task representations from NLP models and employs Representational Similarity Analysis (RSA) (Kriegeskorte et al., 2008), which is commonly used to measure the correlation between brain activity and computational model, to estimate NLP task similarity. CNM trains fully connected neural networks to build the mapping from sentence representations of pretrained models fine-tuned on specific tasks to fMRI signals recorded when human subjects read those sentences. It then uses mapping correlation coefficients as task representations to compute task similarity.

Both methods require sentence representations to compute task representations. We use pretrained language models fine-tuned on specific tasks, particularly BERT (Devlin et al., 2018) and TinyBERT (Jiao et al., 2020), to obtain sentence representations.

We compare the proposed CogTaskonomy against the Analytic Hierarchy Process (AHP) used in Taskonomy (Zamir et al., 2018). We guide TL across tasks with the learned task structure and evaluate the effectiveness of these methods by estimating TL performance from various source to target tasks.

Contributions Our main contributions include:

- We propose CogTaskonomy, a cognitively inspired framework to measure task similarity and to build the task taxonomy in NLP. This is the first attempt to study NLP task structures with cognitive processing data.
- We present two cognitively inspired methods, CRA and CNM, and compare them against AHP. Different from AHP, the two methods do not require $O(m^2)$ exhaustive pairwise transfer learning for task similarity estimation.
- We build a taxonomy tree for 12 NLP tasks, including sentiment analysis, question answering, natural language inference, semantic textual similarity, passage ranking, etc., to guide transfer learning across them.
- TL experiments and analyses validate the effectiveness of the proposed CogTaskonomy and find that CNM is able to learn stable task relations that are general to different underlying models.

2 Related Work

Our work is related to cognitively inspired NLP and a variety of learning formalisms that involve knowledge transfer across different tasks. We briefly review these topics within the scope of NLP and the constraint of space.

2.1 Cognitively Inspired NLP

Using NLP Representations for Brain Activity Prediction Since the pioneering work (Mitchell et al., 2008), connecting statistical NLP representations with cognition has attracted widespread at-

tention. Chang et al. (2009) explore adjective-noun composition in fMRI based on co-occurrence statistics. Huth et al. (2016) use distributed word representations to map fMRI data to activated brain regions, revealing a semantic map of how words are distributed in the human cerebral cortex. A great deal of research (Murphy et al., 2012; Anderson et al., 2016; Søgaard, 2016; Bulat et al., 2017) has been devoted to word decoding. Pereira et al. (2018) extend brain decoding to sentence stimuli, suggesting that neural network language models can be used to interpret sentences in a long-term context. Ren and Xiong (2022) investigate the relationship between linguistic features and cognitive processing signals by developing a unified attentional network to bridge them.

Augmenting NLP Models with Cognitive Processing Signals Recent years have witnessed that many efforts have been devoted to exploring cognitive processing signals (e.g., eye-tracking, EEG, fMRI) in neural NLP models. Muttenthaler et al. (2020) use cognitive data to regularize attention weights in NLP models. Hollenstein et al. (2019a) evaluate word embeddings using cognitive data. Toneva and Wehbe (2019) utilize fMRI scans to interpret and improve BERT. Many other works use cognitive processing signals to improve NLP models (Barrett and Søgaard, 2015; Bingel et al., 2016; Gauthier and Levy, 2019; Hollenstein and Zhang, 2019; Ren and Xiong, 2021), just to name a few.

2.2 Learning across Tasks

A very important trend in recent NLP is that models, algorithms, and solutions are not developed for only a single task, but for multiple tasks or across tasks (Devlin et al., 2018; Radford et al., 2018; McCann et al., 2018; Worsham and Kalita, 2020). Learning methods that are capable of handling a set of tasks simultaneously or sequentially, e.g., multi-task learning, transfer learning, meta learning, have attracted growing research interests in NLP. Beyond learning methods, yet another important dimension to this research trend is task relation learning, which is the topic of this work.[2]

Multi-task Learning is to jointly train all tasks of interests with task linkages, e.g., in the form of regularization or sharing parameters across tasks (Collobert et al., 2011). It is important in multi-task learning to find related tasks for target tasks as auxiliary tasks (Ruder, 2017).

Transfer Learning targets at transferring knowledge from a source task to a target task. According to the task and domain difference in the source and target, TL is divided into transductive TL (same task, different domain, a.k.a. domain adaptation), inductive TL (same domain, different task) and unsupervised TL (both different) (Eaton and desJardins, 2011; Ghifary et al., 2014; Wang et al., 2019; Yuan and Wen, 2021). If the source and target are dissimilar, negative transfer may hurt TL (Niu et al., 2020).

Meta Learning aims to gain experience over a set of related tasks for improving the learning algorithm itself (Hospedales et al., 2020). Existing meta learning methods implicitly assume that tasks are similar to each other, but it is often unclear how to quantify task similarities and their roles in learning (Venkitaraman and Wahlberg, 2020).

Lifelong Learning is to learn continuously and accumulate knowledge along a sequence of tasks and uses it for future learning (Chen and Liu, 2018). The system is tuned to be able to select the most related prior knowledge to bias the learning towards a new task favourably (Silver et al., 2013).

2.3 Learning Task Relations

As task relatedness is important for cross-task learning formalisms mentioned in Section 2.2, efforts have also been made to learn task relations. Crawshaw (2020) groups previous methods on task relationship learning into three categories. The first is task grouping or clustering, which divides a set of tasks into clusters so that tasks in the same cluster can be jointly trained (Bingel and Søgaard, 2017; Standley et al., 2019). The second is learning transfer relationships, which analyzes whether transfer between tasks is beneficial to learning, regardless of whether tasks are related or not (Zamir et al., 2018; Dwivedi and Roig, 2019; Song et al., 2019). The third is task embedding, which learns a specific representation space for tasks (James et al., 2018; Lan et al., 2019).

Our research can be considered as a mix of these categories. CNM learns cognition-based task representations while both CNM and CRA learn task relations aiming at transfer learning. Additionally, significantly different from previous studies, we

[2]Task taxonomy learned by our methods could be applicable to other learning formalisms beyond transfer learning. We leave this to our future work.

learn task structures from a cognitive perspective[3], which is expected to estimate task relatedness in a cognitively tuned space. As will be demonstrated below, our cognitively motivated methods incur a low computation cost and exhibit generalization across underlying models to some extend.

3 CogTaskonomy

Figure 1 illustrates the basic framework of CogTaskonomy. First, we obtain task-specific sentence representations of text stimuli from cognitive data by feeding them into fine-tuned or distilled pretrained language models on 12 downstream tasks (Section 3.1). Subsequently, task-specific representations are fed into two cognitively inspired components, cognitive representation analytics (Section 3.2) and cognitive-neural mapping (Section 3.3), for estimating task similarity and inducing task taxonomy.

3.1 Task-Specific Sentence Representations

Fine-tuning a pretrained language model for an end task is a widely used strategy for quickly and efficiently building a model for that task with limited labeled data. Zhou and Srikumar (2021) find that fine-tuning reconfigures underlying semantic space to adjust pretrained representations to downstream tasks. In view of this, we take sentence-level textual stimuli of cognitive data as input data for a specific fine-tuned model to obtain representations that contain information specific to that task.[4] Additionally, Cheng et al. (2020) suggest that knowledge distillation (KD) helps models to be more focused on task-relevant concepts. Therefore, without loss of generality, we use BERT and TinyBERT (performing KD) to obtain task-specific sentence representations.

BERT Following Devlin et al. (2018), we prepend a special classification token [CLS] to each input sentence in order to extract the contextualized representation of the corresponding sentence. Merchant et al. (2020) find that fine-tuning primarily affects top layers of BERT. Hence, we take the hidden state of the prepended token of each sequence in the last layer as the sentence representation.

TinyBERT TinyBERT (Jiao et al., 2020) performs knowledge distillation at both the pretraining and fine-tuning stage. By leveraging KD, TinyBERT learns to transfer knowledge encoded in the large teacher BERT (Devlin et al., 2018) to itself. As a result, TinyBERT can capture both general and task-specific knowledge. Similarly, we use the hidden state of [CLS] token in the last layer as the contextualized representation for a given sentence.

3.2 Cognitive Representation Analytics

With task-specific representations learned by feeding text stimuli of cognitive data into a fine-tuned model, we can estimate pairwise task similarity for any two tasks in a given task list $T = \{t_1, t_2, ..., t_m\}$. The first cognitively inspired method is the cognitive representation analytics that adapts a common method in computational neuroscience to our scenario. We first briefly introduce the common method, representational similarity analysis, and then elaborate the adaptation.

Representational Similarity Analysis is widely applied in cognitive neuroscience, which can not only realize cross-modal cognitive data comparison but also quantitatively relate brain activity measurements to computational models. It first calculates a representation dissimilarity matrix (RDM) of different modal data, and then estimates the correlation between RDMs. In this way, it successfully captures cross-modal data relationships (Kriegeskorte et al., 2008). RSA can be also applied for the comparison between computational models and cognitive data. The RDM of a computational model is obtained by comparing the dissimilarity of data representations obtained from the computational model in pairs. It is then compared with the RDM of brain activity measurements.[5]

We take all sentence representations R_i generated by a task-specific model PLM_i^{FT} (a pretrained language model (either BERT or TinyBERT) fine-tuned on the ith task) as the base to simulate cognitive representations required by RSA. For each pair of sentence representations $(R_{ij}, R_{ij'})$ for the jth and j'th sentence of the ith task, we compute a dissimilarity score in three metrics (e): Euclidean distance ($euclidean$), Canberra distance ($canberra$) and Pearson correlation coefficient (ρ). Among

[3] Dwivedi and Roig (2019) also use RSA to learn task taxonomy, in some way similar to our CRA. But they learn relations for visual tasks and use different correlation functions from our CRA.

[4] Sentence-level textual stimuli of cognitive data refer to natural textual stimuli, i.e., sentences presented to subjects for collecting cognitive processing signals.

[5] In our CRA, only RDMs from computational models are used. This is because we don't have cognitive data that are curated for specific NLP tasks. In our preliminary experiments, we have created pseudo cognitive data for different NLP tasks by predicting cognitive signals with a mapping model similar to that used in CNM. But it performs poorly.

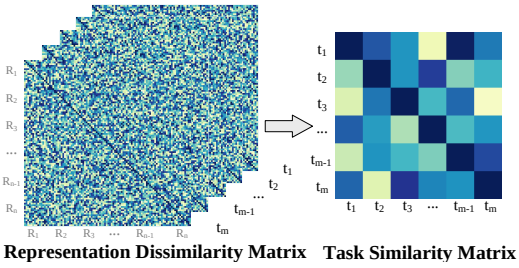


Figure 2: RDMs and task similarity matrix calculated by RSA from task-specific sentence representations.

them, the first two distance metrics can naturally represent dissimilarity (Dis), while the last ρ needs to be converted to $1-\rho$ to indicate dissimilarity, as follows:

$$\text{Dis}^i_{jj'} = \begin{cases} e(R_{ij}, R_{ij'}) & e \text{ is not } \rho \\ 1 - e(R_{ij}, R_{ij'}) & e \text{ is } \rho \end{cases} \quad (1)$$

RDM for the ith task consists of the dissimilarity scores of all sentence pairs. We formulate it as follows:

$$\begin{aligned} \text{RDM}_i = [&\text{Dis}^i_{12}, \text{Dis}^i_{13}, \ldots, \text{Dis}^i_{1n}, \ldots \\ &\text{Dis}^i_{jj'}, \ldots, \text{Dis}^i_{(n-1)n}], \quad j \neq j' \end{aligned} \quad (2)$$

where n is the number of sentences. RDMs computed in this way are then used for estimating similarity between NLP tasks. The pairwise similarity $\text{Sim}_{ii'}$ of the ith and i'th task is computed as follows:

$$\text{Sim}_{ii'} = \text{Similarity}(\text{RDM}_i^\intercal \cdot \text{RDM}_{i'}) \quad (3)$$

$\text{Similarity}(\cdot)$ is a similarity function, which can be Spearman rank correlation (r_s), ρ and cosine (cos, by default).

In summary, we calculate the similarity between each RDM pair and finally obtain a similarity matrix for a set of tasks, as shown in Figure 2.

3.3 Cognitive-Neural Mapping

The idea behind cognitive-neural mapping is to project sentence representations of NLP models fine-tuned in a specific task to cognitive signals (i.e., fMRI voxels in this paper) recorded when humans read those sentences with a neural network. The connections between the specific task and cognitive signals learned in this way could be transformed into cognitively inspired task representations for further task similarity estimation. The mapping can be considered as a way to isolate brain activity related to the specific task from fMRI cognitive signals. Particularly, for the ith task and sth subject, we use a fully connected 3-layer feed-forward neural network to project sentence representation R_{ij} specific to this task to fMRI $\boldsymbol{y}^{is}_j$ of the sth subject reading the jth sentence as follows:

$$\boldsymbol{y}^{is}_j = \boldsymbol{W}^i_2(\text{ReLU}(\boldsymbol{W}^i_1(R_{ij})) \quad (4)$$

To optimize the mapping model, we use the mean squared error (MSE) as loss function. 5-fold cross-validation is performed for each mapping model. Before training, grid search is conducted, and the optimal number of hidden layer units in the mapping network is obtained by three times of cross-validation on the verification set accounting for 20% training data.

Each mapping is run 5 times. We average models over all subjects and 5 runs and then evaluate mapping model performance in all voxels. Particularly, we compute the cognitively inspired task representation CogR_i for the ith task, which consists of the correlation coefficients on all voxels between predicted values and ground-truth values, defined as follows :

$$\begin{aligned} \text{CogR}_i = [&c(\hat{\boldsymbol{y}}^i_0, \boldsymbol{y}_0), \ldots, c(\hat{\boldsymbol{y}}^i_k, \boldsymbol{y}_k), \\ &\ldots, c(\hat{\boldsymbol{y}}^i_v, \boldsymbol{y}_v)], 0 \leq k \leq v \end{aligned} \quad (5)$$

where $\hat{\boldsymbol{y}}^i_k$ is a vector of all predicted values for the kth voxel from all input sentences by the mapping model tuned for the ith task, $\boldsymbol{y}_k$ is a vector of the ground-truth values for the kth voxel from all sentence-level signals of text stimuli in fMRI data, v is the number of voxels used, and $c(\cdot)$ is a function for comparing two input vectors. We instantiate c in two functions: the coefficient of determination (R^2) and ρ.[6]

We then use cosine similarity to calculate pairwise task similarity as follows:

$$\text{Sim}_{ii'} = \cos(\text{CogR}_i^\intercal \cdot \text{CogR}_{i'}) \quad (6)$$

4 Experiments

We conducted experiments with widely-used NLP benchmark datasets and cognitive data to evaluate the effectiveness of CogTaskonomy.

4.1 Cognitive Dataset

The brain fMRI dataset in our experiments is from Pereira et al. (2018), which is recorded on a whole-body 3-Tesla Siemens Trio scanner with a 32-channel head coil by showing 627 natural language sentences to 5 adult subjects.[7] Since voxels were

[6] R^2 is a statistical measure that examines how much a model is able to predict or explain an outcome, usually defined as the square of the correlation between predicted values and actual values. According to the results in Appendix A.1, we set R^2 as c in CNM by default.

[7] This dataset is publicly available at https://osf.io/crwz7/. The cognitive data of subjects who both partic-

randomly selected, Z-Score standardization was carried out for voxels obtained from different stimuli at each location on the basis of the original data set to avoid the influence of outliers. Subjects are asked to read each encyclopedic statement carefully, while the fMRI scanner records brain signals at this point. As a result, each fMRI scan covers multiple words at a time, subject to continuous stimulation. Each fMRI recording contains a number of voxels. We flattened 3d fMRI images into 1d vectors. v voxels were randomly selected, yielding matrices $I_s \in \mathbb{R}^{627\times v}$ for each subject s.

4.2 Tasks

We selected 8 NLP tasks from the GLUE benchmark (Wang et al., 2018), including CoLA, MNLI, MRPC, QNLI, QQP, RTE, SST-2, STS-B. These tasks are considered important for generalizable natural language understanding, exhibiting diversity in domains, dataset sizes, and difficulties (Wang et al., 2018). To cover the spectrum of NLP tasks as much as possible, we also included Extractive Question Answering (QA), Relation Extraction (RE), Named Entity Recognition (NER), and Passage Reranking (PR). The datasets of these four tasks are SQuAD 2.0 (Rajpurkar et al., 2018), Semeval-2010 task 8 (Hendrickx et al., 2010), CoNLL 2003 (Sang and Meulder, 2003), MS MARCO (Nguyen et al., 2016; Craswell et al., 2020), respectively.

4.3 Baselines and Settings

We mainly used two methods as our baselines, including Direct Similarity Estimation (DSE), Analytic Hierarchy Process (AHP) (Zamir et al., 2018). Detailed experimental settings are shown in Appendix A.2.

Direct Similarity Estimation (DSE) A straightforward way to estimate pairwise task similarity is to calculate sentence-level similarities based on task-specific sentence representations and then average them. Concretely, let R_{ij} be the task-specific representation for the jth sentence in the ith task. The task similarity $\text{Sim}_{ii'}$ for a task pair (i, i') is computed as follows:

$$R_{ij} = \text{PLM}_i^{\text{FT}}(\boldsymbol{x}_j) \tag{7}$$

$$\text{Sim}_{ii'} = \frac{\sum_j \text{Similarity}(R_{ij}^{\intercal} \cdot R_{i'j})}{n} \tag{8}$$

ipated in experiments 2 and 3 were chosen in this paper.

where PLM_i^{FT} is the pretrained language model fine-tuned on the ith task. PLM can be instantiated as TinyBERT or BERT.

Analytic Hierarchy Process (AHP) The main idea is to construct a matrix W_t for each target task t, where the element at (i, i') in the matrix shows how many times the ith source task is better than i'th source task in terms of the transferability to the target task on a held-out set. The principal eigenvector of W_t is then taken as the task representation for the corresponding task, and all task representations are stacked up to obtain an affinity matrix.[8] The affinity matrix is then viewed as the task similarity matrix.

4.4 Evaluation Metric

Task Transferring To assess the similarity between tasks, all models fine-tuned on non-target tasks will be used as source models, and continue to be fine-tuned in the same way to transfer on the target task. In task transferring, all parameters of source models are fine-tuned (i.e., not fixed). We used the same learning rate and a number of training steps for all task transferring. This allows a fair comparison between different source tasks.

Oracle Task Ranking The final similarity ranking of source tasks to a given target task is based on the results obtained from the task transferring experiments. Generally speaking, the better the source-to-target transfer performance is, the more similar the two tasks are, since the essence of TL is to apply knowledge learned in the source task to the target task. Based on this concept, we rank tasks in terms of transfer learning performance, for more details please see Appendix A.3.

Task Ranking Score Based on similarity results computed by each task estimation method, we can obtain the most similar task for each target task. We then check the ranking position of the most similar task in the oracle task ranking. We average ranking positions of all target tasks as the final task ranking score for the corresponding task estimation method. Note that we exclude the transfer to the target task itself in computing task ranking scores.[9]

[8] For more details about AHP, please refer to (Zamir et al., 2018). Since the test sets of our NLP tasks are not publicly available, we obtain AHP results based on the validation set of each task except the NER task of which the test set is publicly available. In all experiments, the hyper-parameters are the same for all tasks.

[9] Generally, the lower the task ranking score, the better the task similarity estimation method. A perfect estimation

Method	TRS	
	TinyBERT	BERT
DSE	6.2	4.8
CRA	3.5	4.4
CNM	4.2	4.6
AHP	**1.4**	**2.5**
CRA+CNM	2.8	4.3
Random	6.0	

Table 1: Task ranking scores (TRS) for different task similarity estimation methods. cos was used as the similarity function for DSE. ρ was used in Eq.(1) and cos was used in Eq.(3) for CRA. R^2 was used as c in CNM.

Sent. Diss.	Task Sim.	TRS	
		BERT	TinyBERT
euclidean	r^s	5.6	4.5
	cos	5.0	2.6
	ρ	5.6	3.5
canberra	r^s	5.0	5.2
	cos	5.8	**2.2**
	ρ	5.5	5.1
ρ	r^s	5.5	6.7
	cos	**4.4**	3.5
	ρ	5.1	4.3

Table 2: Task ranking scores of CRA with different combinations of sentence dissimilarity an task similarity measurements.

4.5 Main Results

Task ranking scores (using the ranking of task transferring as the oracle ranking) of different task similarity estimation methods are shown in Table 1. From these results, we have the following observations:

- Both CRA and CNM are better than random ranking and DSE, suggesting that cognitively inspired task similarity estimation is able to capture relations of NLP tasks.
- When TinyBERT is used, DSE is even worse than random ranking. This suggests that simply using task-specific sentence representations cannot well detect task relations and distinguish different tasks.
- TinyBERT performs better than BERT across three task estimation methods (i.e., CRA, CNM and AHP) although the number of parameters in the former is only half of that in the latter. We conjecture that TinyBERT uses knowledge distillation, making sentence representations more relevant to individual tasks and hence resulting in better task similarity estimation.
- We can also combine CRA and CNM (CRA+CNM) by averaging task similarity scores estimated by them. Such combination is better than both methods alone.

Although AHP is better than our methods, it directly uses the results of transfer learning to measure similarities between different tasks, which is very time-consuming. if we have m tasks, we have to perform $O(m^2)$ transfer learning to obtain the task similarity matrix across all task pairs. In contrast, our methods do not require any costly transfer learning between tasks. It is hence easier to perform and able to guide transfer learning across tasks. We further evaluated the actual transfer learning performance of each target task from the most similar source task according to different task similarity estimation methods. Results are shown in Appendix A.4, which further validate the effectiveness of our methods and show that CRA+CNM is very close to that of AHP. In later experiments and analyses, we will show more advantages of our methods over AHP.

4.6 Evaluating CRA with Different Dissimilarity/Similarity Measurement Combinations

CRA adopts RSA to transform the dissimilarity of task-specific sentence representations into the similarity of tasks. We have different options for dissimilarity measurement (e.g., $euclidean$,$canberra$) in sentences and for similarity measurement (e.g., cos,ρ) in tasks. Hence we want to know the impact of the combinations of different measurements in sentence dissimilarity and task similarity on final performance. Results are provided in Table 2. Again, we have several interesting observations. First, with different combinations of these measurements, our CRA significantly outperforms random ranking in almost all cases. This suggests that RSA is able to be adapted to NLP task structure detection. Second, in comparison to the combination of ρ and r^s in the original RSA (Kriegeskorte et al., 2008), in our case, the combination of ρ and cos is better than other combinations in the majority of cases. Third, TinyBERT is more robust to these

similarity method would yield a task ranking score of 1 on each target task. A random method would yield a ranking score of $0.5(1 + 11) = 6$ in our experiments theoretically. We have also conducted random sampling 5000 times on TinyBERT and BERT, and obtained mean task ranking scores of 6.05 and 6.04 respectively. Hence, we take the 6 as the task ranking score for random ranking.

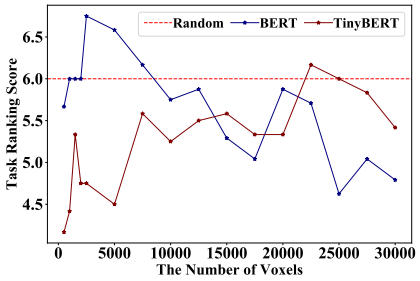


Figure 3: Task ranking scores of CNM with TinyBERT and BERT predicting different numbers of voxels.

different combinations than BERT.

4.7 Evaluating CNM with Different PLMs and Numbers of Voxels

Since CNM bridges pretrained language models on the input side and voxels in fMRI images on the output side, we further evaluated CNM by varying the selection of PLMs (either BERT or TinyBERT) and the numbers of voxels. Results are displayed in Figure 3. It is interesting to find that with a small number of cognitive signals (voxels), TinyBERT for CNM can achieve a good task ranking score. By contrast, without sufficient cognitive signals, BERT for CNM fails in task similarity estimation, obtaining a task ranking score worse than random ranking. This is consistent with our previous finding in the main results that TinyBERT (with KD) captures more task-relevant knowledge than BERT for task relation detection.

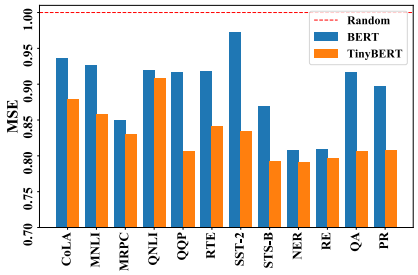
5 Analysis

5.1 CNM: Voxel Prediction Evaluation

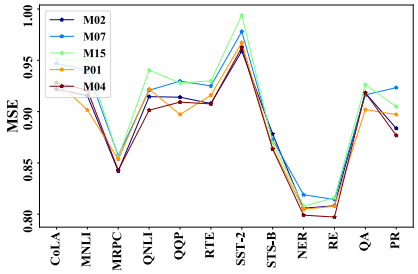
We conducted experiments to take a deep look into the feed-forward neural mapping model in CNM. The number of voxels was set to 30K.

Pretrained Language Models We compared the prediction performance (measured by MSE between predicted results and ground-truth voxels) across different tasks using BERT vs. TinyBERT as the pretrained language model to obtain task-specific sentence representations. Results are shown in Figure 4(a). We can clearly see that both BERT and TinyBERT are better than the random baseline across all tasks. And TinyBERT is better than BERT on all tasks, which resonates with the main results shown in Section 4.5.

Subjects We analyzed prediction performance across different subjects, as shown in Figure 4(b). Although the prediction performance varies across different tasks, the shapes of the prediction performance curve over 12 tasks for different subjects are similar to each other, indicating that similar brain activities are activated for these tasks across different subjects.



(a) MSEs (averaged over 5 subjects and 30K voxels) for different tasks with BERT vs. TinyBERT being used as the pretrained language model.



(b) MSEs (averaged over 30K voxels) for different tasks across different subjects. BERT is used as the pretrained language model.

Figure 4: CNM voxel prediction results (i.e., the mean square errors between predicted results by CNM and ground-truth voxels). Y-coordinate is the ratio of the MSE value of BERT to the MSE of the random prediction baseline.

5.2 Analysis on the Generality of Task Similarity Estimation to Underlying PLMs

Models underlying our cross-task transfer learning are different pretrained language models, which is a widely acknowledged practice for transfer learning in NLP. We therefore want to investigate how general our task similarity estimation methods (e.g., CNM, CRA, AHP) are to the underlying models. This is important as we want to find a task taxonomy method that is not sensitive to underlying models. That is, the learned task taxonomy can be used to guide transfer learning for any model. For this, we first computed the Pearson correlation coefficient (ρ) and the Spearman rank correlation (r_s) between task similarities obtained with TinyBERT and those with BERT using the same similarity estimation method. The correlation coefficients

Method	TB→B			B→TB		
k	3	4	5	3	4	5
CRA	0.39	0.40	0.42	0.61	0.54	0.58
CNM	**0.64**	**0.58**	**0.62**	**0.75**	**0.79**	**0.73**
AHP	0.53	0.52	0.52	0.69	0.65	0.67

Table 3: Probabilities that transferability learned with TinyBERT (TB) can be used for BERT (B) or vice versa.

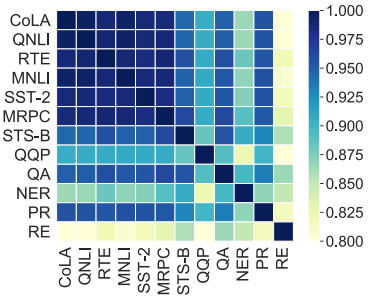
between BERT-based and TinyBERT-based task similarity matrices obtained by the CRA, CNM and AHP are ($\rho = 0.23$, $r^s = 0.11$), ($\rho = 0.85$, $r^s = 0.76$) and ($\rho = 0.36$, $r^s = 0.34$) respectively. Both AHP and CRA show pool correlations between task similarity matrices using BERT and TinyBERT. On the contrary, CNM is very robust to the variations of underlying models. We speculate that both CRA and AHP capture task relations specific to underlying models while CNM could remove such bias by building the task taxonomy based on the cognitive data. In other words, CNM is able to detect model-agnostic task relations, yet another desirable advantage over AHP with exhaustive computation cost.

To further examine this hypothesis, we used the task ranking estimated with another underlying PLM x to guide transfer learning with an underlying PLM y. In our work, this would be using TinyBERT to guide BERT (TB → B) or vice versa. For each target task, we used the top k source tasks according to the task ranking with the guiding PLM x for transfer learning with the PLM y. The results were compared to the actual performance of transfer learning to the target task from the top 6 source tasks according to the task ranking with the PLM y itself. The probability of the top k source tasks occurring in the real top 6 tasks shows how much transferability learned with the PLM x can be used for the PLM y. Results are shown in Table 3, which again suggests the superiority of CNM over AHP.

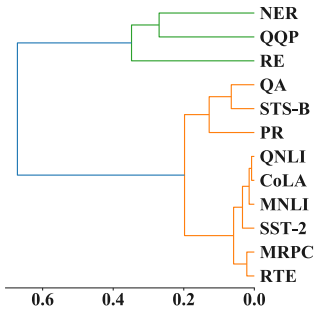
We further analyzed the generality of CNM to different subjects of cognitive data used in CNM, which can be found in Appendix A.5. The experimental results show that the CNM is also robust to different subjects.

5.3 Taxonomy Tree of 12 NLP Tasks

We visualize all pairwise task similarities for 12 tasks learned by CNM (averaged over 5 subjects) as a heatmap, shown in Figure 5(a). It is clear to see from the heatmap that 6 GLUE tasks (i.e., CoLA, QNLI, RTE, MNLI, SST-2, and MRPC)



(a) Task similarity matrix



(b) Taxonomy tree

Figure 5: Task similarity learned by CNM: (a) Task similarity matrix learned by CNM. (b) Taxonomy tree for the 12 tasks learned by CNM.

form a cluster. These tasks are all related to sentence understanding. We further perform hierarchical clustering over the 12 tasks according to their similarities to create a taxonomy tree, which is illustrated in Figure 5(b).

6 Conclusions

In this paper, we have presented a cognitively inspired framework, termed CogTaxonomy, to learn relation and structure for NLP tasks. Experiments demonstrate that the task taxonomy detected by CogTaxonomy can be used to guide transfer learning across 12 different NLP tasks. Both CRA and CNM, the two essential components of CogTaxonomy, do not require exhaustive transfer learning across all source-target task pairs. The former is robust to different combinations of dissimilarity/similarity measurements. The latter resorts to cognitive signals to learn model-agnostic task relations.

Acknowledgments

The present research was supported by Zhejiang Lab (No. 2022KH0AB01) and the Natural Science Foundation of Tianjin (No. 19JCZDJC31400). We would like to thank the anonymous reviewers for their insightful comments.

References

Andrew J. Anderson, Benjamin Zinszer, and Rajeev D. S. Raizada. 2016. Representational similarity encoding for fmri: Pattern-based synthesis to predict brain activity using stimulus-model-similarities. *NeuroImage*, 128:44–53.

Maria Barrett and Anders Søgaard. 2015. Reading behavior predicts syntactic categories. In *Proceedings of the 19th Conference on Computational Natural Language Learning, CoNLL 2015, Beijing, China, July 30-31, 2015*, pages 345–349. ACL.

Joachim Bingel, Maria Barrett, and Anders Søgaard. 2016. Extracting token-level signals of syntactic processing from fmri - with an application to pos induction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Joachim Bingel and Anders Søgaard. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 164–169. Association for Computational Linguistics.

Luana Bulat, Stephen Clark, and Ekaterina Shutova. 2017. Speaking, seeing, understanding: Correlating semantic models with conceptual representation in the brain. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1081–1091. Association for Computational Linguistics.

Kai-min Kevin Chang, Vladimir Cherkassky, Tom M. Mitchell, and Marcel Adam Just. 2009. Quantitative modeling of the neural representation of adjective-noun phrases to account for fmri activation. In *ACL 2009, Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP, 2-7 August 2009, Singapore*, pages 638–646. The Association for Computer Linguistics.

Zhiyuan Chen and Bing Liu. 2018. *Lifelong Machine Learning, Second Edition*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Xu Cheng, Zhefan Rao, Yilan Chen, and Quanshi Zhang. 2020. Explaining knowledge distillation by quantifying the knowledge. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12922–12932. Computer Vision Foundation / IEEE.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1304–1319. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural language processing (almost) from scratch. *CoRR*, abs/1103.0398.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. *CoRR*, abs/2003.07820.

Michael Crawshaw. 2020. Multi-task learning with deep neural networks: A survey. *CoRR*, abs/2009.09796.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Kshitij Dwivedi and Gemma Roig. 2019. Representation similarity analysis for efficient task taxonomy & transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 12387–12396. Computer Vision Foundation / IEEE.

Eric Eaton and Marie desJardins. 2011. Selective transfer between learning tasks using task-based boosting. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011*. AAAI Press.

Jon Gauthier and Roger Levy. 2019. Linking artificial and human neural representations of language. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 529–539. Association for Computational Linguistics.

Muhammad Ghifary, W. Bastiaan Kleijn, and Mengjie Zhang. 2014. Domain adaptive neural networks for object recognition. *CoRR*, abs/1409.6041.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O. K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 344–354. Association for Computational Linguistics.

John Hale, Chris Dyer, Adhiguna Kuncoro, and Jonathan Brennan. 2018. Finding syntax in human encephalography with beam search. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*,

pages 2727–2736. Association for Computational Linguistics.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval@ACL 2010, Uppsala University, Uppsala, Sweden, July 15-16, 2010*, pages 33–38. The Association for Computer Linguistics.

Nora Hollenstein, Maria Barrett, Marius Troendle, Francesco Bigiolli, Nicolas Langer, and Ce Zhang. 2019a. Advancing NLP with cognitive language processing signals. *CoRR*, abs/1904.02682.

Nora Hollenstein, Antonio de la Torre, Nicolas Langer, and Ce Zhang. 2019b. Cognival: A framework for cognitive word embedding evaluation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 538–549. Association for Computational Linguistics.

Nora Hollenstein and Ce Zhang. 2019. Entity recognition at first sight: Improving NER with eye movement information. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1–10. Association for Computational Linguistics.

Timothy M. Hospedales, Antreas Antoniou, Paul Micaelli, and Amos J. Storkey. 2020. Meta-learning in neural networks: A survey. *CoRR*, abs/2004.05439.

Alexander G. Huth, Wendy A. de Heer, Thomas L. Griffiths, Frédéric E. Theunissen, and Jack L. Gallant. 2016. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nat.*, 532(7600):453–458.

Stephen James, Michael Bloesch, and Andrew J. Davison. 2018. Task-embedded control networks for few-shot imitation learning. In *2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings*, volume 87 of *Proceedings of Machine Learning Research*, pages 783–795. PMLR.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Wei-Jen Ko, Ahmed El-Kishky, Adithya Renduchintala, Vishrav Chaudhary, Naman Goyal, Francisco Guzmán, Pascale Fung, Philipp Koehn, and Mona T. Diab. 2021. Adapting high-resource NMT models to translate low-resource related languages without parallel data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 802–812. Association for Computational Linguistics.

N. Kriegeskorte, Marieke Mur, and P. Bandettini. 2008. Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, 2.

Lin Lan, Zhenguo Li, Xiaohong Guan, and Pinghui Wang. 2019. Meta reinforcement learning with task embedding and shared policy. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 2794–2800. ijcai.org.

Cheng Li and Ye Tian. 2020. Downstream model design of pre-trained language model for relation extraction task. *CoRR*, abs/2004.03786.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *CoRR*, abs/1806.08730.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@EMNLP 2020, Online, November 2020*, pages 33–44. Association for Computational Linguistics.

Tom Michael Mitchell, S. Shinkareva, Andrew Carlson, K. Chang, Vicente L. Malave, R. Mason, and M. Just. 2008. Predicting human brain activity associated with the meanings of nouns. *Science*, 320:1191 – 1195.

Brian Murphy, Partha P. Talukdar, and Tom M. Mitchell. 2012. Selecting corpus-semantic models for neurolinguistic decoding. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics, *SEM 2012, June 7-8, 2012, Montréal, Canada*, pages 114–123. Association for Computational Linguistics.

Lukas Muttenthaler, Nora Hollenstein, and Maria Barrett. 2020. Human brain activity for machine attention. *CoRR*, abs/2006.05113.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located*

with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Shuteng Niu, Yongxin Liu, Jian Wang, and Houbing Song. 2020. A decade survey of transfer learning (2010-2020). *IEEE Trans. Artif. Intell.*, 1(2):151–166.

Francisco Pereira, Bin Lou, Brianna Pritchett, Samuel Ritter, Samuel J Gershman, Nancy Kanwisher, Matthew Botvinick, and Evelina Fedorenko. 2018. Toward a universal decoder of linguistic meaning from brain activation. *Nature communications*, 9(1):963.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 784–789. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Yuqi Ren and Deyi Xiong. 2021. CogAlign: Learning to align textual neural representations to cognitive language processing signals. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3758–3769, Online. Association for Computational Linguistics.

Yuqi Ren and Deyi Xiong. 2022. Bridging between cognitive processing signals and linguistic features via a unified attentional network. In *Thirty-sixth AAAI Conference on Artificial Intelligence, AAAI2022*. AAAI Press.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098.

R.W. Saaty. 1987. The analytic hierarchy process—what it is and how it is used. *Mathematical Modelling*, 9(3):161–176.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL.

Daniel L. Silver, Qiang Yang, and Lianghao Li. 2013. Lifelong machine learning systems: Beyond learning algorithms. In *Lifelong Machine Learning, Papers from the 2013 AAAI Spring Symposium, Palo Alto, California, USA, March 25-27, 2013*, volume SS-13-05 of *AAAI Technical Report*. AAAI.

Anders Søgaard. 2016. Evaluating word embeddings with fmri and eye-tracking. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, RepEval@ACL 2016, Berlin, Germany, August 2016*, pages 116–121. Association for Computational Linguistics.

Jie Song, Yixin Chen, Xinchao Wang, Chengchao Shen, and Mingli Song. 2019. Deep model transferability from attribution maps. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 6179–6189.

Trevor Standley, Amir Roshan Zamir, Dawn Chen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. 2019. Which tasks should be learned together in multi-task learning? *CoRR*, abs/1905.07553.

Mariya Toneva and Leila Wehbe. 2019. Interpreting and improving natural-language processing (in machines) with natural language-processing (in the brain). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14928–14938.

Arun Venkitaraman and Bo Wahlberg. 2020. Task-similarity aware meta-learning through nonparametric kernel regression. *CoRR*, abs/2006.07212.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Boyu Wang, Jorge A. Mendez, Mingbo Cai, and Eric Eaton. 2019. Transfer learning via minimizing the performance gap between domains. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 10644–10654.

Joseph Worsham and Jugal Kalita. 2020. Multi-task learning for natural language processing in the 2020s: Where are we going? *Pattern Recognit. Lett.*, 136:120–126.

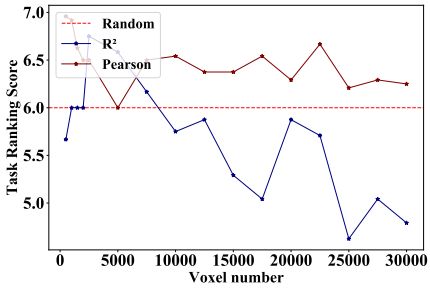
Tiezheng Yu, Zihan Liu, and Pascale Fung. 2021. Adaptsum: Towards low-resource domain adaptation for abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, pages 5892–5904. Association for Computational Linguistics.

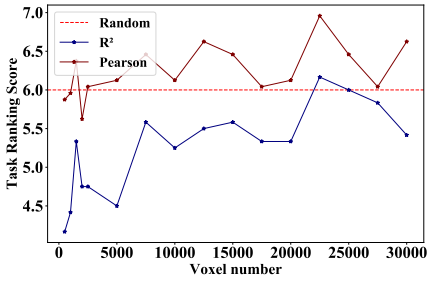
Zhe Yuan and Yimin Wen. 2021. A new semi-supervised inductive transfer learning framework: Co-transfer. *CoRR*, abs/2108.07930.

Amir Roshan Zamir, Alexander Sax, William B. Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling task transfer learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3712–3722. Computer Vision Foundation / IEEE Computer Society.

Yichu Zhou and Vivek Srikumar. 2021. A closer look at how fine-tuning changes BERT. *CoRR*, abs/2106.14282.



(a) BERT



(b) TinyBERT

Figure 6: Task ranking scores with different correlation coefficients in Cognitive-Neural Mapping based on BERT (a) and TinyBERT (b).

A Appendix

A.1 Correlation Selection in Cognitive-Neural Mapping

We have different options for the calculation of CogR (e.g., R^2 and ρ). Therefore, we computed task ranking scores with different options for CNM with BERT and TinyBERT. Results are shown in Figure 6. We find that R^2 is better than ρ in almost all cases.

A.2 Experimental Settings

Fine-tuning and Transferring For most tasks, we fine-tuned BERT to obtain task-specific representations, with the exception of RE and PR. REDN (Li and Tian, 2020) was used for RE tasks, and Cross-Encoder (Reimers and Gurevych, 2019) was used for PR. Table 4 shows the hyper-parameters and configuration for task training in our experiments.

For source-to-target task transfer learning, all source models were fine-tuned on the target task dataset with the same settings as the target model.

Knowledge Distillation Since TinyBERT has officially released models distilled on GLUE, we directly used them on our 8 GLUE tasks. For the PR task, we used the open-source model (Reimers and Gurevych, 2019) with the same TinyBERT architecture directly. For NER, QA, and RE tasks, we adopted the above fine-tuned models as the teacher model and used the open-source General TinyBERT for task-specific distillation, following the recommended practice of TinyBERT (Jiao et al., 2020). The number of epochs in the task-specific distillation phase is shown in Table 5, and the settings of other parameters are consistent with fine-tuning. In CoNLL 2003 (Sang and Meulder, 2003), we also carried out data augmentation according to the method proposed by TinyBERT (Jiao et al., 2020), while no data augmentation was performed on other datasets.

Task	epoch	batch size	lr	optimizer
GLUE	3	32	2e-5	AdamW
NER	3	32	2e-5	AdamW
QA	3	16	3e-5	AdamW
PR	1	8	2e-5	AdamW
RE	50	64	3e-5	AdamW

Table 4: Hyper-parameter settings for fine-tuning and transfer learning. lr: learning rate.

Task	ID	PD
NER	5	3
QA	5	3
RE	25	25

Table 5: The number of epochs for NER, RE, and PR tasks in the task-specific knowledge distillation phase. ID/PD: intermediate/prediction layer knowledge distillation.

A.3 Oracle Task Ranking

Evaluation Metrics for 12 Tasks Among the 8 GLUE tasks, Matthews correlation was used in the CoLA task, Spearman correlation coefficient was used in the STS-B task. For the passage reranking task, NDCG@10 was used. All other tasks used F_1 score as the evaluation metric.

Pairwise Transfer Learning Results and Oracle Ranking We obtain pair-wise transfer learning results based on the validation set of each task except the NER task of which the test set is publicly available. The results with BERT and TinyBERT are shown in the Table 6 and Table 7 respectively. Sorted by transfer performance, the oracle ranking of each source task to a target task is marked in parentheses.

Source Task	Target Task											
	CoLA	QNLI	RTE	MNLI	SST-2	MRPC	STS-B	QQP	NER	RE	QA	PR
CoLA	54.96	90.81(4.5)	54.87(10)	84.05(2)	92.43(2)	76.47(7)	40.95(9)	90.17(4)	90.58(1)	89.62(7)	75.92(6)	67.79(1)
QNLI	43.3(9)	90.98	66.06(3)	83.74(8)	92.09(6.5)	81.62(2)	66.62(1)	90.01(5)	90.34(3.5)	89.09(11)	76.37(3)	61.05(10)
RTE	51.81(1)	90.57(6)	61.01	83.92(3.5)	91.97(8.5)	77.94(6)	45.5(6)	90.19(3)	90.28(8)	90.16(3)	76.18(5)	67.01(4.5)
MNLI	48.15(6)	90.28(8)	74.37(1)	83.94	93.12(1)	82.84(1)	49.22(5)	89.96(7)	90.33(5.5)	89.17(9)	76.9(2)	67.01(4.5)
SST-2	49.12(4)	90.81(4.5)	53.07(11)	84.06(1)	92.55	73.04(9.5)	54.36(2)	90.2(2)	90.06(11)	90.05(5)	75.5(9)	67.04(3)
MRPC	50.46(3)	90.54(7)	62.82(5.5)	83.89(6)	92.2(4.5)	75.25	24.4(11)	89.73(10)	90.22(9)	90.53(2)	75.79(8)	66.41(6)
STS-B	47.2(7)	90.02(10)	64.26(4)	83.68(10)	92.32(3)	79.41(4.5)	49.84	89.41(11)	90.29(7)	90.95(1)	72.74(11)	5.96(11)
QQP	50.51(2)	90.99(2)	62.82(5.5)	83.92(3.5)	92.09(6.5)	80.39(3)	42.43(7)	91.04	90.34(3.5)	90.08(4)	75.85(7)	66.22(7)
NER	48.29(5)	88.65(11)	61.37(8)	83.72(9)	92.2(4.5)	73.04(9.5)	41.26(8)	89.97(6)	90.75	89.57(8)	75.16(10)	62.3(9)
RE	40.65(11)	90.23(9)	56.68(9)	83.91(5)	91.97(8.5)	70.83(11)	30.73(10)	89.92(8)	90.08(10)	89.64	76.19(4)	67.23(2)
QA	43.87(8)	90.98(3)	67.15(2)	83.2(11)	90.6(11)	76.23(8)	50.05(4)	89.78(9)	90.33(5.5)	89.14(10)	74.9	64.66(8)
PR	43.27(10)	91.29(1)	62.45(7)	83.8(7)	91.4(10)	79.41(4.5)	53.02(3)	90.2(1)	90.36(2)	89.77(6)	77.39(1)	65.69

Table 6: Pairwise transfer learning results based on BERT. The oracle ranking of each source task to a target task is labeled in parentheses.

Source Task	Target Task											
	CoLA	QNLI	RTE	MNLI	SST-2	MRPC	STS-B	QQP	NER	RE	QA	PR
CoLA	50.68	82.74(10)	51.26(10)	78.58(9)	90.83(4)	70.1(8)	-14.14(12)	87.66(9)	85.79(9)	89.36(1)	62.84(10)	65.57(10)
QNLI	35.61(1)	91.36	63.18(3)	82.56(1)	91.74(2)	84.07(1)	65.31(2)	89.34(2)	88.37(2)	88.62(2)	74.24(1)	67.92(4)
RTE	20.02(6)	88.67(5)	66.79	81.56(4)	90.14(6)	78.43(4)	60.47(3)	88.66(5)	88.53(1)	87.72(3)	68.15(6)	66.64(7)
MNLI	22.52(5)	90.13(1)	69.68(1)	84.39	91.86(1)	81.86(2)	18.71(6)	89.44(1)	86.64(7)	87.51(5)	72.62(2)	68.72(3)
SST-2	0.0(10)	82.96(9)	53.43(7)	78.3(10)	91.86	68.38(11)	6.73(7)	87.37(10)	84.33(11)	85.15(10)	64.42(9)	65.93(9)
MRPC	28.36(3)	87.2(7)	60.65(4)	81.26(5)	89.91(7)	86.03	-3.4(11)	88.85(4)	88.08(3)	86.46(8)	66.73(7)	69.29(1)
STS-B	28.93(2)	89.04(3)	64.62(2)	81.67(3)	91.17(3)	77.94(6)	75.49	89.19(3)	87.85(4)	86.73(6)	70.16(4)	66.4(8)
QQP	25.73(4)	88.1(6)	58.48(5.5)	81.14(6)	89.33(8)	78.92(3)	42.97(4)	91.06	86.04(8)	86.72(7)	69.39(5)	67.29(5.5)
NER	14.69(8)	80.32(11)	51.99(9)	74.46(11)	88.07(10)	69.36(10)	2.41(8)	84.76(11)	88.44	84.37(11)	58.68(11)	65.26(11)
RE	18.44(7)	86.58(8)	53.07(8)	80.64(7)	90.48(5)	69.85(9)	-2.62(10)	87.95(7)	86.99(5)	87.62	66.61(8)	67.29(5.5)
QA	0.0(10)	89.91(2)	58.48(5.5)	79.86(8)	87.84(11)	77.45(7)	33.58(5)	87.89(8)	84.98(10)	87.54(4)	71.98	68.88(2)
PR	0.0(10)	88.96(4)	46.93(11)	81.7(2)	89.11(9)	78.19(5)	65.98(1)	88.43(6)	86.66(6)	86.08(9)	70.78(3)	66.9

Table 7: Pairwise transfer learning results based on TinyBERT.

A.4 Actual Transfer Learning Performance from the Top 1 Source Task Selected by Different Task Similarity Estimation Methods to Each Target Task

Transfer Learning with Same Underlying PLMs We use different task similarity estimation methods to find the most similar source task for each target task and obtain the transfer learning performance from the most similar source task to the target task. Both the task similarity estimation and transfer learning use the same underlying PLM. For each task similarity estimation method, average performance over all target tasks are reported in Table 8. It can be seen that the CRA, CNM, and CRA+CNM methods show good performance. Significantly, in terms of average target task performance, CRA+CNM is very close to AHP that requires exhaustive transfer learning across all source-target task pairs.

Transfer Learning with Different Underlying PLMs This time the underlying PLMs for task similarity estimation and transfer learning are different from each other. Results are displayed in Table 9. Similarly, CRA+CNM achieves very competitive results to AHP in TinyBERT $\rightarrow$ BERT and even better results than AHP in BERT $\rightarrow$ TinyBERT.

In the two tables, we calculate the average transfer learning performance over 12 tasks shown in the last column of the two tables for easy compar-

PLM	Method	Target Task												AVG
		CoLA	QNLI	RTE	MNLI	SST-2	MRPC	STS-B	QQP	QA	NER	PR	RE	
B	DSE	49.12	90.57	66.06	83.74	92.43	81.62	45.5	**90.2**	76.18	90.28	**67.79**	89.14	76.89
	CRA	**51.81**	90.57	66.06	**83.92**	91.97	77.94	45.5	90.19	76.18	90.34	67.01	90.16	76.80
	CNM	**51.81**	**91.29**	54.87	83.74	92.43	81.62	**66.62**	90.19	76.18	90.08	67.01	89.57	**77.95**
	CRA+CNM	**51.81**	90.57	66.06	**83.92**	91.97	77.94	45.5	90.19	76.18	90.34	67.01	90.16	76.80
	AHP	**51.81**	**91.29**	**74.37**	83.74	**93.12**	**82.84**	41.26	**90.2**	**77.39**	**90.58**	67.01	**90.95**	77.88
	Random	46.97	90.45	62.23	83.8	92.04	77.49	44.75	89.95	75.84	90.3	59.21	89.81	75.24
TB	DSE	20.02	88.67	58.48	78.58	90.14	77.94	-3.4	88.66	66.61	87.85	66.64	87.72	67.33
	CRA	18.44	89.04	63.18	**82.56**	91.74	77.94	65.31	88.66	**74.24**	**88.37**	66.4	88.62	74.54
	CNM	**35.61**	82.74	63.18	**82.56**	91.74	78.43	-3.4	**89.44**	66.73	87.85	68.72	87.54	69.26
	CRA+CNM	**35.61**	89.04	63.18	**82.56**	91.74	**84.07**	65.31	88.66	**74.24**	87.85	67.92	86.73	76.41
	AHP	25.73	**90.13**	**69.68**	**82.56**	**91.86**	**84.07**	**65.98**	89.34	**74.24**	**88.37**	**69.29**	**89.36**	**76.72**
	Random	17.39	86.65	57.24	80.08	89.99	75.79	25.06	88.19	67.75	86.76	67.18	86.98	69.09

Table 8: Actual target task performance when both task similarity estimation and transfer learning uses the same underlying PLM. B/TB: BERT/TinyBERT.

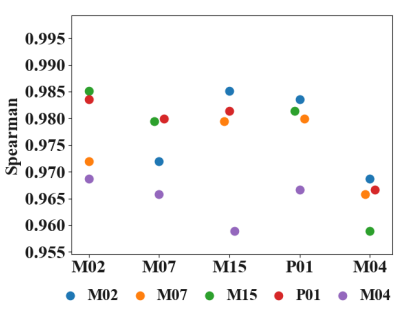
Type	Method	Target Task												AVG
		CoLA	QNLI	RTE	MNLI	SST-2	MRPC	STS-B	QQP	QA	NER	PR	RE	
TB → B	DSE	**51.81**	90.57	62.82	**84.05**	91.97	79.41	24.4	**90.19**	76.19	90.29	**67.01**	90.16	74.91
	CRA	40.65	90.02	66.06	83.74	92.09	79.41	**66.62**	**90.19**	**76.37**	**90.34**	5.96	89.09	72.54
	CNM	43.3	**90.81**	66.06	83.74	92.09	77.94	24.4	89.96	75.79	90.29	**67.01**	89.14	74.21
	CRA+CNM	43.3	90.02	66.06	83.74	92.09	**81.62**	**66.62**	**90.19**	**76.37**	90.29	61.05	**90.95**	77.69
	AHP	50.51	90.28	**74.37**	83.74	**93.12**	**81.62**	53.02	90.01	**76.37**	**90.34**	66.41	89.62	**78.28**
	Random	46.97	90.45	62.23	83.8	92.04	77.49	44.75	89.95	75.84	90.3	59.21	89.81	75.24
B → TB	DSE	0.0	88.67	63.18	**82.56**	90.83	**84.07**	60.47	87.37	68.15	**88.53**	65.57	87.54	72.24
	CRA	**20.02**	88.67	63.18	81.56	90.14	78.43	60.47	**88.66**	68.15	86.04	66.64	**87.72**	**73.31**
	CNM	**20.02**	**88.96**	51.26	**82.56**	90.83	**84.07**	**65.31**	**88.66**	68.15	86.99	66.64	84.37	73.15
	CRA+CNM	**20.02**	88.67	63.18	81.56	90.14	78.43	60.47	**88.66**	68.15	86.04	66.64	**87.72**	**73.31**
	AHP	**20.02**	**88.96**	**69.68**	**82.56**	**91.86**	81.86	2.41	87.37	**70.78**	85.79	**68.72**	86.73	69.73
	Random	17.39	86.65	57.24	80.08	89.99	75.79	25.06	88.19	67.75	86.76	67.18	86.98	69.09

Table 9: Actual target task performance when task similarity estimation and transfer learning use different underlying PLMs. B/TB: BERT/TinyBERT. Type ($x \rightarrow y$) denotes that the most similar source task selected according to a task similarity estimation method with underlying PLM x is used for transfer learning to a target task with underlying PLM y.

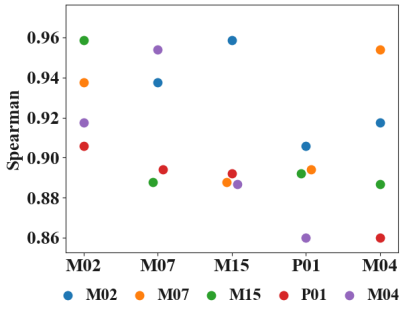
ison. The results of "Random" in the two tables are averaged over 5000 times of random sampling. Specifically, each round of random sampling selects a task other than itself for each target task as the source task for transfer learning.

A.5 Analysis on the Generality of CNM across Subjects

We calculate task similarity matrices that are averaged over 5 subjects in CNM. Are results for individual subjects are consistent with each other? Hence we separately calculated task similarity matrices for each subject and used the Spearman correlation coefficient to measure task similarity matrix correlations among subjects. Results are shown in Figure 7, which indicates high correlations among subjects.



(a) BERT



(b) TinyBERT

Figure 7: Correlations among task similarity matrices calculated in CNM across different subjects with BERT (a) and TinyBERT (b). Dots of the same color refer to the same subject across experiments.