Identifying Suitable Tasks for Inductive Transfer Through the Analysis of Feature Attributions

Alexander J. Hepburn[0000−0001−6630−0258]
Richard McCreadie[0000−0002−2751−2087]

University of Glasgow, University Avenue, Glasgow, G12 8QQ, Scotland
a.hepburn.1@research.gla.ac.uk
richard.mccreadie@glasgow.ac.uk

Abstract. Transfer learning approaches have shown to significantly improve performance on downstream tasks. However, it is common for prior works to only report where transfer learning was beneficial, ignoring the significant trial-and-error required to find effective settings for transfer. Indeed, not all task combinations lead to performance benefits, and brute-force searching rapidly becomes computationally infeasible. Hence the question arises, *can we predict whether transfer between two tasks will be beneficial without actually performing the experiment?* In this paper, we leverage explainability techniques to effectively predict whether task pairs will be complementary, through comparison of neural network activation between single-task models. In this way, we can avoid grid-searches over all task and hyperparameter combinations, dramatically reducing the time needed to find effective task pairs. Our results show that, through this approach, it is possible to reduce training time by up to 83.5% at a cost of only 0.034 reduction in positive-class F1 on the TREC-IS 2020-A dataset.

Keywords: Explainability · Transfer Learning · Classification.

1 Introduction

Transfer learning is a method of optimisation where models trained on one task are repurposed for another downstream task. The intuition behind this approach is clear; as human beings, we often apply knowledge learned from previous experience when learning a new, related skill. Hence, transfer learning aims to mimic this biological behaviour by exploiting the relatedness between tasks.

However, there remains an ever-present question that researchers have long strived to answer, *Why is pretraining useful for my task?* More specifically, *What information encoded in a pretrained model is transferrable for my task?* If, hypothetically, we are capable of approximating, prior to training, which auxiliary tasks will be useful in practice, we are then able to avoid the often laborious process of trial-and-error over all task and parameter combinations. Hence, we propose a solution which leverages recent research in explainability to identify the properties that characterise particular tasks and by extension, the properties which make these tasks related.

Through the evaluation of 803 models, we calculate the per-document term activity for each task and use these to predict the performance outputs of each combined task pair. We show that there exists correlation between strongly-attributed shared terms between pairs of single tasks and their combined performance output, and that, by ranking each task pair by their performance, we can reduce the time it takes to find the best-performing model by up to 83.5% (with a cost of only 0.034 reduction in positive-class F1).

2 Improving Performance Through Inductive Transfer

The concept of *inductive transfer*, introduced by Pan and Yang [9], can be considered a method of transfer learning wherein the source and target tasks are different, the goal of which is to leverage domain information in the source task—encoded in the training signals as an inductive bias—to be transferred to a downstream, target task.

However, the necessary conditions for what constitutes a suitable auxiliary task for use in pretraining is unclear. Mou et al. [7] note that the difficulty in transferability in this domain lies in the discreteness of word tokens and their embeddings. Similar to this work, Bingel and Søgaard [1] identified beneficial task relations for multi-task learning and found that performance gains were predictable from the dataset characteristics. While ground has been covered in understanding and quantifying the relationship between pairs of tasks, what constitutes task relatedness remains an open question. To this end, we first demonstrate the efficacy of transfer learning as a method of improving classifier performance. We utilise the dataset provided by the TREC Incident Streams Track (TREC-IS) which features a number of multi-label classification tasks wherein each label is representative of some information need (known as *information types*) to end users of automated crisis and disaster systems. More importantly, these labels exhibit some level of conceptual relatedness, and as such, is an appropriate framework for this investigation. The track features 25 labels which manual assessors may ascribe to each document, however, to limit the number of models trained, we use the track's **Task 2** formulation, which restricts the number of information types to 12[1].

We experiment with transfer learning across these information types, that is to say, we train a particular classifier on one, *source* task and then use the resulting model as a pretrained baseline for tuning another downstream *target* task, using a pretrained BERT transformer model as defined by Devlin et al. [3] as the base model for our experiments.

Table 1 shows the single- and multi-task model results from previous experiments, containing each task's baseline performance (omitting 4 tasks which showed no performance change) and their respective best-performing auxiliary task when used as a prior. With the exception of those omitted tasks, we observed performance increases across the board, as can be seen from comparing

[1] More information on metrics and tasks can be found at http://trecis.org

		Inductive Transfer (Source)				Target Parameters			Evaluation Scores	
Target	Model	Transfer-From	LR	#E	B#	LR	#E	B#	Positive F1	Accuracy
New Sub Event	BERT→Target	None	-	-	-	2e-05	4	16	0.0258	0.9604
	BERT→Source→Target	Other (Best)	1e-05	2	32	2e-05	2	32	0.0578	0.9432
First Party Observation	BERT→Target	None	-	-	-	2e-05	4	32	0.0259	0.9646
	BERT→Source→Target	Move People (Best)	1e-05	2	32	1e-05	1	32	0.1142	0.9538
Service Available	BERT→Target	None	-	-	-	3e-05	3	16	0.0944	0.9821
	BERT→Source→Target	Other (Best)	1e-05	1	32	1e-05	1	32	0.1095	0.9783
Move People	BERT→Target	None	-	-	-	2e-05	3	32	0.1964	0.9835
	BERT→Source→Target	Other (Best)	1e-05	1	32	1e-05	2	32	0.2423	0.9853
Emerging Threats	BERT→Target	None	-	-	-	3e-05	2	32	0.2329	0.8323
	BERT→Source→Target	Location (Best)	1e-05	2	32	1e-05	1	32	0.2612	0.8135
Multimedia Share	BERT→Target	None	-	-	-	2e-05	3	32	0.4356	0.6760
	BERT→Source→Target	Other (Best)	1e-05	2	32	2e-05	1	32	0.4709	0.6422
Location	BERT→Target	None	-	-	-	3e-05	2	16	0.5904	0.6939
	BERT→Source→Target	Multimedia Share (Best)	1e-05	1	32	1e-05	1	32	0.6178	0.7196
Other	BERT→Target	None	-	-	-	5e-05	4	16	0.6831	0.5638
	BERT→Source→Target	Multimedia Share (Best)	2e-05	1	32	1e-05	2	32	0.6853	0.7187
AVERAGE	BERT→Target	None	-			*Varies*			0.2856	0.8321
	BERT→Source→Target	*Varies*	*Varies*			*Varies*			0.3199	0.8443

Table 1: Information type categorisation performance with and without inductive transfer from a source task. Metrics are micro-averaged across events and range from 0 to 1, higher is better.

the BERT→Target and BERT→Source→Target rows for each task in the above table. However, obtaining these improvements was not a trivial process. We found that performance increases were highly dependent on the target information type and that the effectiveness of transfer was highly sensitive to changes in model hyperparameters. Moreover, there were no easily discernible patterns that we could use as heuristics to speed up the process of finding the best model, leading to an exhaustive grid-search over all task and parameter combinations, calling into question the practicality of such an approach in production. Hence, if we are to realise these performance gains, a cheaper approach to finding effective pairs of tasks is needed.

3 Optimising Transfer Learning with Explainability

As the complexity of deep neural models grows exponentially, there is an increasing need for methods to enable a deeper understanding of the latent patterns of a neural model, such as when trying to understand cases where that model has failed. In order to understand this behaviour, we must explore methods of explaining the inner working of language models.

Explainability is a field focused on model understanding and the predictive transparency of machine learning-based systems. A number of explainability techniques take the form of gradient-based approaches [8,10]. One such gradient-based approach, known as *attribution*-based explanations, allow us to assess what the dominant features were that contributed to a particular prediction. Various algorithms can assign an importance score to each given input feature and effectively summarise and visualise these scores in a human-readable manner.

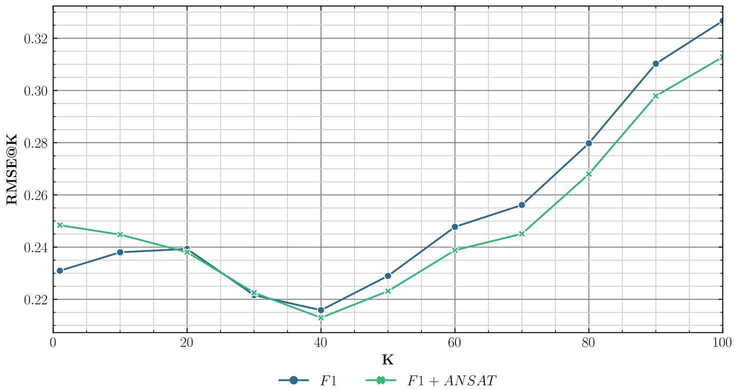

Fig. 1: RMSE@k results from our XGBoost regression model. RMSE metrics range from 0 to ∞, lower is better.

Attribution-based explainability has become especially popular in the literature [5, 6, 12–14], however, research into explainability for transfer learning is sparse.

In this work, we investigate: 1) whether there exists correlation between the shared, *important* linguistic properties of a pair of tasks and their combined performance output; 2) whether we can compute this relationship prior to training these combined models; and 3) whether we can, as a result, reduce the time taken to produce high-performance models. As such, we divide the remainder of this paper into the following research questions:

RQ1. Does there exist some degree of correlation between the shared, *active* terms between pairs of tasks and their combined performance output?

RQ2. Can we leverage this knowledge, prior to training, to reduce the overall runtime required to produce effective models?

To this end, we compute the *conductance* of latent features in the context of each document. Introduced by Dhamdhere et al. [4, 11] the conductance of a hidden unit can be described as the flow of attributions via said unit. By computing the conductance, we are able to quantify the bearing each individual input feature has on a particular prediction (with respect to a given input sequence).

For each BERT→Target model and each document in our test set, we calculate the effect any individual feature (term) had on the prediction output of its document using conductance. The conductance c of each term x_i within a document is scored $\{c_{x_i} \in \mathbb{R} : -1 \leq c_{x_i} \leq 1\}$ wherein $c_{x_i} \in [-1, 0)$ represents conductance scores that attribute towards the negative class and $c_{x_i} \in (0, 1]$ attribute towards our target class. We eliminate negatively attributed terms in order to capture the most *active* terms that represent our target class. We determine activity by testing against a range of thresholds for term activity (TAT), beginning from the mean of positively-attributed conductance scores, 0.05, and increasing to a reasonable upper bound at 0.05 increments. As such, we decided to test the set of conductance thresholds: $[0.05, 0.7] \cap 0.05\mathbb{Z}$. We then averaged

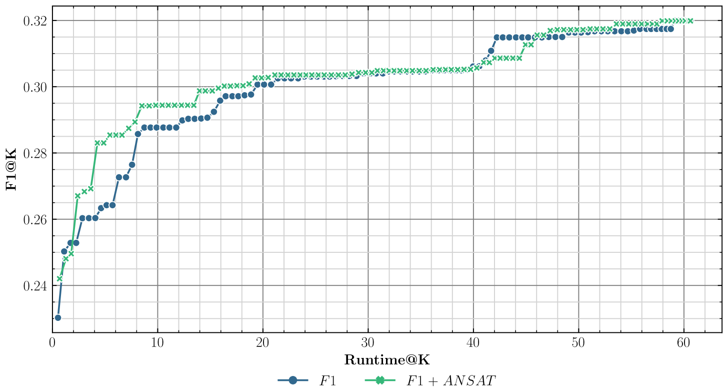

Fig. 2: Performance vs. Runtime results from XGBRegressor. F1 metrics range from 0 to 1, higher is better. Runtime is reported in hours.

the total number of active terms across documents and only consider those terms which are above said thresholds. Our calculations result in the following formulation, Average Number of Shared Active Terms ($ANSAT$), which provides a quantified comparison metric between each pair of models:

Definition 1. *Let $\mathcal{M}$ represent a neural model with layers $l \in L$ and $\mathcal{D}$ represent the collection of positive-class documents (with respect to task sets A, B, and AB) containing words $w \in d \in D$, and with conductance threshold TAT then:*

$$ANSAT(M_A, M_B, D, TAT) = \sum_{d \in D} \frac{\left(\sum_{w \in d} \begin{cases} 1, & \textit{if } \left(\frac{\sum_{l \in L_{M_A}} conduct(w,l)}{|L_{M_A}|} \geq TAT \right) AND \left(\frac{\sum_{l \in L_{M_B}} conduct(w,l)}{|L_{M_B}|} \geq TAT \right) \\ 0, & \textit{otherwise} \end{cases} \right)}{|D|} \tag{1}$$

Through this formulation, we can estimate the pretraining similarity between two tasks (A and B) via their underlying datasets (positive-class documents only) D_A and D_B, as well as the intersection of both, D_{AB}. We then use these estimates to predict the effectiveness of a combined model M_{AB} created via transfer learning, i.e. BERT→Source(A)→Target(B). In particular, we train an XGBoost [2] regression model (XGBRegressor) to produce a prediction of the effectiveness of M_{AB}, given various feature combinations. We use this model to predict the performance of M_{AB} combinations for each target task B given a set of source tasks A∈S, using Positive F1 as our target metric.

To answer RQ1, we compare the performance predicted by our XGBoost model when using only individual model effectiveness (M_A and M_B F1-scores) as features vs. those same features + the ANSAT similarity estimations. If active terms as defined by ANSAT are indicative of transfer performance then the XGBoost model with these features should be more effective than the one without. Fig. 1 shows the results of our experiment, reporting RMSE at ranks 10–100 with different feature sets, where $F1$ denotes M_A and M_B F1-scores and $ANSAT$ denotes the ANSAT scores for D_A, D_B and D_{AB} (under TAT values $[0.05, 0.7] \cap 0.05\mathbb{Z}$).

Near the top of the ranking (K=5, 10), we observe the feature set using F1 only to marginally outperform F1 + ANSAT by 1.76% and 2.76%, respectively. At ranks K=40 and above, however, we observe that the inclusion of ANSAT results in considerably lower error than using F1 scores alone. Indeed, from these results we can conclude that the overlap of active terms between tasks as measured by ANSAT is valuable evidence when attempting to determine whether the combination of tasks will result in performance gains, answering RQ1.

To answer RQ2, we consider the potential real-world benefits of such performance prediction models when used to reduce task-pair training time. For this experiment, we assume you have a certain budget to train K task-pair combinations and check their performance. For a task, the more combinations you try, the more likely you will find a good combination. As our XGBoost models are predicting which combinations will work well together, we can use this to determine the order of combinations to try, where the goal is to find the best performing combination for each task as early as possible, such that we can end the search early. Fig. 2 reports the Positive F1 performance of the best performing model for different depths K, where the x-axis is a conversion of K into the number of hours needed to train that many models for all tasks (Runtime@K).

From the collection of 803 models used as the dataset for our regression model, our best, average performance (F1) was 0.3199, which took 60.6 hours to train. By utilising our regression model, we are able to achieve an F1-score of 0.3003 (only 6.12% worse than our best-performing F1 model), at only 30 hours or 50.5% less training time, using the $F1 + ANSAT$ feature space. If we were to accept a 0.034 or 10.78% reduction in F1-score, we can further reduce our time to 10 hours or a 83.5% reduction in training time. We note that at lower ranks of K, we observe a consistent increase in performance when including ANSAT in our feature space. At ranks 7, and 10, we observe 8.71%, and 8.01% performance increases, respectively, when including ANSAT alongside F1. Considering these results, there is clearly significant scope for improving performance by leveraging attribution-based techniques, answering RQ2.

4 Conclusions and Future Work

In this work, we presented an approach for estimating the suitability for pairs of tasks to be used in transfer learning by comparing their shared, active terms. It is clear that there exists some correlation between term activity and performance, as highlighted by our results. By predicting the projected performance output of each task pair, we managed to achieve up to 83.5% reduction in training time (for only a 0.034 or 10.78% reduction in F1). However, while we have demonstrated the value of using conductance to estimate combined model performance pre-training, there is clearly more work needed to increase the accuracy of these estimations, and hence further reduce the space of models that need to be searched. As such, for future work, we propose further analysis into the quantifiable properties that constitute related tasks which could further improve inductive transfer between such tasks.

References

1. Bingel, J., Søgaard, A.: Identifying beneficial task relations for multi-task learning in deep neural networks. arXiv:1702.08303 [cs] (Feb 2017)
2. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. CoRR **abs/1603.02754** (2016), http://arxiv.org/abs/1603.02754
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (2019)
4. Dhamdhere, K., Sundararajan, M., Yan, Q.: How important is a neuron? CoRR **abs/1805.12233** (2018), http://arxiv.org/abs/1805.12233
5. Ismail, A.A., Gunady, M.K., Bravo, H.C., Feizi, S.: Benchmarking deep learning interpretability in time series predictions. CoRR **abs/2010.13924** (2020), https://arxiv.org/abs/2010.13924
6. Liu, N., Ge, Y., Li, L., Hu, X., Chen, R., Choi, S.: Explainable recommender systems via resolving learning representations. CoRR **abs/2008.09316** (2020), https://arxiv.org/abs/2008.09316
7. Mou, L., Meng, Z., Yan, R., Li, G., Xu, Y., Zhang, L., Jin, Z.: How Transferable are Neural Networks in NLP Applications? arXiv:1603.06111 [cs] (Oct 2016)
8. Mundhenk, T.N., Chen, B.Y., Friedland, G.: Efficient saliency maps for explainable AI. CoRR **abs/1911.11293** (2019), http://arxiv.org/abs/1911.11293
9. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**, 1345–1359 (2010)
10. Ribeiro, M.T., Singh, S., Guestrin, C.: "why should I trust you?": Explaining the predictions of any classifier. CoRR **abs/1602.04938** (2016), http://arxiv.org/abs/1602.04938
11. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. CoRR **abs/1703.01365** (2017), http://arxiv.org/abs/1703.01365
12. Wu, Z., Kao, B., Wu, T.H., Yin, P., Liu, Q.: Perq: Predicting, explaining, and rectifying failed questions in kb-qa systems. In: Proceedings of the 13th International Conference on Web Search and Data Mining. p. 663–671. WSDM '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3336191.3371782, https://doi.org/10.1145/3336191.3371782
13. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. CoRR **abs/1311.2901** (2013), http://arxiv.org/abs/1311.2901
14. Zhang, Z., Rudra, K., Anand, A.: Explain and predict, and then predict again. CoRR **abs/2101.04109** (2021), https://arxiv.org/abs/2101.04109