Predictive Model Selection for Transfer Learning in Sequence Labeling Tasks

Parul Awasthy[†], **Bishwaranjan Bhattacharjee**[†], **John R Kender**[†§] and **Radu Florian**[†]
[†] IBM Research AI
Yorktown Heights, NY 10598
USA
awasthyp,bhatta,raduf@us.ibm.com

[§]Columbia University
NY 10027
USA
jrk@cs.columbia.edu

Abstract

Transfer learning is a popular technique to learn a task using less training data and fewer compute resources. However, selecting the correct source model for transfer learning is a challenging task. We demonstrate a novel predictive method that determines which existing source model would minimize error for transfer learning to a given target. This technique does not require learning for prediction, and avoids computational costs of trial-and-error.

We have evaluated this technique on nine datasets across diverse domains, including newswire, user forums, air flight booking, cybersecurity news, etc. We show that it performs better than existing techniques such as fine-tuning over vanilla BERT, or curriculum learning over the largest dataset on top of BERT, resulting in average F_1 score gains in excess of 3%. Moreover, our technique consistently selects the best model using fewer tries.

1 Introduction

When deploying deep learning in real-life scenarios, training data is often sparse. Transfer learning improves learning of such target tasks by leveraging knowledge from a source task, as shown in Figure 1. The improvement in learning could be measured by either improvement in accuracy (for example, F_1 score), or reduction in the time taken to learn the task.

With the increased popularity of the large transformer-based models Devlin et al. (2019), transfer learning in the form of fine-tuning a base model is ubiquitous in NLP. However, the performance of the learned target model depends critically on the chosen source model. Simply selecting the largest dataset can lead to sub-optimal performance, and trying all sources is computationally expensive.

We demonstrate a prediction technique for the sequence labelling task, which given a target model, selects the "best" source model from among a set of available source models, according to a novel and inexpensive metric. Then, only that single selected (trained) source model is further fine-tuned on the target dataset. We show that our selection technique is effective at selecting the source model that most improves F_1 score, over nine different tasks, with no additional training. Further, our technique results in an average gain of over 3% in F_1 over selecting a base model randomly, and over 4% in F_1 over training a model without transfer learning from any source model.

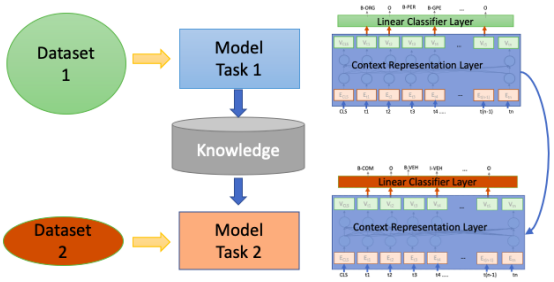

Figure 1: Transfer Learning Methodology. The rightmost column of the figure shows the architecture of a Sequence Labelling Model. Transfer learning reuses the context representation layer of Model 1 to fine-tune a new representation layer for Model 2.

In the rest of the paper, we chronicle prior work in the area of transfer learning, describe the sequence labeling task, and follow with the description of our predictive model selection methodology.

2 Prior Work

The transfer learning literature spans different topics and strategies such as few-shot learning (Socher et al., 2013), domain adaptation (Patricia and Caputo, 2014), weight synthesis (Sussillo and Abbott, 2017), and multitask learning (Jiang, 2009; Nguyen et al., 2016; Torrey and Shavlik, 2009). Some works propose novel combinations of these approaches, in order to improve transfer performance

Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing, pages 113–118
Online, November 20, 2020.

Figure 2: Example of Span Similarity (SS) and Weighted Span Similarity (WSS). Top line: input sentence. Next line: ground truth labeling. Last two lines: labelings due to two different transfer learning sources. Rightmost columns: metric values * 100, showing the "goodness" of each source as SrcF1. Note that SrcF1 * SS = WSS.

under conditions of domain transfer with limited or incomplete annotations (Luo et al., 2017).

Some prior research on optimizing source selection has focused on instance transfer techniques, like Zhou et al. (2016) and Lin et al. (2013), which select a subset of examples from a source for transfer learning. Other approaches, like Schultz et al. (2018), propose methods to select the right set of source domain datasets for a task such as sentiment classification. Yet another approach, Afridi et al. (2018), which is more popular in computer vision research, selects sub-models of the source model and re-trains only on those.

Alternatively, Bhattacharjee et al. (2020) focuses on predicting the best source model, in the domains of computer vision and of semantic relations, by measuring the similarity between the source and target datasets, using a mix of metrics like KL divergence and source dataset size. In NLP, this approach works for sentence-level classification tasks such as sentiment classification. But it does not adapt well for sequence labelling tasks, where labels span across tokens rather than sentences, and where the similarity of the label spaces need to be accounted for. Additionally, their approach requires either the entire source dataset or its feature vector representation to be available. In contrast, the method we demonstrate here only needs the source model itself.

3 Task Definition

A sequence labelling task assigns a label to each member of a sequence of observed values. An example is Named Entity Recognition (NER), which identifies in unstructured text all contiguous typed references to task-specific real-world entities, such as persons, organizations, facilities, locations, etc. An example is shown in Figure 2.

We formally define the Transfer Learning task for this paper as follows: given a set $\mathcal{M}$ of N source models trained for sequence labelling, $\mathcal{M} = \{M_1, M_2, \ldots, M_N\}$, and one target set t, the task is to find the best source model M_k, which when used as a base model for transfer learning, would result in a model with highest performance.

We use F_1 as the metric to measure performance, and compute relative gain in F_1 to measure the improvement in performance.

4 Predictive Model Selection

To select the best transfer learning base model, our method compares the target and source using a novel similarity metric. Instead of comparing the source and target datasets, our method compares the target test-set with the output of the source model on the target test-set. This comparison takes label weights into account.

To compare a target with each of the source models, we decode the target test-set through the source model. We call this output $\hat{y}$, and the original target annotation y. We next compare y and $\hat{y}$ using the metrics described. The source model with the highest metric score is chosen as the best source model.

4.1 Metrics

For predicting which model would be the best for a target dataset, we have experimented with two measures, called Span Similarity and Weighted Span Similarity, described here.

For Named Entity Recognition, an extracted span is customarily considered correct if the offset of the span matches that of the reference span, and the type of the span matches that of the reference span. In this work, however, we ignore the types of the spans. We therefore define *Span Similarity (SS)* based on the score computed between gold (y) and system output ($\hat{y}$) using only the offsets.

$$SS = \frac{2 * TP}{2 * TP + FP + FN} \quad (1)$$

where TP = number of true positives, FP = number of false positives, and FN = number of false negatives, as decided by the above selection criteria. This is basically the Sorensen-Dice coefficient.

Dataset	Description	Label-Set Size	Source Size	Source F_1	Target Size	Target F_1
7 Categories (Coucke et al., 2017)	7 similar workspaces	71	2802	96.46	468	86.49
Alchemy1	News	47	2100	86.68	431	62.93
Alchemy2	News	54	7994	87.19	799	64.83
ATIS (Dahl et al., 1994)	Airline Travel	121	5873	96.91	647	92.84
CoNLL (Sang and De Meulder, 2003)	News	9	18467	96.81	1847	91.49
Klue Forum (Florian et al., 2004)	User Forum part of Klue	1050	19323	84.14	1933	75.28
Klue News (Florian et al., 2004)	Large News part of Klue	1050	14586	87.31	1514	82.95
TAC (LDC, 2019)	News	13	9639	79.76	1082	75.5
Cybersecurity	Cybersecurity articles	85	55386	83.14	2405	73.7

Table 1: Details of all datasets. Dataset size is in number of sentences. F_1 values are scores * 100. Each target dataset is down-sampled from its source dataset, allowing a full comparison matrix of sources versus targets.

To account for the "goodness" of the source model, we weight Span Similarity by the F_1 score of the source model on the source test-set, $F_1(s)$. This we call the *Weighted Span Similarity (WSS)*.

$$WSS = F_1(s) * SS \quad (2)$$

We select the source model with the highest WSS score to be the best base model for transfer learning.

4.2 Transfer Learning

The architecture of a typical transformer-based Named Entity Recognition model is shown in Figure 1. The model can be divided into two parts, the context representation encoding layer (e.g., a BERT model), and the classifier layer (e.g, a linear classifier).

Once the source model with the highest WSS is selected, we use it as the base for transfer learning. To capture the knowledge of the source model, we use the context representation layer of the source model, but replace its classifier layer with a new classifier mapped to the target model space. We then fine-tune this new model on the target dataset.

5 Experimental Evaluation

5.1 Datasets and Source Models

We test our method on the various datasets shown in Table 1, all comprising of named entity annotated data, with different number of types as described in the lined citations. Alchemy1 and Alchemy2 are newswire datasets labeled internally with 47 and 54 types (person, organization, company, etc), respectively. Cybersecurity, the other dataset that is not cited, is a dataset of cybersecurity related articles (descriptions of virus attacks, etc.), labeled internally.

For each of the datasets, we sample a small percentage (5–20%) of examples in order to create our target sets. We use the full dataset as a source, and the small sampled sets as target sets. We train NER models using the method described in Devlin et al. (2019) on full source datasets, using the setup described in Section 5.4.

5.2 Generating Ground Truth

To test our method, we need to determine which source is truly the best for a given target. We proceed as follows.

We formally denote the N source datasets as $\mathcal{S} = \{s_1, s_2, \ldots, s_N\}$, and the N target datasets as $\mathcal{T} = \{t_1, t_2, \ldots, t_N\}$.

To set up the evaluation of our method, we first train NER models on all source datasets, to get a set of source models $M = \{M_1, M_2, \ldots, M_N\}$. Each one of these is comprised of a context layer and a classifier layer.

Next, to get the absolute ground truth model for a given target dataset t_k, we train a model G_k for t_k without any transfer learning.

Lastly, we train a suite of ground truth transfer models for each dataset t_k. We fine-tune each of the source models $M_i, i \neq k$, by retaining its context layer but adapting its classifier layer. This gives for each t_k a suite of ground truth transfer models, $\mathcal{G}_k = \{G_{k,i}, i \neq k\}$, where $G_{k,i}$ is the ground truth transfer model for t_k using source M_i.

For each of the models in M_k we then compute the relative gain in F_1 in the usual way:

$$RGF_1(k,i) = (F_1(k,i) - F_1(k))/F_1(k) \quad (3)$$

where $F_1(k,i)$ returns the F_1 score of the model $G_{k,i}$, and where $F_1(k)$ does the same for G_k. Therefore, the best ground truth transfer model for t_k is defined to be: $G_k^* = \underset{i}{argmax}\, RGF_1(k,i)$

5.3 Baselines

We compare our method to the following baselines:

Target	SS Predict Correct	WSS Predict Correct	Largest Source $(F_1 - B_1)/B_1$	Random Selection $(F_1 - B_2)/B_2$	Cosine Similarity $(F_1 - B_3)/B_3$	KL Divergence $(F_1 - B_4)/B_4$	Target Only $(F_1 - B_5)/B_5$
Alchemy1	Yes	**Yes**	25.66	15.95	0	5.45	23.17
Alchemy2	Yes	**Yes**	4.08	4.09	4.42	4.09	11.23
Atis	Yes	**Yes**	1.12	0.93	0.43	1.12	0.4
Klue Forum	Yes	**Yes**	0	2.24	3.26	2.33	3.13
Klue News	Yes	**Yes**	0	1.99	3.32	2.08	2.78
CoNLL	No	**Yes**	1.57	1.33	0	1.85	1.05
Cyber	No	**Yes**	0.48	0.93	0	0.48	1.75
7 Categories	No	No	-2.48	1.15	-0.93	-2.48	-0.68
TAC	No	No	-3.11	2.25	0	-2.68	0.83
$\overline{RGF_1}$			3.04	3.43	1.17	1.36	4.85

Table 2: Results, showing values * 100. $\overline{RGF_1}$ = Average Relative Gain using our method. $B_1 = F_1$ when source is largest training set. $B_2 = \overline{F_1}$, average of randomly picked source models. $B_3 = F_1$ when source is model with max cosine. $B_4 = F_1$ when source is model with lowest D_{KL}. $B_5 = F_1$ when model learnt only on target data.

1. Largest Source: This method picks the source with the largest dataset size as the best base model.

2. Random Selection: This method picks a source at random as the best base model.

3. Cosine Similarity: Cosine similarity has been frequently used in distributional semantics (Mikolov et al., 2013; Peterson, 2009; Wagstaff et al., 2001). We compute the cosine similarity between a target model G_k and each of the source models M_i (see Section 5.2), by decoding the t_k test-set with both target and source models, and using the outputs of their respective context representation layers, called A and B, to compute:

$$CosSim(A, B) = \frac{\Sigma_i A_i \times B_i}{\sqrt{\Sigma_i A_i^2} \times \sqrt{\Sigma_i B_i^2}} \quad (4)$$

We do this for all $M_i, i \neq k$, and choose the M_i with highest cosine similarity with the test-set t_k.

4. KL Divergence: Bhattacharjee et al. (2020) use KL divergence as selection metric in their method. To compute the KL Divergence between the source dataset s_i and the target dataset t_k, we decode both datasets with M_i, and compare their context representation layers, called P and Q, to compute:

$$D_{KL}(P, Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \quad (5)$$

5. Target-only Model: We also compare our method with models trained directly with vanilla BERT over the target test-set, i.e., over t_k as described in Section 5.2.

5.4 Experimental Setup Details

The models are built using the HuggingFace PyTorch implementation of Transformers Wolf et al. (2019). Our model uses *bert-base-cased* with the standard hyperparameters. We train the source and target models for 20 epochs, with a learning rate of 5e-5 and a batch of 32. We use K80 gpus to train our models.

6 Results and Discussion

6.1 Accuracy and Time Cost

Table 2 shows a summary of the F_1 gains by using our method to predict the best source model, compared to other baseline selection methods. Our SS method is able to predict the correct source model 5 out of 9 times, and our WSS method can predict the correct source model 7 out of 9 times. This is significantly better than any other baseline.

In terms of accuracy, our WSS method outperforms the baselines, as follows: largest source, 6 out of 9 times, with average gain of 3.04%; random selection, 9 out of 9 times, with average gain of 3.43%; cosine similarity, 4 out of 9 times, with average gain of 1.17%; KL divergence, 7 out of 9 times, with average gain of 1.36%; and no NER transfer learning, 8 out of 9 times, with average gain of 4.85%.

We note that some of our performance improvements can be attributed to a fundamental difference between our method and other baseline methods like Cosine Similarity, KL divergence, etc. Whereas other methods compare similarity of the input text space, our method computes similarity within the label space, taking advantage of learned contextual relationships.

Our method consistently works across diverse domains, and is able to correctly predict models particularly for news, forum, airline travel, and cybersecurity domains. None of the baselines work as well across these domains.

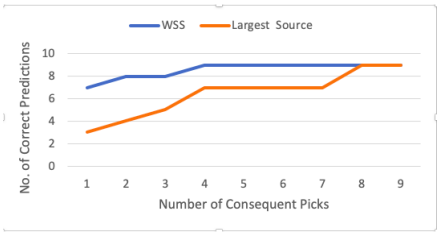

Figure 3: Number of correctly selected best models versus number of attempts. Our method in one try does as well as the largest source baseline does after four.

Our method also saves substantially on computational time and resources, as it finds the best source model with fewer tries, as shown in Figure 3. In most cases, our method is able to predict the best model on the first try, whereas the largest source baseline needs four tries. With the experimental setup described in section 5.4 it takes on average 1.5 hours to train a target model, as compared to the 6 hours it takes the largest source baseline to produce the best model. This is a compute cost saving of 75%.

6.2 Potential Application to Computer Vision

The problem of finding both a span boundary and a label is not limited to sequence labelling in NLP. Object Detection (Szegedy et al., 2013) in vision research, has similar requirements: one is expected to label objects in an image and to mark the bounding box of the objects individually. Figure 4 shows an example where the task is to detect cookies and mark the bounding boxes around them. Moreover, both Object Detection and Sequence Labelling have similar measures of accuracy.

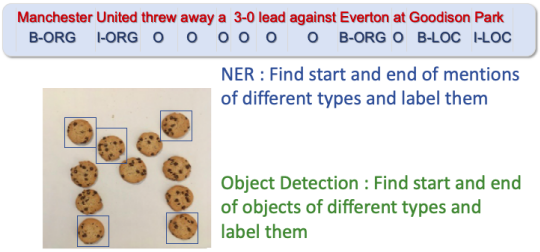

Figure 4: Similarity between Mention Detection in NLP and Object Detection in Vision.

It would be interesting to expand the method proposed in this paper to determine which source object detection base model would be a good fit for a given target data set from a collection of source object detection models. We plan to explore this in future.

7 Conclusion

In this paper we present a simple and effective method to predict the best source model for transfer learning in a sequence labelling task, NER. We show our method outperforms popular baselines such as the selection of the largest source, by an average relative 3 F_1 points. And, it is more than average relative 4 F_1 points better than a method that does not use NER transfer learning. Moreover, our method consistently selects the best model with fewer tries, saving computational cycles by roughly 75%.

8 Acknowledgments and Disclaimer

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. FA8750-19-C-1001.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

References

Muhammad Jamal Afridi, Arun Ross, and Erik M Shapiro. 2018. On automated source selection for transfer learning in convolutional neural networks. *Pattern recognition*, 73:65–75.

Bishwaranjan Bhattacharjee, John R Kender, Matthew Hill, Parijat Dube, Siyu Huo, Michael R Glass, Brian Belgodere, Sharath Pankanti, Noel Codella, and Patrick Watson. 2020. P2l: Predicting transfer learning for images and semantic relations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 760–761.

Alice Coucke, Adrien Ball, ClÃ©ment Delpuech, ClÃ©ment Doumouro, Sylvain Raybaud, Thibault Gisselbrecht, and Joseph Dureau. 2017. Benchmarking natural language understanding systems: Google, facebook, microsoft, amazon, and snips.

Deborah A Dahl, Madeleine Bates, Michael K Brown, William M Fisher, Kate Hunicke-Smith, David S Pallett, Christine Pao, Alexander Rudnicky, and Elizabeth Shriberg. 1994. Expanding the scope of the atis task: The atis-3 corpus. In *HUMAN LANGUAGE TECHNOLOGY: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Radu Florian, Hany Hassan, Abraham Ittycheriah, Hongyan Jing, Nanda Kambhatla, Xiaoqiang Luo, H Nicolov, and Salim Roukos. 2004. A statistical model for multilingual entity detection and tracking. Technical report, IBM Thomas J. Watson Research Center, Yorktown Heights, NY.

J Jiang. 2009. Multi-task transfer learning for weakly supervised relation extraction. In *4th International Joint Conference on Natural Language Processing. Association for Computational Linguistics*.

LDC. 2019. Tac kbp entity discovery and linking - comprehensive training and evaluation data 2014-2015.

Di Lin, Xing An, and Jian Zhang. 2013. Double-bootstrapping source data selection for instance-based transfer learning. *Pattern Recognition Letters*, 34:1279–1285.

Z Luo, Y Zou, J Hoffman, and L Fei-Fei. 2017. efficient learning of transferable representations across domains and tasks. In *Conference on Neural Information Processing Systems (NIPS)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

T.H Nguyen, L Fu, K Cho, and R Grishman. 2016. Two-stage approach for extending event detection to new types via neural networks. In *ACL Representation Learning for NLP Workshop*.

N Patricia and B Caputo. 2014. Learning to learn, from transfer learning to domain adaptation: A unifying perspective. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Leif E Peterson. 2009. K-nearest neighbor. *Scholarpedia*, 4(2):1883.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Lex Razoux Schultz, Marco Loog, and Peyman Mohajerin Esfahani. 2018. Distance based source domain selection for sentiment classification. *arXiv preprint arXiv:1808.09271*.

R Socher, M Ganjoo, H Sridhar, O Bastani, D Manning, C, and A.Y Ng. 2013. Zero-shot learning through cross-modal transfer. pages 935–943.

D Sussillo and L Abbott. 2017. Transferring learning from external to internal weights in echo-state networks with sparse connectivity. In *PLoS ONE*.

Christian Szegedy, Alexander Toshev, and Dumitru Erhan. 2013. Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561.

L. Torrey and J. Shavlik. 2009. Transfer learning. In *Handbook of Research on Machine Learning Applications*.

Kiri Wagstaff, Claire Cardie, Seth Rogers, Stefan Schrödl, et al. 2001. Constrained k-means clustering with background knowledge. In *Icml*, volume 1, pages 577–584.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Shuang Zhou, Gijs Schoenmakers, Evgueni Smirnov, Ralf Peeters, Kurt Driessens, and Siqi Chen. 2016. Largest source subset selection for instance transfer. In *Asian Conference on Machine Learning*, pages 423–438.