CS4125 Coursework B

Julián Urbano

March 18, 2020

1. Introduction

In the context of Machine Learning, transfer learning broadly refers to the technique in which a model can be trained to solve one problem, and later refined to solve another problem. For instance, a neural network trained to recognize musical instruments in a music audio clip, could be reused to classify musical genre.

Let us consider the traditional Machine Learning scenario in which a researcher wants to build a model for some specific target task. These steps are followed:

1. Collect a test dataset (TeD) for the target task.
2. Use cross-validation to train a model that maximizes some metric.
3. The average metric score is the indicator of model performance.

In the transfer learning scenario, these steps are followed:

1. Collect one or more training datasets TrD1, TrD2, etc, for related tasks, and a test dataset TeD for the target task.
2. Train a model with all TrD's simultaneously to maximize their task-specific metric.
3. Use cross-validation to refine the model to optimize for TeD.
4. The average metric score is the indicator of model performance.

The purpose of this assignment is to analyze the data from an existing experiment to study transfer learning.

2. Research Questions

There are three main research questions we want to study:

1. RQ1: How much does transfer learning improve over typical non-transfer learning?
2. RQ3: What is the effect of different strategies to simultaneously learn one model from multiple TrD's?
3. RQ2: What is the effect of the TrD's on the final model performance?

2.1 RQ1: Improvement of transfer learning

The first question of interest is whether transfer learning from multiple training datasets (second scenario) achieves better performance than simple models trained for the specific target task without transfer learning (first scenario). To this end, we have developed three simple models that can act as baselines in the first scenario: B1, B2 and B3. These models are trained directly for the target TeD of interest.

2.2 RQ2: Effect of learning strategy

Given a set of training datasets, there are different options as to how to simultaneously train a model with them. For our case, we developed one generic strategy in which the model is trained with the multiple TrD's, and only a fraction of it is refined for the particular TeD. The difference consists in how large this fraction is, which is controlled by a single scalar parameter.

Table 1: Models tested in the experiment.

B1	Baseline model with no TrD
B2	Baseline model with no TrD
B3	Baseline model with no TrD
S	Single TrD model, refined for TeD
MN	Multiple TrD model, No retraining for TeD
M1	Multiple TrD model, 1 part refined for TeD
M2	Multiple TrD model, 2 parts refined for TeD
M3	Multiple TrD model, 3 parts refined for TeD
MF	Multiple TrD model, full model refined for TeD

Table 2: Testing Datasets.

Name	Type
TeD1	Classification
TeD2	Classification
TeD3	Classification
TeD4	Classification
TeD5	Recommendation
TeD6	Regression
TeD7	Regression

Following this strategy, we consider the following models:

- MN: No retraining specific to TeD.
- M1: 1 part refined for the TeD.
- M2: 2 parts refined for the TeD.
- M3: 3 parts refined for the TeD.
- MF: full model refined for the TeD.

In addition, we consider a simple model, called S, in which a single TrD is used to train the model, which is then refined for the target TeD.

2.3 RQ3: Effect of training datasets

To make our results general, we studied eight different TrD datasets to train models. They all contain the same instances but differ in the particularities of the task (eg. classification, ranking, etc). No other description is available about them.

3. Experiment

An experiment has been carried out to study our three research questions. The main factors considered are:

- Model. We tested the three baselines (B1, B2 and B3), the single-TrD model (S), and five models that use multiple TrD's (MN, M1, M2, M3 and MF). Table 1 contains a summary.
- Training datasets. We consider eight different training datasets as mentioned earlier.
- Test dataset. We consider seven different test datasets, as summarized in Table 2.

The following process was followed for every TeD:

- All three baselines were trained and tested, yielding $3 \times 7 = 21$ datapoints.
- The S models were trained on each TrD and tested on the TeD. This was repeated 6 times to minimize random effects, leading to $8 \times 6 \times 7 = 336$ datapoints.
- The five Mx models were trained on a random combination of 2 to 8 TrD's and tested on the TeD. This was repeated 377 times, for a total of $377 \times 7 = 2639$ datapoints.

In total, there are therefore 2996 datapoints.

4. Assignment

For Coursework B you are given the data from this experiment in file `data.csv`. You and are expected to carry out a proper analysis of the data in order to answer the three research questions. You will write your findings in a short report, as if it was the corresponding *Results* section of a paper, including graphics and text. Your submission must include a single `.pdf` file with the results, and a single `.R` file with the code you used to analyze the data; both files will be submitted through Brightspace as a single `.zip` file or similar. Alternatively, you may also submit a single `.Rmd` file with the text and code together.

During the lecture on March 27th, all groups will give a small presentation of their work. Slides are suggested but not necessary; you may present using the report.

The structure of the data is as follows:

```
> d <- read.csv("data.csv")
> str(d)
'data.frame': 2996 obs. of  11 variables:
 $ TeD  : Factor w/ 7 levels "TeD1","TeD2",..: 1 2 3 4 5 6 7 1 2 3 ...
 $ TrD1 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD2 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD3 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD4 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD5 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD6 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD7 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ TrD8 : int  0 0 0 0 0 0 0 0 0 0 ...
 $ model: Factor w/ 9 levels "B1","B2","B3",..: 1 1 1 1 1 1 1 2 2 2 ...
 $ score: num  0.6932 0.5807 0.6772 0.5664 0.0373 ...
```

Each of the `TrD` columns is `1` if the training dataset is used, and `0` if not.