Final Project – Transfer Learning

IDS-705 Principles of Machine Learning

*Team 8:*

*Yuan Feng [yf115], Sebastián Soriano Pérez [ss1072], Vishaal Venkatesh [vv58],*

*Abhiraj Vinnakota [agv9], Roderick Whang [rjw34]*

1. **Abstract** [150 words maximum]

[What is the concept you created a tutorial for and why should the reader be interested? How is it valuable for machine learning? How might a practitioner use it?]

1. **Introduction**

[Provide a description concept/algorithm, motivate your reader as to why he/she should care about this question. How has the technique been used in practice and what are some motivating examples of its use?]

1. **Background**

[This section should cite problems that have been previously addressed that relate to your work, and the key takeaways of the studies that explored that work. The idea here is to place the problem you’re working on in context and to let the reader know that you’re not working in a knowledge vacuum. For finding relevant literature, a good starting point is Google Scholar.]

1. **Methods**

In order to understand transfer learning, we must define two important concepts first: *domain* and *task*. In a machine learning, the **domain** consists of the feature space in the dataset and its marginal probability distribution , where there are features or predictors in the feature space: . The marginal probability distribution is the probability distribution of each of the variables with no reference to the values of the rest of the variables. A **task** consists of the label space and the conditional probability distribution learned from the training data , where . For a graphical representation see Figure 4.1.

Transfer learning is done by first training a machine learning model with training data having a source domain and a source task . For a binary classification problem, the training dataset would consist of pairs of values, where , the feature vector is , and . It is assumed that there is sufficient training data available to train the machine learning model, and this data is usually balanced. Then, transfer learning utilizes the information learned from this model and applies it to train new model with data having a specific target domain and a target task . At this point, it is possible to try to estimate the target conditional probability distribution by either fixing certain model parameters found in the source model or using them as a starting point with really small learning rates. See Figure 4.2.

The transfer learning techniques can be applied to different scenarios, depending on whether it is the target and source domains, or tasks that differ. The domains differ when the source and target feature spaces are different or when the source and target marginal probability distributions are different. Likewise, the tasks differ when the source and target label spaces are different, when the conditional probability distributions are different, or both. Transfer learning could be applied in any of these situations.

Transfer learning is particularly popular for training neural networks for image classification, especially for convolutional neural networks. When we need to train a classification model but there is limited amount of labeled data available, we can use the information obtained from training a different model with more reliable data in a related domain. While transfer learning can be applied to various machine learning techniques for each of the scenarios described above, we will now focus on the application of transfer learning on deep learning methods for image classification problems with domain adaptation (when the source and target marginal probability distributions differ).

A neural network for image classification consists of an input layer where each neuron corresponds to one of the features (where ), a varying number of hidden layers, and an output layer where each neuron corresponds to one of the labels of . The general idea behind how a neural network “learns” is that each layer after the input layer captures slightly more complex features in the images. The first layers’ neurons activate when there are specific edges in certain regions of the images, and subsequent layers start detecting more complex patterns and shapes. The final layer receives as an input more easily identifiable and complex features, so that it is easier to correctly detect which features are found on each label of .

A common approach to applying transfer learning for computer vision is to utilize a previously trained model on data with the same feature space but different marginal probability distribution, when the labeled data is sufficient. Then, the final layers of the pre-trained model are removed, and the remaining layers and their weights are used as feature extraction models for the new data in the target domain and task. This is accomplished by fixing or freezing the weights and calculating the values of the final activations for the new target data, and then train the final layers of the model again. This process is shown in Figure 4.3.

1. **Examples of the technique in practice**

[Apply your technique to sample problems to demonstrate the technique. Clearly describe the application and evaluate performance. Illustrate both strengths and weaknesses. Comparisons to related techniques are strongly encouraged.]

1. **Summary**

[Summary of the method, its uses, as well as its strengths and weaknesses as compared to other similar techniques.]

1. **Roles**

[Since this is a team project, we want to know what your specific contribution was to this project. Provide detail on your individual role and how it contributed to the competition. Each team member should clearly articulate an individual role.]

**Yuan Feng**:

**Sebastián Soriano Pérez**:

**Vishaal Venkatesh**:

**Abhiraj Vinnakota**:

**Roderick Whang**:

1. **References** [no word limits]

[An alphabetical list of references cited in this work. A minimum of 10 are required. Consider using the Zotero citation manager for collecting and compiling your references.]