Survey of Transfer Learning with Relation to Network Traffic

Byron Barkhuizen
Repository at https://github.com/byronbark/IOTProject

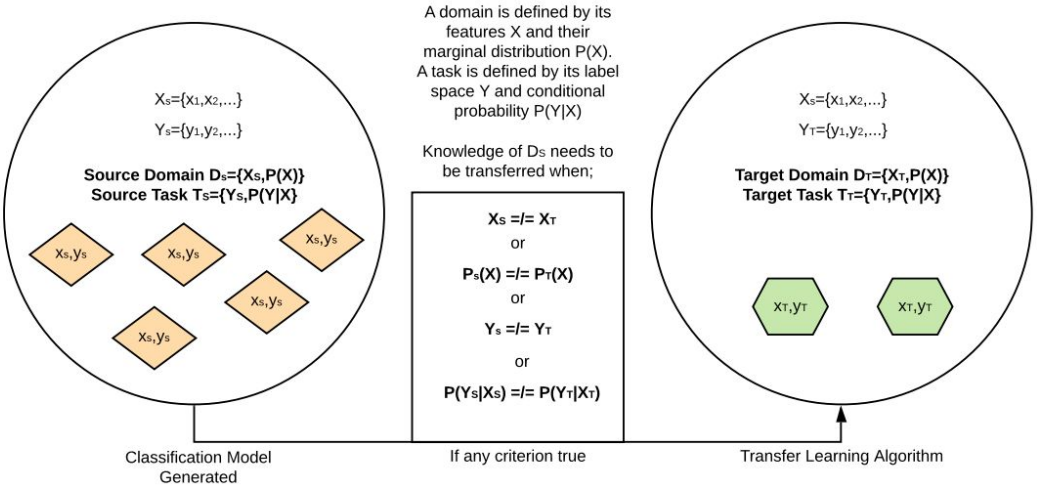
1 Motivation

Transfer learning is a developing research field in machine learning that seeks to 'learn to learn' in the same way that humans might approach a new task having some experience in a related one. This is akin to being able to recognize one type of object that is similar but not the same to another (a common example being fruits). The advantages of an algorithm that can mimic this ability are much lower computation requirements (time, power), lower complexity of approaching new problems, less feature engineering, and possibly higher accuracy depending on how well the knowledge transfers. Traditionally the transfer of a model to a new subset of tasks is based on an assumption that their characteristics such as features and marginal distributions are similar, however we know that this is not always the case and it can lead to losses in accuracy and performance if incorrectly assumed.

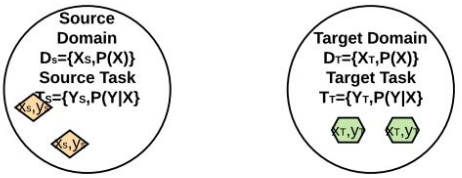
Particularly in the age of big data where there are large masses of rapid data being produced there may exist a need for transfer learning. Machine learning and deep learning is attempting to perform classification in close to real-time, but some changes can occur within even short timeframes. In this scenario there is a source and target domain created by this difference in temporal space, the source and target domain may also be defined by an object or group of objects. A model can be retrained in the new scenario however it will take a long time and it will require a large quantity of newly labelled data. Instead the research problem of transfer learning will try to find out how we can relate the source and target domain, and find a way in which to harmonize this knowledge. The ultimate goal is to eliminate or reduce the need to recollect data in the target environment in order to rebuild a deep learning model.

2 A Formal Definition of Transfer Learning

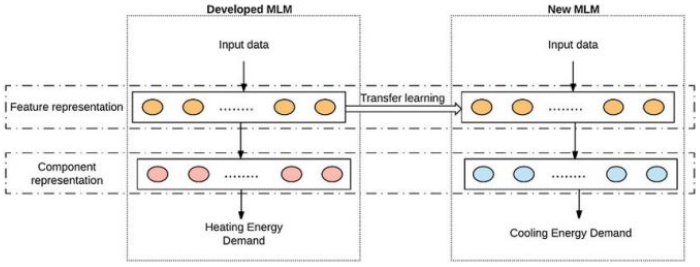
Transfer learning is the establishing of a relationship between a source domain and a target domain, along with source tasks and target tasks. DARPA (Defense Advanced Research Projects Agency) defines the mission of transfer learning as 'the ability of a system to recognize and apply knowledge and skills learned in a previous task to novel tasks. The existence or assumption that there is some similarity between the source and target is the theoretical basis for the application of transfer learning (1). The features within a source or target domain have some probability distribution. When these features distributions are the same then the new dataset can essentially just be classified in the same way that the original dataset was. If the source and target dataset do not conform to the same probability distribution then this cannot be done accurately. However this does not mean that a model needs to be completely re-trained and re-classified on the new dataset, there theoretically can exist some knowledge that can be transferred between the two datasets. A graphical representation of these definitions can be seen on the following page in figure 1.



Instance-Transfer



Feature-representation-transfer



Idea: use outputs of one or more layers of a network trained on a different task as generic feature detectors. Train a new shallow model on these features.

Assumes that $D_S = D_T$

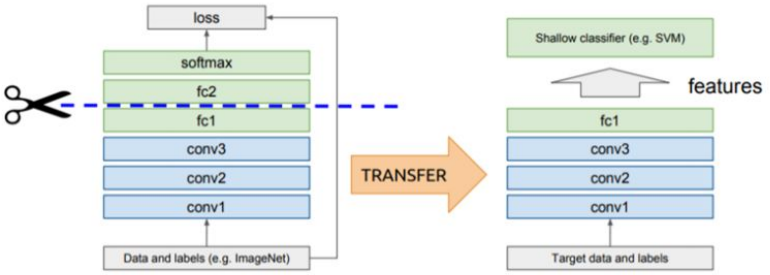


Figure 1) Fundamental concepts of transfer learning and some approaches

Some assumptions for the success of transfer learning are:

1) Low level features (shapes, edges, colours, sentence structure) are general and domain agnostic -> There always exist some basic shared knowledge
2) Domains or tasks between source and target are related enough so as not to cause negative transfer, in which case performance/accuracy would be decreased
3) Low quantity of labelled data available in target domain, or no labelled data at all in target domain
4) Source knowledge is labeled and rich

Transfer learning seeks to loosen ‘same distribution’ requirements through the application of transfer learning algorithms. These algorithms will seek to expand upon their similarities, finding the shared features according to some similarity conditions, typically in higher-dimension feature space. Once the shared features are established then knowledge transfer has occurred and there exists some information with the same distribution, allowing for meaningful continued classification algorithms to be used. The knowledge of the source data has been imposed upon the target data in a useful way.

3 Applying Transfer Learning

When we decide that we want to apply transfer learning we first have to ask a few questions. We need to know ‘what’ to transfer, ‘how’ to transfer, and ‘when’ to transfer. What to transfer tells us which parts of the knowledge from the source domain remain relevant and will be a useful piece of knowledge to keep. We do not want to keep knowledge that is specific to source or target tasks, instead we want to find the commonalities. Learning algorithms are then responsible for the ‘how’ to transfer. They operate on the basis that there is some knowledge to be shared and depending on the conditions of the source and target domains and tasks they will find a way to align their knowledge.

An important step in the transfer learning framework is to identify the setting in which the problem is situated. The main settings are inductive transfer learning, transductive transfer learning and unsupervised transfer learning. Unsupervised transfer learning is still a very new area of research.

Inductive transfer learning concerns when the target task is different to the source task, regardless of similarities in the source and target domain. It aims to improve the learning of the predictive function relating target domain and target task using the knowledge of the domain. The instance-transfer subsetting of inductive transfer learning looks to identify the parts of the source domain data that can be reused along with some labeled data in the target domain. TrAdaBoost is an example of a solution for this transfer learning setting. Feature representations is the second subsetting of transfer learning that aims at finding “good” feature representations in order to minimize the differences between domains and classification error. Common features are identified by an optimization problem unique to the type of source and target data.

Transductive transfer learning requires that some unlabeled target domain data must be available at training time. This setting concerns when the tasks to be performed are the same, but the source and target domains are different. An assumption of this setting is that

the predictive function between source domain and source tasks can be adapted in the target domain using some unlabeled data. Further subsettings involve when the feature spaces between the source and target domains are different, or when they are the same but their marginal distributions of the inputs are different.

4 Methods

Tradaboost operates on the assumption that some of the old data remains useful for a new task. This is the case when the target domain and source domain are somewhat related, even if the tasks are not the same. It allows the use of a small quantity of newly labeled data in a target domain, while leveraging old data from a related source domain, to construct a high quality classification model. This works even when the quantity of new data is not sufficient to train a model on its own and the old data is stale. The TLA (transfer learning algorithm) seeks to combine the old data and new data in the target domain and highlight a new set of labeled data called same-distribution data. The rest of the remaining old data is called diff-distribution data. The steps taken by the tradaboost algorithm can be simplified to the following;

1) Labeled training data (source) with same-distribution to test (target) data identified
2) Labeled training data (source) with diff-distribution to test (target) identified, these are plentiful but classifiers learned from them would be inaccurate for test data
3) AdaBoost boosting algorithm is applied to same-distribution new data set to build a base model
4) Most useful (lowest error) diff-distribution instances used as additional training data

5 Current Works

(2) demonstrates the application of unsupervised transfer learning. They aim to explore the capabilities of computer vision applications in order to learn behavioral states of vehicular traffic, with an unknown number of states. An example of the exploitation of image form for conversion of vehicle traffic information is shown in figure 2. The utilization of a popular pre-trained model called Inception Resnet-v2 by Google as a feature extractor yielded greatly reduced number of significant features with little to no training required. This demonstrates the use of a pre-trained neural network and transferring this to gain knowledge about the features of a target dataset. This is a very popular transfer learning approach that works by freezing the initial layer weights which are associated with basic features such as edges and gradients and replacing the final fully connected layer which deals with classification. This helps to identify common features and define a new feature map or the target dataset. The knowledge of the source domain is passed to the target domain. There are a few modifications to this approach which concern the extent to which existing layers in the pre-trained model are kept frozen, or they can be allowed to slightly modify weights in a fine-tuning process which is also very popular.

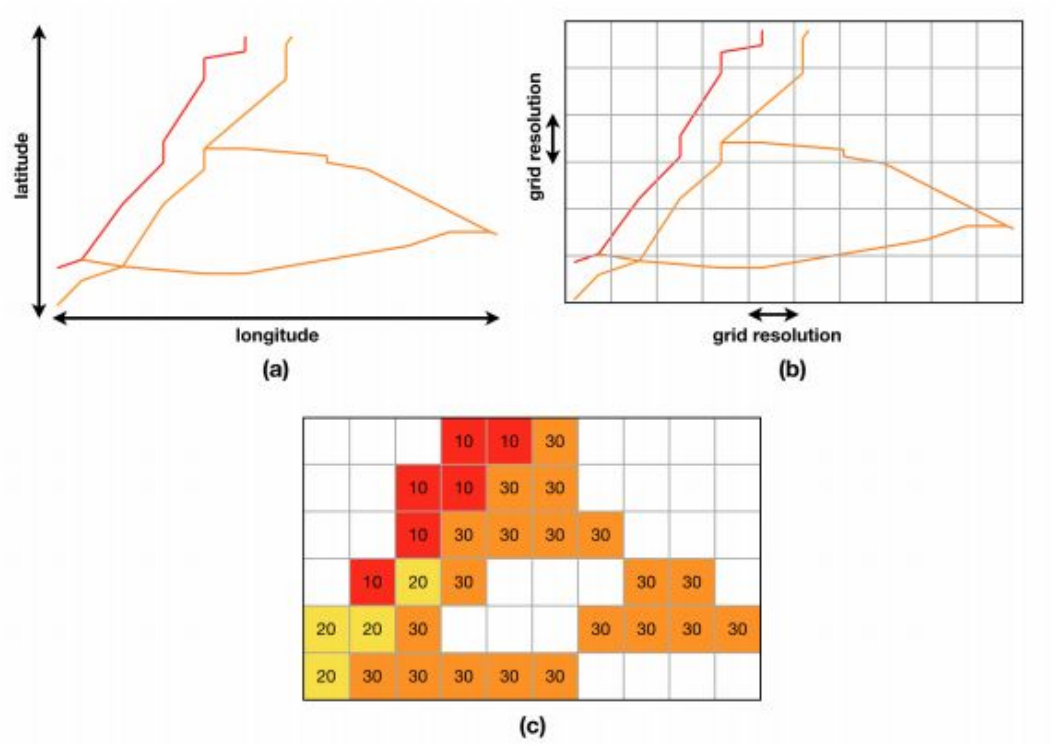
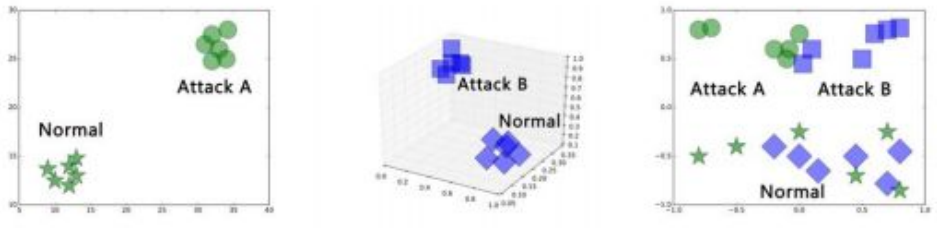


Figure 2) Traffic data converted into image form. A, b and c show different methods.

(3) takes a more advanced approach to aligning source and domain datasets by the definition of a feature-based transfer learning algorithm called HeTL. It observes different types of botnet attacks across networks, along with their flow level information in order to identify a signature. The experiment's aim was to transfer knowledge about known cyber attacks to the introduction of unknown attacks in order to classify them as malicious or not. A feature extraction process is undertaken through a CNN style approach. The feature spaces of the source and target domain are represented in a common latent space. An optimization function is applied to assess the distances between data of the source and target domains. This optimization function along with a distortion function attempts to keep the structure of the original data as much as possible, while maximizing similarities between the source and target domain by minimizing the difference in the latent space. This is done through a learning process using a gradient approach while observing the changes in the probability distribution of the source and domain data. HeTL performed extremely well in comparison to the situations where no transfer learning was used, and it performed better than other transfer learning algorithms. The two images below are from the paper and demonstrate the transformation of the source and target domain from their individual feature space to a shared latent space where some similar distributions can be observed. HeTL successfully finds a subspace that makes distributions of different attacks similar. The resulting modified feature-based transformations resulted in the second visual representation of attack vs. norma data where there is a clear decision boundary, greatly assisting classification, which would not be otherwise observed without transfer learning.



(a) Source domain (b) Target domain (c) Projected space

Fig. 4. Illustration of proposed feature space transformation concept.

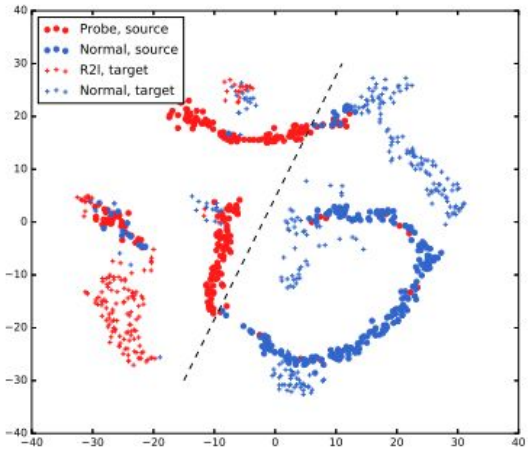


Figure 3) Observation of clear separator between malicious and non-malicious data points

(4) extends the fine-tuning approach to transfer learning. It uses the same data-to-image conversion idea of many other transfer learning approaches to represent botnet data. This approach relies on using a pre-trained model and maintaining some of the weights, it is one of the more straightforward applications of transfer learning with high success. An image representation of network traffic data is used alongside a pre-trained image network and comparisons are made between using no transfer learning and transferring some of the weights. The transfer learning approach performs extremely well compared to the no transfer learning approach, outperforming it by nearly 50%. The images generated for use in the pre-trained network are shown below. The transfer learning approach is measured at different levels of fine-tuning, and it performs at its best with only a few first layers freezed. When too many layers become frozen the features being explored are source domain specific and it does not perform well, going down to 33.51% accuracy. The testing phase for one packet data is extremely quick and it shows promise of this type of promise being utilized in real time for intrusion detection.

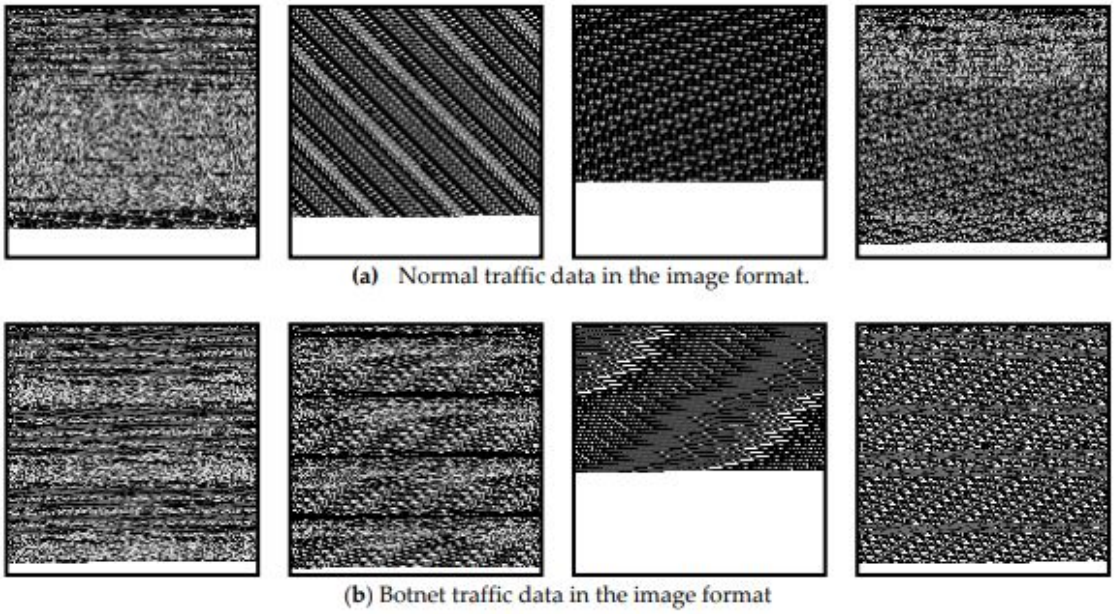


Figure 5)

(1) is another demonstration of a feature-based transfer learning algorithm. Similar to (3) the approach seeks to find similarities between the source and target domain feature distributions, or it seeks to 'force' them. The source and target features are mapped to a high dimensional space in order to assess their similarity by a method such as Euclidean distance. This work extends the feature mapping component of transfer learning by using a convolutional neural network to classify the newly identified features as source or target. This optimizes the parameters of the mapper so that after it is finished the traffic data in the target domain is more similar to the source domain. The output of the network is binary, and features will be classified as being part of the source domain or not. A loss function is defined by the similarity between the two, and they seek to minimize this. When it is minimized it means that the maximum possible similarity between the two domains has been reached. The method showed impactful results, effectively being able to transfer knowledge between the source and target domains in a short time (1 epoch).

6 IoT Identification Recommendation

The transforming of network traffic data to image, as supported by (4) and (5), as an application agnostic approach to network flow analysis and therefore an approach for uniquely and accurately identifying IoT devices has high accuracy and it is something that has an abundance of resources to build upon. Python with Keras is highly performant for computer vision problems and there is a large amount of documentation as well as research papers available on the manipulation of image data for deep learning. The abstraction away from textual information taken from our PCAP files allows a more adaptable approach, better suited for transfer learning.

The issue becomes whether the image representation of the pcap file, similar to (5), can be enriched somehow in order to show more unique information about the network flow from an IoT device. This would allow more accurate IoT device identification. (6) demonstrates the

relevance of using other flow level information apart from the packet data. Semantic relationships are created between statistical attributes such as activity cycles, port numbers and cipher suites to find features that best represent unique IoT devices and therefore act as a sort of signature. The issue with this is that many of these features can theoretically be altered by vendors and be unreliable. Incorporating these features also requires a domain expert for feature engineering, and the architecture involved is more complex than some other approaches. Nevertheless it would be useful to find a meaningful way to include some of these important features alongside the image of network traffic flow.

Transfer learning should be implemented between the source and target domains as the representation of flow data as an image allows a better understanding of the features and therefore will improve the feature extraction phase. Once the extracted features are understood then a dataset from a target domain, such as a new environment with new iot devices, the probability distributions of features does not have to be identical. A feature-based approach between the two domain datasets can be used to normalize some key features and align their feature distributions in such a way that it can be easily classified. The knowledge of the known IoT devices can be transferred to a new network environment with some minor adjustments being made through a transfer learning algorithm.

References

(1) Xiong P., Cui B., Cheng Z. (2021) Anomaly Network Traffic Detection Based on Deep Transfer Learning. In: Barolli L., Poniszewska-Maranda A., Park H. (eds) Innovative Mobile and Internet Services in Ubiquitous Computing. IMIS 2020. Advances in Intelligent Systems and Computing, vol 1195. Springer, Cham

(2) P. Krishnakumari, A. Perotti, V. Pinto, O. Cats and H. van Lint, "Understanding Network Traffic States using Transfer Learning," *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, 2018, pp. 1396-1401, doi: 10.1109/ITSC.2018.8569450.

(3) J. Zhao, S. Shetty and J. W. Pan, "Feature-based transfer learning for network security," *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*, Baltimore, MD, 2017, pp. 17-22, doi: 10.1109/MILCOM.2017.8170749.

(4) Taheri, S., Salem, M., & Yuan, J. (2018). Leveraging Image Representation of Network Traffic Data and Transfer Learning in Botnet Detection. *Big Data And Cognitive Computing*, *2*(4), 37. doi: 10.3390/bdcc2040037

(5) Kotak, Jaidip & Elovici, Yuval. (2020). IoT Device Identification Using Deep Learning.

(6) Sivanathan, A., Gharakheili, H., Loi, F., Radford, A., Wijenayake, C., Vishwanath, A., & Sivaraman, V. (2019). Classifying IoT Devices in Smart Environments Using Network Traffic Characteristics. *IEEE Transactions On Mobile Computing*, *18*(8), 1745-1759. doi: 10.1109/tmc.2018.2866249