List of Papers

1. https://arxiv.org/pdf/2205.10070.pdf On the Prediction Instability of Graph Neural Networks **Satvik**
    - Graph neural networks are inherently unstable
    - lots of factors influence instability:
        - Dataset
        - GPU causes instability, less than data order shuffling
        - Churn and error rate are correlated
        - L2 regularization increases stability slightly
        - Dropout generally increases churn, but you can finely tune it to work well
        - Wider models are more stable
        - Depth increases stability

2. https://arxiv.org/abs/1904.04755 Hypothesis Set stability and generation **Satvik**
    - Theoretical bounds for distillation covering maths around hypothesis sets, difficult to understand, not very useful

3. http://proceedings.mlr.press/v98/cotter19a/cotter19a.pdf Two-Player Games for Efficient Non-Convex Constrained Optimization **Satvik**

    - Difficult paper, very mathematical
    - Deals with optimizing under constraint and use an alternate lagrangian to make the surface smoother

4. https://papers.nips.cc/paper/2018/file/7180cffd6a8e829dacfc2a31b3f72ece-Paper.pdf To Trust Or Not To Trust A Classifier **Satvik**

    - Trust score is used to determine how confident a model is about a prediction
    - Using predicted probabilities as a proxy for confidence doesn't work out that well
    - Trust score is defined as distance of another sample of different class / distance of another sample of same class, in a intermediate layer vector-space

5. https://arxiv.org/pdf/1804.03235.pdf LARGE SCALE DISTRIBUTED NEURAL NETWORK

TRAINING THROUGH ONLINE DISTILLATION **Satvik**

    - Use to speed up training time
    - Multiple models are trained parallely, each using a subset of data
    - Loss functions include a term to measure divergence between model's prediction vs other models prediction on the same point
    - Can help reduce churn as ensemble models are more consitent

6. https://papers.nips.cc/paper/2016/file/dc4c44f624d600aa568390f1f1104aa0-Paper.pdf Satisfying Real-world Goals with Dataset Constraints **Dauren**

Main idea: in a one particular task we might need to satisfy different goals including high accuracy, precision & recall metrics, fairness, churn etc. Sometimes this goals are not aligned with each other. The authors propose some crazy math optimization to find optimal solution to satisfy all goals using different datasets with different importance weights.

7. https://arxiv.org/abs/1604.04326 Improving the Robustness of Deep Neural Networks via Stability Training **Dauren**

Main idea: the idea of stability is somewhat close but still different than prediction churn. Due to some techinal reasons (e.g. data compression, different data format) training data may confuse NN which in turn will lead to unstable predictions (small perturbations in the input may lead to drastically different predictions and human eyes may not be able to see the difference. Some examples are: frame by frame instability during training on the video data, finding nearly duplicates). Training stability is helpful against adversarial attacks.

Authors used noise adding, data augmentation techinques to reduce sensitivity to small changes in the input space which increases robustness of NN predictions.

8. https://arxiv.org/abs/1809.04198 Optimization with non-differentiable constraints with applications on fairness, recall, churn, and other goals. **Dauren**

Main idea: this paper is 58 pages long and math heavy. I'll come back to this one during this week.

9. https://arxiv.org/abs/1906.02629 When does label smoothing help? **Dauren**

Main idea: Label smoothing helps calibrating NN, reduces over-confidence. Doesn't work with knowledge distillation i.e. if a teacher model has been trained using label smoothing then training a student model won't be as effective as it could be without label smoothing.

Interesiting article on model calibration:

https://www.unofficialgoogledatascience.com/2021/04/why-model-calibration-matters-and-how.html

10. https://arxiv.org/abs/2004.12289 **Dauren**

Main idea: Filter (authors propose simply to remove data points which are not in agreement with knn) questionable data points using knn (k=500 is a common choice when data set size is

large) on the intermediate data representation. Can be used as signals for human annotators when knn doesn't match current label.

The result of the experiment is that knn method helps improving prediction accuracy on the test set. Label smoothing paper use this idea to reduce churn (but they don't use corrupted data set and don't remove data points). *I like their experiment design.*

11. https://arxiv.org/abs/2105.13093 Towards understanding knowledge distillation **Himangshu**

The paper is about current intuitive and theoretical understanding of distillation, shows detailed proof on how distillation works, and gives a better view on linear distillation.

- Describes what to do when X happens. For instance: When few data points are available, approaches like gradient descent update can be utilized.
- **Identifies three key factors that explain the success of distillation**: *data geometry* – geometric properties of the data distribution, in particular class separation, directly influence the convergence speed of the student's risk; *optimization bias* – even though the distillation objective can have many optima, gradient descent optimization is guaranteed to find a particu- larly favorable one; and *strong monotonicity* – increasing the training set always decreases the risk of the student classifier.

This paper goes on depth on linear distillation:

To understand linear distillation properly, reading paper below is super useful:

https://www.arxiv-vanity.com/papers/1906.05431/

"Linear Distillation Learning (LDL) is a simple remedy to improve the performance of linear networks through distillation. In deep learning models, distillation often allows the smaller/shallow network to mimic the larger models in a much more accurate way, while a network of the same size trained on the one-hot targets can't achieve comparable results to the cumbersome model".

12. https://arxiv.org/abs/1512.00567 Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826, 2016. **Himangshu**

- Gives a brief overview on **several design principles to scale up convolutional networks and studies them in the context of the Inception architecture**.

- Good paper to get understanding of design principles of CNN and approaches but to be noted every project has their own way of utilizing CNN for example referring to paper in terms of filters and layers.
- Currently reading again; too complex to understand in once or twice reading.

13. https://arxiv.org/abs/1905.10964 Combating Label Noise in Deep Learning Using Abstention **Himangshu**

- This paper is excellent if we are trying to use **deep learning and have different types of noises while training deep neural networks for classification**.
- The general understanding of this paper is that it introduces a loss function that permits abstention during training thereby allowing the DNN to abstain on confusing samples while continuing to learn and improve classification performance on the non-abstained samples.
- Its talks about using abstention where it means training with abstention enables representation learning for features that are associated with unreliable labels

In short, it explains how an **abstention-based approach** can be used as an effective *data cleaner* when training data contains arbitrary (or unstructured) label noise.

Himangshu

14. https://papers.nips.cc/paper/2000/file/49ad23d1ec9fa4bd8d77d02681df5cfa-Paper.pdf

Himangshu

Algorithmic Stability and Generalization Performance

Good paper if we follow PAC style frame work, this paper provides mathematical explanation.

- This paper focuses on the Regression method which doesn't talk much about classification.
- Focuses on **PAC approach** , which is a framework for analyzing the generalization error (a measure of how accurately an algorithm is able to predict outcome values for previously unseen data.) of a learning algorithm in terms of its error on a training set and some measure of complexity.
- Utilizes leave one out method.
- Talks about regularization networks with SVM, where it works well.

ABSTRACT: We present a novel way of obtaining PAC-style bounds on the generalization error of learning algorithms, explicitly using their stability properties. A stable learner is one for which

the learned solution does not change much with small changes in the training set. The bounds we obtain do not depend on any measure of the complexity of the hypothesis space (e.g. VC dimension) but rather depend on how the learning algorithm searches this space, and can thus be applied even when the VC dimension is infinite. We demonstrate that regularization networks possess the required stability property and apply our method to obtain new bounds on their generalization performance

15. https://www.jmlr.org/papers/volume16/vapnik15b/vapnik15b.pdf

Tigran

Learning Using Privileged Information: Similarity Control and Knowledge Transfer

This paper describes a new paradigm of machine learning, in which Intelligent Teacher is involved. During training stage, Intelligent Teacher provides Student with information that contains, along with classification of each example, additional privileged information (for example, explanation) of this example. The paper describes two mechanisms that can be used for significantly accelerating the speed of Student's learning using privileged information: (1) correction of Student's concepts of similarity between examples, and (2) direct Teacher-Student knowledge transfer.

16. https://arxiv.org/pdf/2102.03349.pdf

Tigran

On the Reproducibility of Neural Network Predictions

Standard training techniques for neural networks involve multiple sources of randomness, e.g., initialization, mini-batch ordering and in some cases data augmentation. Given that neural networks are heavily over-parameterized in practice, such randomness can cause churn– for the same input, disagreements between predictions of the two models independently trained by the same algorithm, contributing to the 'reproducibility challenges' in modern machine learning. In this paper, we study this problem of churn, identify factors that cause it, and propose two simple means of mitigating it. We first demonstrate that churn is indeed an issue, even for standard image classification tasks (CIFAR and ImageNet), and study the role of the different sources of training randomness that cause churn. By analyzing the relationship between churn and prediction confidences, we pursue an approach with two components for churn reduction. First, we propose using minimum entropy regularizers to increase prediction confidences. Second, we present a novel variant of co-distillation approach [Anil et al., 2018] to increase model agreement and reduce churn. We present empirical results showing the effectiveness of both techniques in reducing churn while improving the accuracy of the underlying model.

17. https://storage.googleapis.com/pub-tools-public-publication-data/pdf/43146.pdf

Tigran

Machine Learning: The High-Interest Credit Card of Technical Debt

Machine learning offers a fantastically powerful toolkit for building complex systems quickly. This paper argues that it is dangerous to think of these quick wins as coming for free. Using the framework of technical debt, we note that it is remarkably easy to incur massive ongoing maintenance costs at the system level when applying machine learning. The goal of this paper is highlight several machine learning specific risk factors and design patterns to be avoided or refactored where possible. These include boundary erosion, entanglement, hidden feedback loops, undeclared consumers, data dependencies, changes in the external world, and a variety of system-level anti-patterns.

18.https://www.mandiant.com/resources/blog/churning-out-machine-learning-models-handling-changes-in-model-predictions

Tigran

Churning Out Machine Learning Models: Handling Changes in Model Predictions

Machine learning (ML) is playing an increasingly important role in cyber security. Here at FireEye, we employ ML for a variety of tasks such as: antivirus, malicious PowerShell detection, and correlating threat actor behavior. While many people think that a data scientist's job is finished when a model is built, the truth is that cyber threats constantly change and so must our models. The initial training is only the start of the process and ML model maintenance creates a large amount of technical debt. Google provides a helpful introduction to this topic in their paper "Machine Learning: The High-Interest Credit Card of Technical Debt." A key concept from the paper is the principle of *CACE*: *change anything, change everything*. Because ML models deliberately find nonlinear dependencies between input data, small changes in our data can create cascading effects on model accuracy and downstream systems that consume those model predictions. This creates an inherent conflict in cyber security modeling: (1) we need to update models over time to adjust to current threats and (2) changing models can lead to unpredictable outcomes that we need to mitigate.

19 https://hal.archives-ouvertes.fr/hal-01418129/file/article.pdf

Tigran

Incremental learning algorithms and applications

Incremental learning refers to learning from streaming data, which arrive over time, with limited memory resources and, ideally, without sacrificing model accuracy. This setting fits different application scenarios where lifelong learning is relevant, e.g. due to changing environments, and it offers an elegant scheme for big data processing by means of its sequential treatment. In this contribution, we formalise the concept of incremental learning, we discuss particular challenges which arise in this setting, and we give an overview about popular approaches, its theoretical foundations, and applications which emerged in the last years.

Literature Review

- **Modifying labels**
  - **Knowledge Distillation** - Uses the base (teacher) model's predictions as a convex combination with true labels
  - **Adaptive Label Smoothing** - Uses locally smoothed labels based on KNN on logit layer
  - **Anchor Method** - Consists of two parts: stabilization (RCP and Diplopial operators) and randomization to mimic future changes

- **Modifying dataset**

  For noisy labels:
  - **KNN elimination:** Uses filter approach.
  - **Deep Learning Abstention based approach:** It introduces a loss function that
  - permits abstention during training thereby allowing the DNN to abstain on confusing samples while continuing to learn and improve classification performance on the non-abstained samples.

  For stability :
  - **L2 regularization**
  - **Augmentation:** Strategy to improve label stability.

  Trust score : A new effective approach to when a classifier's predictions should and should not be trusted.

- **Changing training procedure**
  - Using co-distillation (ensemble of models learning off of each other), each working on a subset of the data
  - Entropy regularizer (to increase the prediction confidence)
  - Symmetric KL divergence regularizer (to enhance the prediction confidence)
  - Computing platforms (GPU,TPU)
  - Sources of randomness
    - Model initialization
    - Mini-batch ordering
    - Dataset augmentation