PITCH N1

Use deep reinforcement learning to tune the hyperparameters (learning rate, lambda – regularization parameter, number of layers, number of units in each layer, different activation functions) of a Neural Network. The overall cost function of RL agent will include the metrics such as accuracy of the NN (or F1 score) on training and validation sets, time taken to learn, the measures of over/underfitting. This network would be trained on different types of problems.

JOMAA, H. S., GRABOCKA, J., and SCHMIDT-THIEME, L. 2019. Hyp-rl: Hyperparameter optimization by reinforcement learning. *arXiv preprint arXiv:1906.11527*.

- This paper is directly related to my research in that it formulizes hyperparameter tuning as a reinforcement learning problem. Their algorithm involves a policy based on Q-learning (Hyp-RL). I'm using policy gradient to navigate high-dimensional hyperparameter spaces and arrive at the best configuration. Their loss function reflects the difference between the accuracy gained on training and validation sets. So, the goal is to minimize the generalization error. This way they are avoiding overfitting, but the issue of underfitting still remains. My idea is to try to minimize the error rate on the training set as well as the difference of error between training and validation sets. I would probably use the same hyperparameters for configuration including number of neurons, number of hidden units, number of epochs, different regularization methods.

SHAHRIARI, B., SWERSKY, K., WANG, Z., ADAMS, R. P., and DE FREITAS, N. 2015. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, *104*(1), 148-175.

- This paper describes a method called Bayesian Optimization. One of the applications of this algorithm that the paper talks is solving the issue of finding the optimal hyperparameters for a neural network. As the paper mentions, the framework of Bayesian optimization is data efficient and is useful in situations when the evaluation of a function is costly and the derivatives with respect to this function is not-accessible. This paper is related to my research as it presents another automated method for finding the hyperparameters and I could use the results presented in this paper to compare the results to my approach.

James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *The Journal of Machine Learning Research* 13, 1 (2012), 281–305

- This paper presents yet another method for finding the optimal hyperparameters for a neural network. The process that's described in this paper automates the search of hyperparameters by randomly choosing the values for hyperparameters. There are a lot of comparisons drawn with other methods for solving this problem – grid search and manual search. The objective of the search is defined in the same way as in the paper by H. Jomaa et al. - minimizing the generalization error. Random search has successfully been used in practice and proved to be more robust than other methods. This provides a baseline that I can compare my results to. The paper also proposes that random search could be improved by automating what the manual search does – that is sequentially optimizing the hyperparameters with some sense of logic of how each hyperparameter affects the final performance of the network. This also suggests that my method of using Reinforcement Learning has the potential for learning how to navigate the hyperparameter space by learning the logic of sequential

Steven R. Young, Derek C Rose, Thomas P. Karnowski, Seung-Hwan Lim, and Robert M Patton. 2015. Optimizing deep learning hyper-parameters through an evolutionary algorithm. In *Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments*. 1–5.

- This paper proposes evolutionary algorithm as a method for optimizing the hyper-parameters in the neural network. The paper compares genetic algorithms to other popular methods for hyper-parameter search – grid search, manual search and random search. I can use the results to draw the comparisons between my approach and the one presented in this paper. One downside of this approach is that it is computationally very expensive. The paper also mentions that for different network architectures the same hyper-parameters have different effect. This suggests that the hyper-parameter space is hard to navigate and I might need to employ different exploration techniques to ensure that my algorithm doesn't get stuck in the local optima.

PITCH N2

For this idea, I'm using the game of Pong (ATARI) as a test environment. My plan is to introduce a specific pipeline in training the AI agent to play the game. Instead of directly using the Policy Gradients, I will train the agent to guess the next frames in the game. First, I will use RNN to learn (approximate) the transition function in an unknown environment. The transition function, modeled by a Recurrent Neural Network, will take some previous n states (in raw pixel form) and agent's action, and output the state representation that corresponds to the future state of the environment. The intuition behind this is that, the agent will first learn the 'laws of physics' of a certain environment (exploration) and this will help the agent learn how to play the game more efficiently. After learning the weights of the transition function, I will implement Reinforcement Learning algorithm (Policy Gradients) that reuses the learned weights (transfer learning) and train this deep neural network by letting in play a number of games and learn from experience.

ZHANG, A., SATIJA, H., and PINEAU, J. 2018. Decoupling dynamics and reward for transfer learning. *arXiv preprint arXiv:1804.10689*.

- The idea of this paper is very similar to my approach. The basic idea is that learning is decomposed into two complementary objectives, one for learning the state dynamics model and the other for learning the reward function. This paper makes a few claims about why decoupling dynamics and learning the reward function makes sense. The first reason is that such decoupling enables adaptation to new tasks. Secondly, the model enables forward search and planning. My approach is different from this one because I'm using transfer learning from the dynamic model to the reward function. Therefore, the advantage of my model is not the adaptation to the new tasks but decreasing the complexity of learning the reward function. Also, my model enables forward search and planning but in not by directly being given the future frame, but rather by learning the features of the state dynamics and then deciding which of these features are relevant in learning to maximize the reward function. My objective is to decrease the training time but because I'm using transfer learning instead of having different components, I'm also letting the dynamic model's weights to change instead of keeping them fixed.

WEBER, T., RACANIERE, S., REICHERT, D. P., BUESING, L., GUEZ, A., REZENDE, D. J., ... and PASCANU, R. 2017. Imagination-augmented agents for deep reinforcement learning. *arXiv preprint arXiv:1707.06203*.

- This paper proposes the use of the internal model that reasons about the future. This is closely related to my research idea and provides the theoretical framework for why it makes sense for the agent to be able to predict the future frames. As the paper claims, the main motivation for abandoning the model-free approach is the amount of training data needed before the agent before its able to produce good results in terms of maximizing the reward function. The agent learns to seek the positive outcomes in the future and avoids the poor decisions. Internal models can also be alleviating the problem of credit assignment. This paper also touches on the fact that mental simulations play a big role in animal and human learning. The paper reports better performance compared to model-free approaches and more importantly achieving these results with less data.

Haiyan Yin, Jianda Chen, and Sinno Jialin Pan. 2017. Hashing over predicted future frames for informed exploration of deep reinforcement learning. *arXivpreprint arXiv:1707.00524* (2017).

- The research presented in this paper also decouples the dynamics and reward modules so that the agent learns to approximate the transition function separately from learning how to act. However, the main concentration of this paper is the informed exploration of the environment where the agent chooses to visit the states that it hasn't explored yet so that the exploration is more efficient. Another difference is that the deep prediction model is trained to predict the frames based on the action and the current state. This way the agent can only learn how its actions are changing the states but is less informed about how the environment changes naturally, without any action from the agent. The part of this research that is relevant to my idea is how they are training the deep prediction model and what additional methods they are using to represent the states.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy HCampbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, et al. 2019. Model-based reinforcement learning for atari. *arXivpreprint arXiv:1903.00374* (2019).

- This paper is directly related to my research. The basic idea is introducing the prediction model. It is based on the same motivation that I had – understanding the intuitive physics enables the agent to make more informed decisions and this way it takes less time for the agent to learn to play the game. In this paper the researchers describe a new architecture called Simulated Policy Learning which is a model-based RL algorithm. The prediction module is trained using supervised learning. The results that they are reporting are state-of-the-art both in terms of the agent's ability to play and the time taken for training. The difference between my approach and this one is that, they are updating both modules (one concentrated on prediction, and the other for learning the policy) at the same time based on agent's interactions with the world, instead of training these modules separately.

PITCH N3

I will train the CNN to be able to verify, given the images of handwritten text, if two handwritings belong to the same person. In order to generate more labeled data, I will use a dataset with images of handwritten texts and break up each image into the windows containing a few words. I will assume that each word written on a single image belongs to one person.

XING, L., and QIAO, Y. 2016. Deepwriter: A multi-stream deep CNN for text-independent writer identification. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 584-589

- This paper is directly related to my research as it is tackling the problem of writer identification using CNN. The paper points out some useful distinctions between on-line and off-line identification, as well as text-dependent and text-independent methods. In addition to that, they are using some data augmentation methods that might be useful for my approach too. This paper also mentions the datasets that include the image with written text accompanied with labels corresponding to the writers to whom the handwriting belongs. If I use the same datasets, I will be able to compare my results with the ones presented in this paper. Since, I'm also taking a text-independent approach, I might need to implement the patch scanning strategy presented in this paper as a preprocessing step. The goal of the research in this paper is to build a classifier that identifies the writers of the handwritten texts instead of verifying if the two handwritings come from the same person. They call the network DeepWriter, which takes multiple local regions as input and is trained with softmax loss on identification. The main distinction between this one and my research is that I will be learning embeddings as the way to measure the similarity between two handwritings.

HAFEMANN, L. G., SABOURIN, R., and OLIVIERA, L. S. 2017. Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recognition*, *70*, 163-176.

- Handwriting verification could be seen as a generalization of signature verification, so the methods presented in this paper are closely related to my research. However, the main goal of the research presented in this paper is to build a classifier good enough to be able to distinguish between genuine and skilled forgeries, while I'm trying to find a classifier that discriminates between the users in the development set. This paper approaches the problem of signature verification with a two-phased approach – writer-independent feature learning followed by writer-dependent classification. However, one downside of this method is that you need to re-train the writer-dependent classifier every time you need to verify a signature from a different person. This paper proposes that CNNs are the most effective for feature learning of the signatures and that manually designing the feature extractors doesn't yield good results. The dataset of skilled forgeries of the handwritings is not available to my knowledge, so I am simply aiming to create a classifier that finds the linear separator the handwritings in the dataset.

SCHROFF, F., KALENICHENKO, D., and PHILBIN, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815-823.

- The method is based on learning Euclidean embeddings for images using DCNN. The output of the network is a vector such the squared L2 distances in the embedding space directly correspond to face similarity: faces of the same person have small distances and faces of distinct people have large distances. This paper has been very influential and by the time it was published, it achieved the state-of-the-art results. The motivation for learning the embeddings instead of building a classifier is that we don't have to retrain the network for new faces in the database. I'm formulating the problem of handwriting verification analogously to the problem of face verification. The difference is the input – either the image of a face or an image of handwriting. The paper defines triplet loss function that enables the network learn the embeddings and I'm planning to use the same loss. Another relevant topic from this paper is the problem of choosing good triplets (anchor, positive example, negative example). It turns out, according to this paper, that choosing which triplets to use is very important in making the learning converge faster and leads to more accurate results.

SIMARD, P. Y., STEINKRAUS, D., and PLATT, J. C. 2003. Best practices for convolutional neural networks applied to visual document analysis. In *Icdar* (Vol. 3, No. 2003).

- This paper claims that Convolutional Neural networks have proved to be best for handwriting recognition tasks. This suggests that my approach has a potential and reasoning behind it – convnets are the state-of-the-art feature extractors for visual document analysis. The paper talks about different choices of kernel and how it affects the performance. I will use some of the tips presented in this paper for tuning my CNN. This paper is also talking about augmenting the datasets through elastic distortions. However, I'm not sure if this will help my research because after introducing the elastic distortions to the raw images, some valuable information containing cues about the specific writing style of a person might get lost.

Vincent Christlein and Andreas Maier. 2018. Encoding CNN activations for writer recognition. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS*). IEEE, 169–174.

- This paper is aimed to solve the same problem of off-line writer recognition as I am. The datasets mentioned in this paper might be useful for my research so I should look them. The method for writer identification that is described in this paper is the use of CNNs to produce the the encodings for the handwritten images. However, the difference with my approach is that they are training the CNN on classification with softmax loss and then extracting the penultimate layer that serves as a feature vector. These encodings of the handwritten text images are then aggregated into VLAD encoding and compared to encodings to find the similarity measure. The paper also experiments between triangulation embedding and VLAD encoding. Even though idea follows the same strategy, I'm using the unified embeddings that have proved to be more successful for face identification problem and I'm hoping to arrive at better results. I'm training the network directly to produce better encodings. Another question that they are asking that is relevant to my research is if max pooling is better than sum pooling. I might build on that and choose to use the pooling method that proved to be perform better in their research.

Stefan Fiel and Robert Sablatnig. 2015. Writer identification and retrieval using a convolutional neural network. In *International Conference on Computer Analysis of Images and Patterns*. Springer, 26–37.

- This paper also uses the feature encodings, however, in their approach they are taking it as a classification task and then cutting off the last layer. (similar to Vincent Christlein and Andreas Maier's approach) Facenet paper argues that this is not the most efficient way to go and that it's better to train the network to make the representations better. The paper also talks about the challenges of writer identification that I can cite in my paper. These challenges include changing the media for writing, the age of the writer, distractions during the writing process. They also employ some preprocessing steps that I might also use. These are the binarization of the image, sliding window approach for extracting the square patches and other methods for cleaning images from background noise.

Dominik Scherer, Andreas Müller, and Sven Behnke. 2010. Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks*. Springer, 92–101.

- This paper explains the theory behind the pooling operations. I'm using these operations in extracting the writer specific features so I need some theoretical framework for evaluating how pooling affects this process. Pooling operations combine low-level features that are topologically closer together and it can have a big influence on the robustness of the model as well as the mode's ability to detect the useful features in handwriting. The paper claims that maximum pooling operation performs better than other pooling methods. I can test if that hypothesis holds for the topic of my research.

Manuel Keglevic, Stefan Fiel, and Robert Sablatnig. 2018. Learning features for writer retrieval and identification using triplet CNNs. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE, 211–216.

- This paper presents the method of using Triplet CNNs to tackle the writer verification/identification problem (if the general encodings are produced, then it can be used for both verification and identification). This method of using triplet architecture is exactly what I'm planning to use, however, there is a lot of room for improvement. First of all, the results that this paper presents were not as good as the results achieved by state-of-the-art method. There are a lot of ways I could experiment with different preprocessing steps, different inputs and datasets to build upon the work that's presented in this paper. The primary experiment that I will conduct is related to what we feed the CNN for producing the encodings. In this paper, the researchers created encodings for images patches (32x32 pixels) and then these encodings were aggregated into VLAD encodings. Instead, I propose to apply the similar method for the entire line of the handwriting or the entire image of handwriting by creating square images from different patches in the image.

Youbao Tang and Xiangqian Wu. 2016. Text-independent writer identification via CNN features and joint Bayesian. In *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. IEEE, 566–571.

- This paper presents the same method for text-independent writer identification that was used by Vincent Christlein and Andreas Maier, Stefan Fiel and Robert Sablatnig. They report the state-of-the-art results by the time the paper was published and their results still remain among the best in the field. In this paper, they are trying to learn the encodings based on the last layer of the CNN trained to classify the images based on the writer. The key difference in their technique, compared to the papers that use the same approach, is that they are not simply breaking up the image of handwriting into the patches but instead that are generating the encodings for the artificial document images that are created by combining the random patches from the original image. They are also using data augmentation methods. I think that this is the right way to go in order to detect the features of a person's handwriting instead of concentrating on the encodings of small patches that might not even have enough information necessary to distinguish between individual styles of handwriting.

Weixin Yang, Lianwen Jin, and Manfei Liu. 2016. Deepwriterid: An end-to-end online text-independent writer identification system. *IEEE Intelligent Systems* 31,2 (2016), 45–53.

- This is the only paper that I read related to online writer identification. Even though online handwriting captures much more information about the writer's handwriting idiosyncrasies, the research in this area is less developed because of the scarcity of the data. To alleviate this problem, the paper suggests the use of data augmentation technique that they call DropSegment which is inspired by the Dropout regularization technique. The idea behind DropSegment is to produce the variations of handwriting images by randomly getting rid of different strokes in the handwriting. They are talking about a method used for separating the characters into segments. This is relevant to my research because I'm trying to find new ways for data augmentation and preprocessing and I might employ this technique.