Do Auxiliary Task-based Network Regularization benefits depend on size of Training Dataset?

Namusale Chama

***Abstract*—Auxiliary tasks are used for network regularization to counter overfitting. Even when they are successful in aiding achieve the network main task(s), it does not seem clear whether this success depends on the amount of data used in training or not. As such this paper works to investigate whether using a small data affects auxiliary tasks' aiding in network regularization. Preliminary work to address this issue has been to inspect datasets to be used. Initial obtained results show that there are some patterns or features in the datasets. Because dataset inspection is only a small part of the methodology, future work will cover the rest of the tasks for this work.**

***Index Terms*—Auxiliary Tasks, Network Regularization, Auto encoders**

I. Introduction

The rise of various machine learning techniques has enabled different problems ranging from classification to clustering to be solved. These techniques can be decision trees, neural networks and naïve Bayes classifier to mention but a few. However, there seem to be a challenge for these learners when they require training sets to perform a task. For effective learning, they may require considerable large training sets and using a big dataset to train a network can present two problems. Firstly, training becomes more time consuming as the dataset grows. Secondly, having a large dataset may not always be feasible. Use of an insufficient or limited dataset may result into a learner to suffer from overfitting. This implies that the learner fails to generalize and away from the training data, the learner performance is likely to be suboptimal.

To counter overfitting, network regularization is used. Regularization techniques are methods aimed at helping a network generalize when learning. A number of regularization approaches are in place to counter the overfitting problem of networks [1][2]. Some of these are dropout, early stopping and using auxiliary tasks among others. In terms of dropout mechanism, [1] reports that drop out in a neural network is synonymous to adding noise to the network. The authors speculate that noise addition enables the network to learn more on the principle features mechanism. Equally, auxiliary tasks have been used for network regularization [2][3][4]. Auxiliary tasks have been employed in networks to provide a rich environment for the learner to use as it performs its main task(s).

Although considerable work has already been done in network regularization using axillary tasks, it does not seem clear to what extent network generalization using auxiliary tasks success depends on the size of the data set to be used for training. This paper aims to investigate whether small datasets can benefit from network regularization using auxiliary tasks. The paper is organized as follows; section II covers literature that is related to the problem statement. Then, section III provides an overview of the methodology to be followed. Details of the experiments to be carried out are discussed in section IV. Given that a preliminary analysis has been carried out on the problem statement, section V provides a discussion on some information that has been extracted from the data to be used in the experiments. Finally, conclusion follows in section VI.

II. Literature Review

Considerable literature exists where auxiliary tasks have been used to help with network regularization [2][3]4][5]. For example, Choi *et al.* used transfer learning for their music classification and regression tasks [2]. They used a pre-trained convolutional network feature for a general purpose music representation. The learned feature would then be used for classification and regression tasks which served as main tasks.

Another research that used auxiliary features to improve the main task is that of Libel and Korner [3]. Liebel and Knorner observed that many applications in computer vision required that several atomic tasks be performed using a single image as an input. Although some tasks such as single image depth estimation and semantic segmentation are related, they noted that these tasks were solved individually. As such, they are of the view that solving these related tasks simultaneously could save time and improve the performance of individual tasks. In addition, they argue that related auxiliary tasks have the capability to enrich the learning of an image. To validate their concept, a vision based application of road scene understanding was used. The tasks (main) used in the validation were single image depth estimation and semantic segmentation. Time of day and weather conditions served as auxiliary tasks. Obtained results from validation showed improved performance in the main task.

In addition to the research covered above, Li *et al.* also employed auxiliary tasks in computer vision but for human facial alignment [4]. Having observed that facial alignment and facial attributes are correlated, they developed a mechanism that would align the face fast by using head pose information to handle large view variations. The head pose, in this case, was used as an auxiliary task to aid generalization of the main task. The authors used auto encoders both for generalization and performing the main task.

Other research that has used auxiliary tasks for generalization is Poronkov *et al.* [5]. Poronkov et al. used the tasks in speech identification. They investigated whether a multi-task approach could be used to support inter-speaker awareness when I-vectors are used as auxiliary or not. Using a *Recurrent Neural Network Long term Short term (RNN-LSTM)* model, i-vector extraction and phone-state posterior was performed simultaneously.

Another study on auxiliary task is by Jaderberg *et al.* [7]. Their work was aimed at developing an agent, using reinforcement learning, that is capable of predicting and controlling features from a stream of data obtained a sensorimotor. They emphasize that their built architecture should have the capability to approximate the optimal policy and optimal value function for different pseudo-rewards. In addition, the architecture would have the capability to make auxiliary predictions that would aid to focus the agent on important aspects of the task. Having analyzed their obtained results, it was concluded that adding auxiliary control and reward prediction tasks could improve data efficiency.

In addition to studies mentioned above, Mirowski *et al.*, like Jaderberg *et al.*, employed reinforcement learning to show that auxiliary tasks could improve network performance in particular data efficiency and task performance [8]. Faced with a challenge of navigating an environment, the authors lament the sparsity of rewards and the dynamicity of elements in the environment. To improve statistical efficiency, the learning procedures were bootstrapped by increasing the loss of the auxiliary tasks to support navigational related representation learning. In their case, the auxiliary tasks used were depth prediction and loop closure of a maze. It was observed that their tasks (auxiliary) provided a richer training signal and improved data efficiency.

Although the above mentioned literature have successfully used auxiliary tasks to aid network generalization, it is not clear whether this success depends on the size of the dataset used for training. As such, the author aims to investigate whether small datasets benefit from using auxiliary task to aid network regularization.

III. Methodology

To meet the objective of the paper, the following methodology is to be followed; firstly datasets to be used are to be identified. Secondly, the feature extraction procedure is to be explored. Thirdly, from obtained features, a neural network will be trained. To check the performance of the trained network, a classification procedure on the datasets is done. Lastly, a comparison is done, in terms of performance, between our classifier and traditional classification techniques of machine learning.

A. Dataset Acquisition

To enable tasks like feature extraction, classification and comparison in terms of classifiers performance, datasets are needed. As such, 3 datasets were chosen from UCI machine learning depository [9]. The three datasets to be used are all classification datasets and these are Breast Cancer Coimbra, Seeds and the Cryotherapy Datasets.

B. Feature Learning

To extract features from the obtained datasets, small parts of each of the data sets are used to train auto encoders. Auto encoders have the capability to extract features from datasets which can be used to aid network training.

C. Neural Network Training

Using the learned features, a neural network is trained. The advantage of a neural network is that not only can they be used for feature extraction; they can also be employed to perform classification and regression which is supervised learning tasks.

D. Classification

The trained network will then be used to classify the datasets progressively. The classification will be done in stages, starting will small part of dataset, and then increase the dataset until the whole dataset is classified. To understand the performance of the classifier, Area under the Curve will be used as a performance metric.

E. Comparison

Given that there are a number of traditional classifiers in machine learning, our classifier performance will be compared with 2 classifier techniques using the same datasets. The classifiers to be used are *K Nearest Neighbour (KNN)* and *Support Vector Machine (SVM)* [11].

IV. Experiments

There are a number of experiments to be carried out to achieve the aim of this paper. Firstly, the three dataset will be analyzed to see if any features can be extracted before beforehand. To do this, the different attributes of the data haven been analyzed to see if a correlation exists between them. A number of graphs have been plotted to aid with this analysis. Secondly, to extract salient features that may exist in the datasets, the datasets will be loaded into a neural network

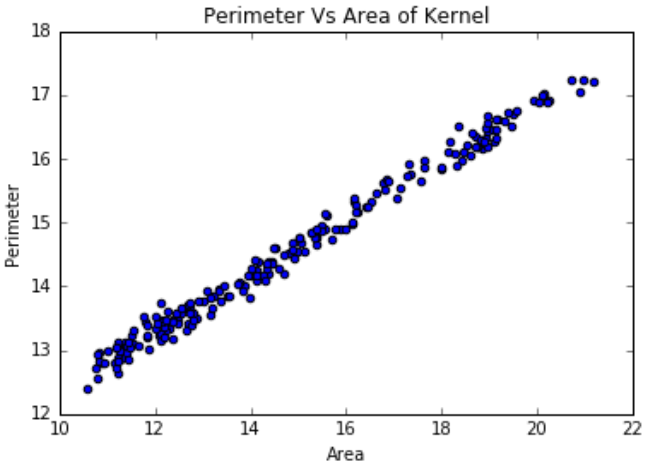

Fig. 1. Perimeter and Area Attribute Linear Correlation

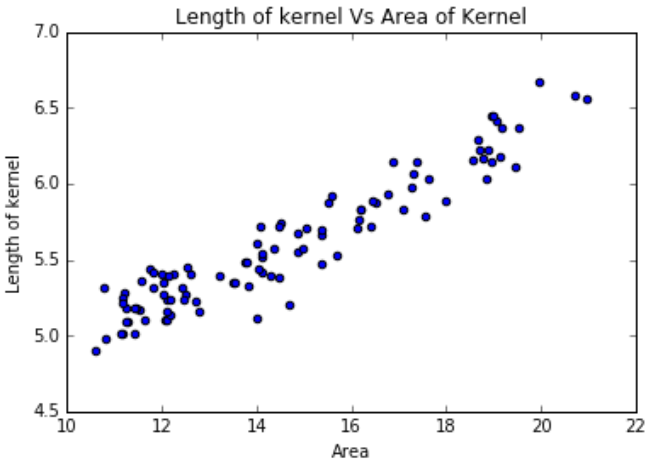

Fig. 2. Length of Kernel and Area Attribute Linear Correlation

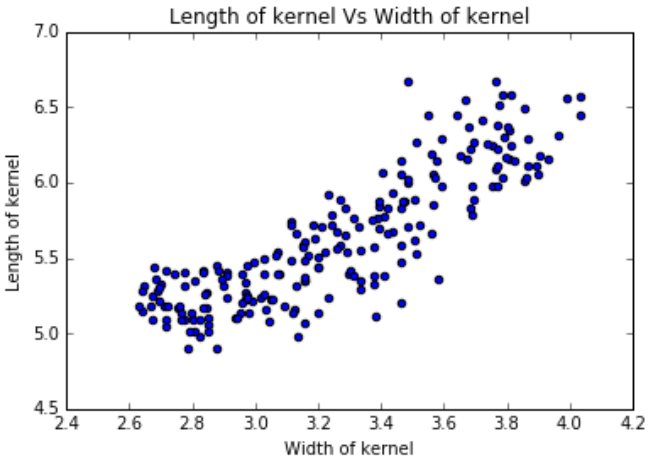

Fig. 3. Length and Width of Kernel Correlation

that will operate in an unsupervised manner using sparse auto-encoders. Sparse auto-encoders have been proposed because of their capability to extract features using their competitively learning rule. Thirdly, a neural network, using learned features will be trained using data from the datasets with varying amounts (i.e. progressively increasing in the number of data to learn from.) Fourthly, the performance of the trained network

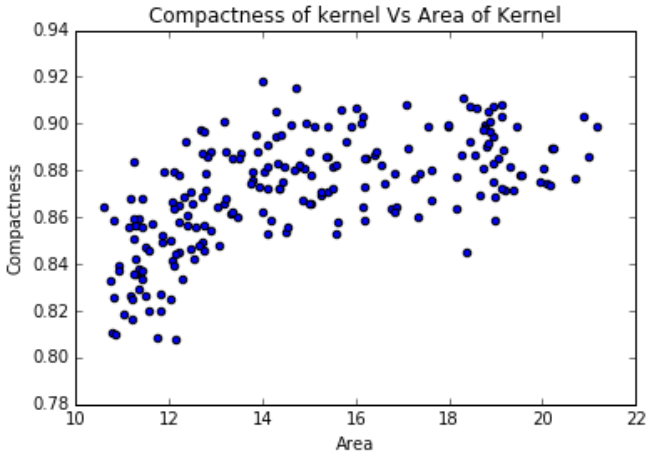

Fig 4. Compactness and Area of Kernel Correlation

will be evaluated using Area under the Curve as the performance metric. Lastly, the performance of the trained network will be compared to SVM and KNN learners.

V. Discussion

Some preliminary work has been done where datasets have been inspected and some initial analysis of the attributes has also been done. Preliminary analysis of the dataset seems to show that there seem to be some level of correlation between different attributes of the datasets used in the experiment. In addition, some patterns seem to exist between attributes. This section works to provide an insight of some of the findings from the analysis.

The following are the findings of the seed dataset attribute analysis. The attributes relation to each other could be summarized to be falling into four categories; linear relation, exponential (positive and negative) and the last one seem to show absence of correlation between some attributes. To show prevalence of linearity among some attributes, figure 1 depicts the relation of perimeter of kernel and area of kernel relation. In addition, figure 2 shows relation of length of kernel attribute and perimeter of kernel which is also linear.

Other than linear correlation, some attributes seem to bear positive exponential correlation. For example, figure 3 shows exponential correlation between Length and width of kernel which seem to be exponentially correlated with positive coefficient. Some attributes are also exponentially correlated but with negative coefficient. An example is compactness and area of kernel shown in figure 4. However, some attributes do not seem to have any correlation. Examples of these are

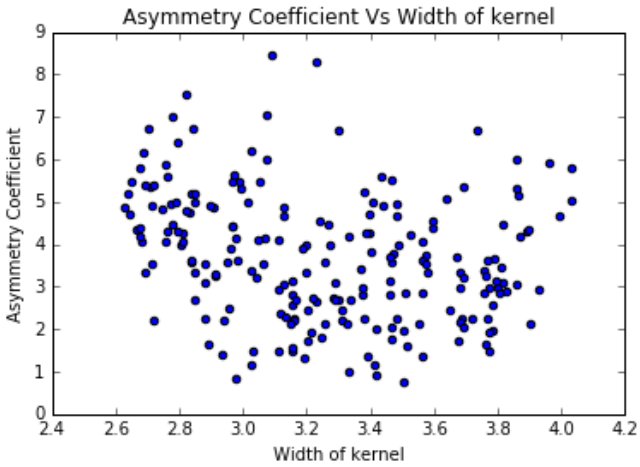

Fig 5. Example of Absence of Correlation between Attributes

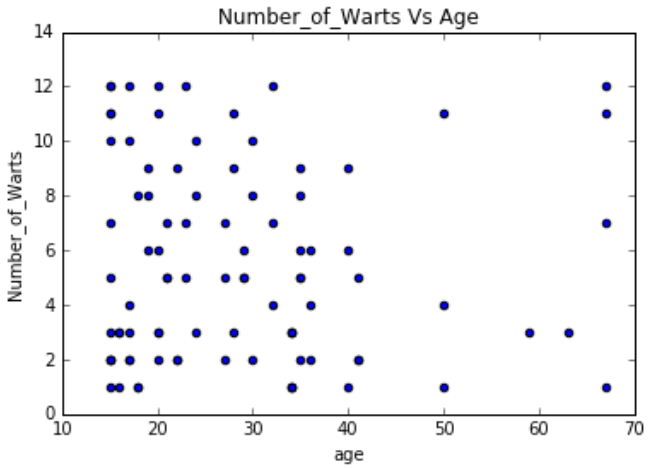

Fig 7. Existence of Warts Vs Age

Asymmetric Coefficient and with of kernel. Equally, there seem to be no correlation between asymmetric coefficient and area of the kernel. Figure 5 shows this characteristic when asymmetric coefficient and width of kernel attributes are plotted.

Another dataset that has been analyzed is that of cryotherapy dataset which covers warts in patients. Preliminary analyses of the dataset seem to indicate some interesting information about the population from obtained attributes. For example, out of the 3 types of warts recorded, types 1 and 3 seem to be more prevalent. This is illustrated in figure 6. Another observation from the attributes is that warts appear to be more prevalent to the population below the age of

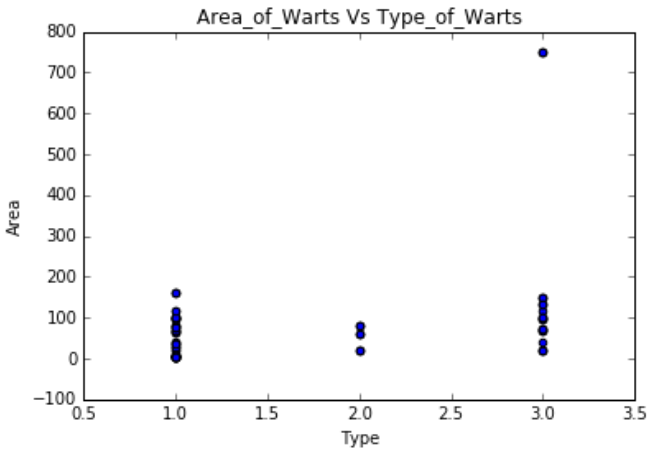

Fig. 6. Area and Type of Warts Distribution

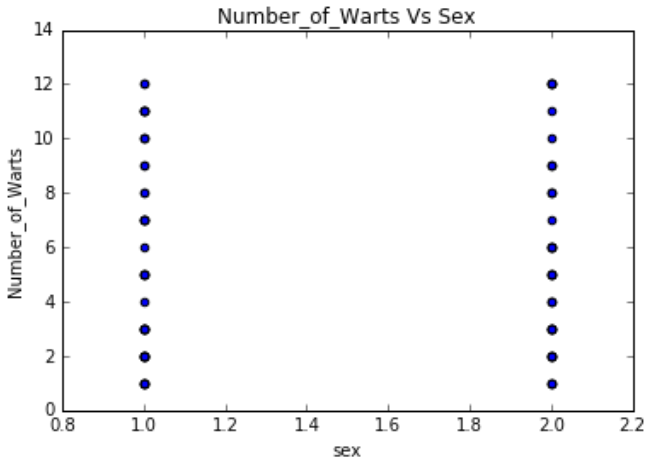

Fig 8. Number of Warts Vs Gender

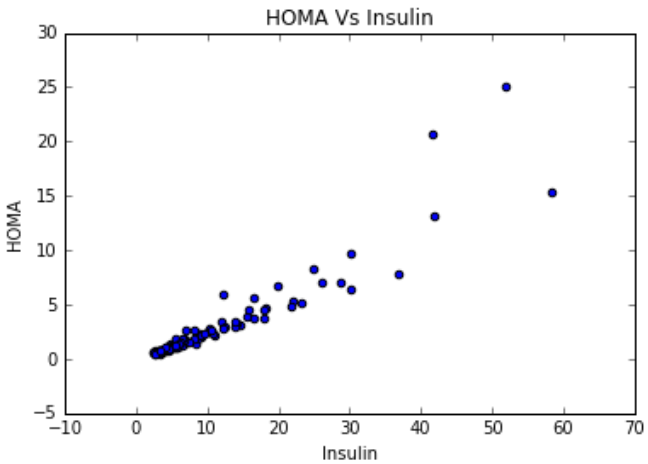

Fig. 9. HOMA and Insulin Attribute Linear Correlation

40 years. Figure 7 depicts a graph of occurrence of warts in the population according to age. Even though warts seem to affect a certain age, in as far as gender is concerned, they do not appear discriminatory. Figure 8 shows the distribution of warts according to gender.

The last dataset used was the breast cancer dataset from

Coimbra. Like other datasets, a number of correlations seem to exist among attributes. Some are linearly correlated, for example HOMA and Insulin attributes. Other attributes seem to be scattered across others, such as BMI across age. Figure 9 depict linearity of HOMA and Insulin attributes while figure 10 shows the distribution of BMI across different age groups.

The above preliminary analysis shows that a number of relations, patterns and features seem to exist between attributes of the datasets. It is assumed that such features will aid with the classification tasks of the neural network classifier to be built. However, to what extent the network will benefit from these features are what the paper will investigate by varying dataset amounts.

Further analyses to be done are the performance of the classifier using different amount of data and comparison of the classifier with SVM and KNN. Using area under the curve metric the classifier performance will be evaluated with changes in amount of datasets. A classifier with achieving a bigger area under the curve is considered a better classifier. Therefore, the variations of the area under the curve with different amount of data will be noted. Depending on the nature of the variation, some conclusion is expected to be drawn on how amount of datasets affect classification.

VI. Conclusion

Auxiliary tasks seem to be widely used in network regularization. To what extent this success depends on the amount of datasets is what this paper works to address. Initial work shows that there are features that can be extracted from the dataset.

VII. Future Work

Given that only preliminary work has been done so far, most of the tasks that have been listed in section IV are yet to be done and they are all to be done in the future.

References

[1] H. Zheng, M. Chen, W. Liu, Z. Yang and S. Liang, "Improving deep neural networks by using sparse dropout strategy," 2014 IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP), Xi'an, 2014, pp. 21-26.

[2] K. Choi, G. Fazekas, M. Sandler and K. Cho, "Transfer learning for music classification and regression tasks", arXiv.org, 2017. [Online]. Available: http://arxiv.org/abs/1703.09179. [Accessed: 08- Feb- 2019].

[3] L. Liebel and M. Körner, "Auxiliary Tasks in Multi-task Learning", arXiv.org, 2018. [Online]. Available: http://arxiv.org/abs/1805.06334. [Accessed: 08- Feb- 2019].

[4] Q. Li, Z. Sun and R. He, "Fast multi-view face alignment via multi-task auto-encoders," 2017 IEEE International Joint Conference on Biometrics (IJCB), Denver, CO, 2017, pp. 538-545.

[5] G. Pironkov, S. Dupont and T. Dutoit, "I-Vector estimation as auxiliary task for Multi-Task Learning based acoustic modeling for automatic speech recognition," 2016 IEEE Spoken Language Technology Workshop (SLT), San Diego, CA, 2016, pp. 1-7.

[6] K. Choi, G. Fazekas, M. Sandler and K. Cho, "Transfer learning for music classification and regression tasks", arXiv.org, 2019. [Online]. Available: http://arxiv.org/abs/1703.09179. [Accessed: 08- Feb- 2019].

[7] M. Jaderberg et al., "Reinforcement Learning with Unsupervised Auxiliary Tasks", arXiv.org, 2016. [Online]. Available: http://arxiv.org/abs/1611.05397. [Accessed: 11- Feb- 2019].

[8] P. Mirowski et al., "Learning to Navigate in Complex Environments", arXiv.org, 2017. [Online]. Available: http://arxiv.org/abs/1611.03673. [Accessed: 11- Feb- 2019].

[9] Archive.ics.uci.edu, 2019. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/. [Accessed: 5- Feb- 2019].

[10] S. Lu, H. Liu and C. Li, "Manifold Regularized Stacked Autoencoder for Feature Learning," 2015 IEEE International Conference on Systems, Man, and Cybernetics, Kowloon, 2015, pp. 2950-2955.

[11] S. Kotsiantis, I. Zaharakis and P. Pintelas, "Machine learning: A review of classification and combining techniques", 2006, Artificial Intelligence Review. 26. 159-190. 10.1007/s10462-007-9052-3.

Project Plan

Tasks	**Start Date**	**End Date**	**Duration**
Train Auto encoders	25th February 2019	8th March 2019	2 Weeks
Train Classifier	11th March 2019	22nd March 2019	2 Weeks
Compare Performance of Classifier with Change of Data Amount	25th March 2019	5th April 2019	2 Weeks
Compare Classifier with SMV and KNN	8th April 2019	19th April	2 Weeks
Report Write up	22nd April	24th April	3 Days