The performance of LSTM-Casper neural network with classification task on Adult dataset

Bing Yu
Research School of Computer Science
Australian National University
Email: u5991070@anu.edu.au

Abstract. LSTM-Casper is a deep neural network which combines Long Short-Term Memory and Casper network. It is a powerful neural network technique for different tasks in machine learning area. In this paper, LSTM-Casper was implemented to solve the classification task on UCI adult dataset. It could predict whether the income of one person is higher than 50K, based on their different personal attributes. Training with whole adult dataset and evaluating it by 10-fold cross-validation and Area under curve of Receiver Operating Characteristic (AUC). Casper-network has reached the accuracy of 86.19 % and 0.8775 AUC value. Compare with the accuracy of other classification technique applied on adult dataset like Casper, Naïve Bayes and 3-layer feedforward neural network. The result of LSTM-Casper is much better. The training of Casper net is much faster than 3-layer neural network. What's more, comparing the AUC value of LSTM-Casper with Casper, Support Vector Machine and Decision Tree applied on Adult dataset, LSTM-Casper is not that good.

Keyword: LSTM, Casper, Deep Learning, Adult Dataset, 10-fold cross-validation, AUC

1 Introduction

"Artificial neural networks (ANNs) are biologically inspired computer programs designed to simulate the way in which the human brain processes information." (Agatonovic-Kustrin & Beresford 2000) It has solved many classification and regression task in machine learning area. In this paper, one modified Casper algorithm, which is the cascade correlation algorithm with progressive RPROP, was provided (Treadgold & Gedeon 1997). It connected the Long Short-Term Memory Network with Casper network. And it was implemented to solve the classification task on adult dataset in UCI Machine Learning repository (Dua & Karra 2017). The task is to predict the salary of person based on their 14 different attributes. To compare with the LSTM-Casper net, three layers feedforward neural network and general Casper net are also implemented. Other machine learning technique like Naïve Bayes, Decision Tree and Support Vector Machine are also used for comparison.

The extraction of adult dataset is finished by Barry Becker from the 1994 Census database (Dua & Karra 2017). It has 14 attributes and 48842 instances. The dataset is designed for predicting the whether a person's income is higher than 50K per year. The attributes used for prediction are age, workclass, fnlwgt, education, education-num, martial-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week and native-country. Adult dataset has large number of instance, which could ensure the amount of training data and avoiding the under fitting problem. In addition, a lots of machine learning techniques have been successfully applied on this task, such as Naïve-Bayes and Decision-Tree-Hybrid (Kohavi 1996). These previous works could justify the feasibility of the task and provided many results to compare with my work.

Casper is a modified algorithm from cascade correlation algorithm. Similar as cascade network, it has growing neural topology (Treadgold & Gedeon 1997). One hidden neuron was added on each training stage. But instead of the correlation measure or weight freezing strategy (Fahlman & Lebiere, 1990) used in cascade correlation algorithm, Casper use RPROP which could adjust the learning rate during the training process to speeding up the training process and getting better result (Riedmiller & Rprop 1994). Also, Casper use different learning rate in different part of the network. The network is separated into three parts, each part will have unique learning rate (Treadgold & Gedeon 1997)). By using different learning rate, the training of network will be easier and faster.

Long Short-Term Memory network is a popular network structure invented by Hochreiter and Schmidhuber (Hochreiter & Schmidhuber 1997). It is recurrent, at each time step, it could get one input and two hidden states from previous step, then generating two hidden states as the input of next step (Hochreiter & Schmidhuber 1997). It has three gates and one memory cell to help the net deal with the information from previous step (Hochreiter & Schmidhuber 1997). Thus, it is good for sequence prediction problem which need to remember and analyze the information from previous input.

The LSTM-Casper network connects the LSTM and Casper net. The inputs are put into the LSTM first. In this paper, I use Pytorch to implement this net. Data with different attributes is treated as one sequence. During the forward process, one attribute is put into the LSTM in each step. Finally, the output of LSTM, the combination of all hidden states in each step, are put into the Casper net. Two-dimension vectors are generated at last. The Casper net implemented in this paper could have 16 hidden neurons in maximum.

The performance of LSTM-Casper net is delightful. On whole dataset, it gets the accuracy of 86.19% by using 10-fold cross-validation. This result is better than the performance of general Casper net (84.65%), Naïve Bayes (83.5%) and the three layers feed forward neural network (82.01%). Not only 10-fold cross validation, the Area under curve of Receiver Operating Characteristic (AUC) is also used for evaluation. We used 4000 data for training and 35222 data for testing. The AUC of LSTM-Casper is 0.8775, which is worse than Casper (0.8913), Support Vector Machine (0.8980) and Decision Tree (0.8901). What's more, I found that the training of general Casper net is faster than normal neural network with same amount of hidden neuron. Based on the test, the number of epochs that normal network needed to get the accuracy of 84% on adult dataset is more than 50 times of the epochs needed on Casper network.

2 Method

LSTM-Casper is a neural network technique modified from Casper algorithm. To clarify this model, I will start with the Basic idea of general Casper net and LSTM. Then, I will introduce the details about LSTM-Casper net. Also, I will do some analyzation on adult dataset and talk about the encoding strategy I used based on my analyzation. Finally, I will talk about my implementation and evaluation.

2.1 Cascade correlation algorithm Employing Progressive RPROP(CASPER)

Casper algorithm is a modification from cascade correlation algorithm. Similar to cascade correlation algorithm, the neuron will be added to network during training process. New added neuron will get inputs from the original inputs and the outputs from all of previous added neuron. Different to Cascade Correlation Algorithm, it won't freeze the weights of previous trained network. It will separate the neural network into different parts and sets different learning rate for them (See Figure 3). One part is the layer from previous network to new added network, the learning rate of this part is L1. Another is the layer from new added hidden neuron to the output layer, the learning rate is L2. All of the previous trained network is the last part, the learning rate is L3. Usually, L1>>L2>L3 (Treadgold & Gedeon 1997). Also, instead of the correlation measure or weight freezing strategy used in cascade correlation algorithm, CASPER use RPROP, which could adjust learning rate during the training process. (Treadgold & Gedeon 1997). By using these strategies, the neural network could be adjusted during the whole process, we won't need to worry about the poor features learned at very beginning. The network is more flexible. Also, by avoiding to use correlation measure, the "jagged edge" problem in network output is not that serious. The quality of the generation is better.

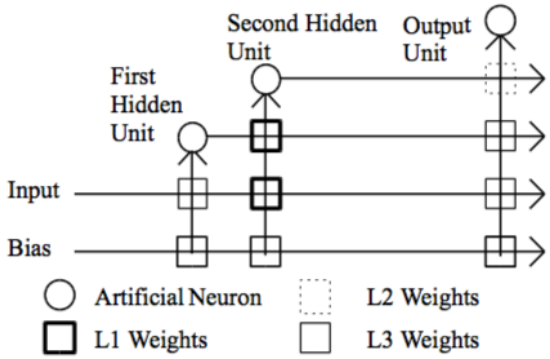

Fig 1. The structure of Casper net (Treadgold & Gedeon 1997)

2.2 Long Short-Term Memory

Long Short-Term Memory is modified from recurrent neural network (Hochreiter & Schmidhuber 1997). In each step, the network could get input and hidden state from previous step (Hochreiter & Schmidhuber 1997). These information from current and previous input will generate the information for future step by using the gate in network. But different to general RNN, LSTM has one memory cell, two parallel hidden states: hidden state and cell state, and three different gates: forget gate, input gate and output gate, which make the network more powerful (See Fig 2).

Here are the specific steps of information processing in LSTM. First, previous hidden state h_{t-1} and current input x_t are put into forget gate (see formula (1)). Since σ is a sigmoid function, forget gate will generate a value between 0 and 1. This value will multiple the previous cell state C_{t-1} to decide how many previous memories should be "forget" in currency step. Then, h_{t-1} and x_t are put into the input gate (see formula (2)) and memory cell (see formula (3)) to decide how many input should be added to current cell state. Current cell state is calculated by formula (4). Previous cell state is multiplied by the value generated by forget gate and the current memory cell is multiplied by the value generated by input gate. The current cell state is the sum of two scaled "cell state". Finally, h_{t-1} and x_t are put into the output gate (see formula (5)). Current cell state will multiply the output gate the generate new hidden state h_t, which will be used in next step. After finishing all of the steps in one sequence, we could use the last hidden state, cell state or all of the previous hidden state as output for specific tasks.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$
$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

2.3 LSTM-Casper

The LSTM-Casper network is the combination of LSTM and Casper net. The input will put into LSTM as sequence first. After finishing all of the steps in LSTM, all of the hidden state generated in each step will be combined as a vector and put into Casper net. Similar to Casper, the LSTM will also use RPROP as the optimizer and it will have two different

learning rates. Before adding the first new neuron into Casper, the learning rate of LSTM is L1 (same as L1 in Casper). After adding the neuron into Casper, the learning rate of LSTM will always be L2 (same as the L2 in Casper).

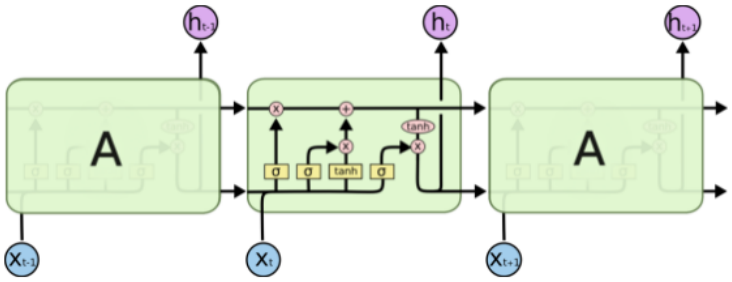

Fig 2. The structure of LSTM (Colah 2015)

2.4 The Adult Dataset

Basic information. Adult dataset is the dataset in UCI machine learning repository. It is built by Barry Becker in the 1994 from the extraction Census database (Dua & Karra 2017). It has 14 attributes and 48842 instances and It was designed for predict whether one person's salary is higher than 50K. To clarify the dataset, here is a specific example of data in adult: "39, State-gov, 77516, Bachelors, 13, Never-married, Adm-clerical, Not-in-family, White, Male, 2174, 0, 40, United-States, <=50K". (Dua & Karra 2017)

It has 14 attributes, based on the sequence shown in example, they are age, workclass, fnlwgt, education, education-num, Marital-status, occupation, relationship, race, sex, capital-gain, capital-loss, hours-per-week, native-country and the salary. For specific explanation on these attributes, see Appendix.

Dataset Analyzation. Before perform encoding on dataset, here is some statistical result of the dataset (See fig3 and fig4). The reason for analyzing the data is to get better understanding of it before implementation. So that we could find more efficient encoding technique. It might also help us to explain some different classification result.

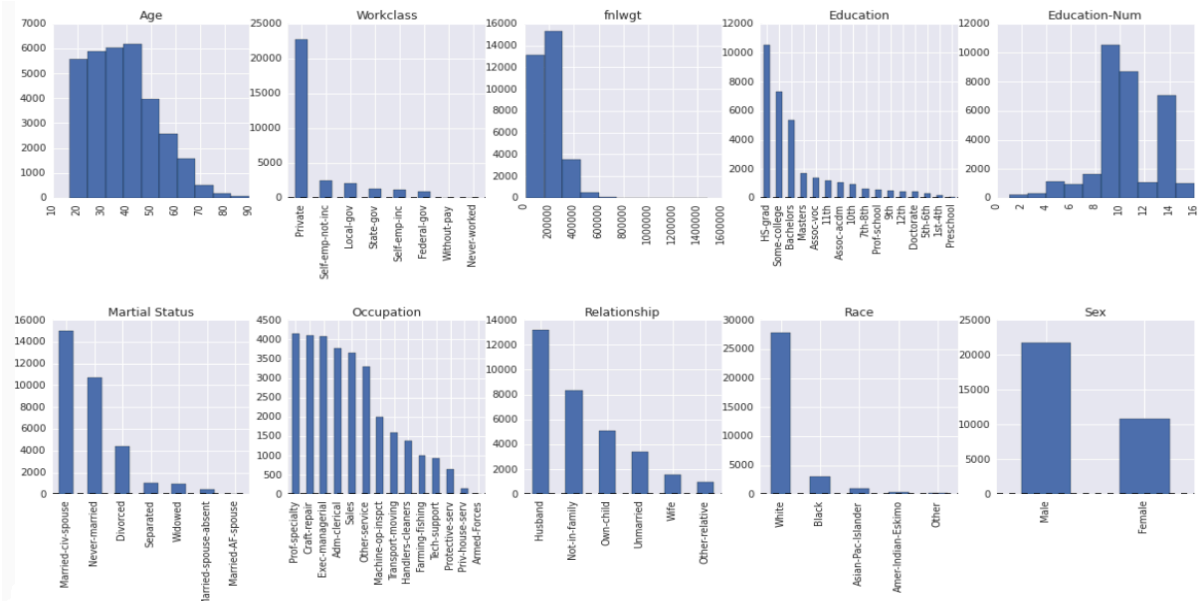

Fig 3. Statistical Result of Adult (Valentin 2015)

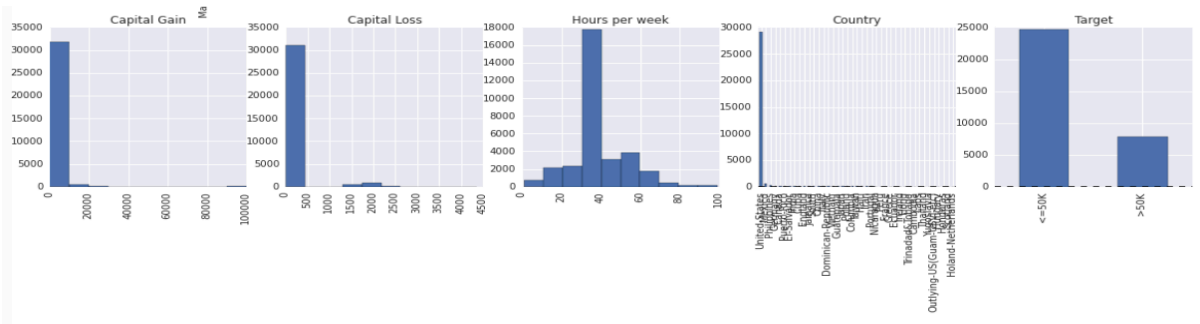

Fig 4. Statistical Result of Adult (Valentin 2015)

As you can see in above most of the people's native country are US, most of the people's race are white and most of the people's workclass are Private. Most of the people have never got income or lost money from their investment. The general income of investment is around 20000 and the general loss of the investment is around 2000. Also, the amount of people whose salary is lower than 50K is much more than the people whose salary is higher than 50K. More specifically: Probability for the label '>50K': 23.93%. Probability for the label '<=50K': 76.07% (Dua & Karra 2017). So even if the accuracy of classification on network could get 76%. It is still possible that the network doesn't learn anything. Because it could only predict the '<=50K' class and getting the accuracy of 76%.

In addition. Based on the observation of the dataset, some of the attributes has very close correspondence. For the education and education-num attributes. There is one to one correspondence between them. The two columns have the same information. The correspondence is shown in Table1.

Also, the relationship attribute has connections to sex, if the relationship is husband, the sex is Male. If the relationship is wife, the sex is Female.

Table 1. Correlation between 'education' and 'education-num'

Education	Education-num	Education	Education-num
Preschool	1	HS-grad	9
1st-4th	2	Some-college	10
5th-6t	3	Assoc-voc	11
7th-8th	4	Assoc-acdm	12
9th	5	Bachelors	13
10th	6	Master	14
11th	7	Prof-school	15
12th	8	Doctorate	16

Encoding the data. Here is the encoding strategy. The age: Use the original number. The fnlwgt: Compare with the encoded number of other attribute (usually less than 50), the number of fnlwgt are too large (usually 200000-400000). If we use it without encoding. Then, the training might be very hard, because it will lead to large neural output at the beginning. So, encode the original value k of fnlwgt into k/10000**.** The education and education-num: Attribute has exactly same information of education-num. Deleting this education and using the original number of education-num**.** The capital-gain: Although most of the people do not have capital gain. The original number is still too high compare with other encoded number. Same reason as fnlwgt, we need to encode them. The income is around 20000-100000. Encode the original value k of fnlwgt into k/10000. The capital-loss: Most of the people do not have capital loss. The original number is still too high for them (larger than 10000). Same reason ad fnlwgt, we need to encode them. The loss is around 0-3000. Encode the original value k of fnlwgt into k/10000. The capital-gain and capital-loss should use the same encoding strategy because they all stand for the amount US dollars. At the beginning, their relative 'influence' on the output should not be changed by encoding. The hours-per-week: Using the original number**.**

To check the specific encoding strategy of other attributes, see Appendix.

2.5 Implementation & Evaluation

The LSTM-Casper is implemented with Pytorch. For LSTM part, the input dimension is 1, the sequence steps are 13. That's because after the encoding process, each adult dataset has 13 attributes, each attribute is one single number. The attributes will be put into the LSTM one by one to generate the output. The hidden dimension of LSTM is 10. Thus, the dimension of LSTM output is 13*10=130. For Casper part, the maximum number of hidden neuron is 16. All of the network structure was defined before the training and the training is separated into 17 different stages. During different stages, different forward() function will be used and different parts of the network will be optimized by different optimizer. In each stage, there will be a variable called num_loss_without_decrease to storage the number of epoch which didn't decrease the loss. If it is larger than 3, stage will switch to the next one, which is equal to adding neuron into the network. The training will be finished until the all of neuron is added into the network and the loss becomes stable. As mentioned before, the network has three different learning rate L1, L2, L3 in different part of the network. In my implementation, L1=0.01, L2=0.005, L3=0.001. Also, all training data is squeezed in one batch.

One evaluation method of the model is 10-fold cross-validation (Kohavi 1995). The dataset used to train the network will be separated by 10 different part. Ten models will be trained, each model use one of ten parts as testing date and the rest as training data. After the training, each model will get input from their testing data and output of model will be delivered into an softmax() function. Then comparing output of softmax() with the original class. Next, calculating the class that are correctly predicted and calculating the accuracy for this model. Finally, calculating the mean of all of the testing accuracy of the model as the score of 10-fold cross-validation. The benefits for choosing 10-fold cross-validation instead of testing the accuracy of model by using training data is very obvious. It could prove that the network is not overfitting because we use different dataset to train and test (Kohavi 1995). Also, compare with the strategy which separate the data into training set and testing set, all of the data are used for training and testing in 10-folf cross-validation (Kohavi 1995). When training one net, one 10% of the data are not used. Also, the training is repeated for 10 times, the mean accuracy could avoid the random factors in training.

Another evaluation used for this model is the Area under curve of Receiver Operating Characteristic (Rich & Alexandru 2004). It is designed for evaluating binary classifier. Since adult dataset only has two class which are ">50k" and "<=50K", it is suitable for using AUC in this task. The x axis of Receiver Operating Characteristic (ROC) is the False positive

rate(FPR) (see formula (6)). FP is amount of negative data which is predicted as positive. TN is the amount of negative data which is predicted as negative. The y axis of ROC is True positive rate (TPR) (see formula (7)). TP is amount of positive data which is predicted as positive. FN is amount of positive data which is predicted as negative. To draw the ROC, we need to list all of the data from large to small by their probabilities belongs to positive class first. Then, start from the first data, get the probability of the data one by one, using it as the threshold for judging all of the data (Rich & Alexandru 2004). If the probability of one data is larger than the current threshold then it is predicted as positive class, otherwise it is predicted as negative class (Rich & Alexandru 2004). The FPR and TPR will be calculated and drawn as one point each time we change the threshold. After the smallest probability in dataset is used for threshold. The drawing will stop. Connecting all of the points, we could get the ROC. The first point in ROC should always be (0, 0), the last point should always be (1,1). Because at the beginning, all of the data are predicted as negative, FP and TP are equal to 0. At the end, all of the points are predicted as positive, TN and FN are 0. Calculating the area under this curve, we will get AUC (Rich & Alexandru 2004).

The reason for AUC could be used by evaluating binary classification is clear. If the training is useless, the probabilities of data should distribute randomly, thus, in each point, TPR should be similar to FPR, because the right prediction should be similar to wrong prediction. But if the model is trained well, the TPR should be higher than FPR. The more accurate the model is, the larger AUC value it will have. Compare with evaluation based on accuracy, using AUC has several advantages. First, when the binary class are not equally distributed, for example 80% of the data belongs to class 0, we can get the accuracy of 80% by predicting very data as class 0. The accuracy is high but the is almost useless. By using AUC, this problem could be solved. With the increasing of threshold, FPR should grow as fast as TPR in the bad model mentioned before.

To evaluate the AUC of LSTM-Casper, we use the roc_curve() and roc() function in scikit-learn package (Scikit-Learn 2018). "<=50k" is set as the positive class. The output of each data is a two-dimension vector. The vector was put into one softmax() function to scale it between 0 and 1. Based on the meaning of this vector, the first number is used to represent the possibility of data belongs to "<=50k" class.

$$FPR = \frac{FP}{FP+TN} \quad (6)$$

$$TPR = \frac{TP}{TP+FN} \quad (7)$$

3 Result and discussion

3.1 10-fold cross-validation

The LSTM-Casper net, is trained with different amount of data and test by 10-fold cross-validation. To compare with my model, I also trained the Casper net with same amount of maximum hidden neural in LSTM-Casper. Also, to compare with the Casper net, a 3-layers feedforward neural networks with the same number of hidden neurons are built. Based on the observation of training, generally, Casper training should be finished in 1000 epochs. Thus, the amount of training epochs for feedforward networks are set to 1000. Also, same dataset and evaluation method are used for training these networks.

Based on the works of Kohavi who applied Naïve Bayes on adult dataset classification task and using 10-fold cross-validation the evaluate the model (Kohavi 1996). I choose Kohavi's work to compare with my implementation. Finally, we got four groups of accuracy scores to compare with (see Fig 5).

Based on the result shown in Fig 5, in general, with the increasing of the instance amount, the accuracy score of all of the models tends to increase, that is because the model will get more information with increasing the amount of training data. It will have higher possibility to learn more patterns hidden in data and the prediction will be more accurate. More specifically, for Casper and feedforward neural network. The more training the data we have, the more chances the network will get to make adjustment of the weight matrix so that we might get better accuracy.

The performance of LSTM-Casper network is the best, the accuracy is always higher than the performance of Casper without LSTM. One reason for that is the topology of network, it is more complex now, not surprise it could get better performance with same task. Also, the encoding strategy used in this paper is simple and straightforward, which might not be perfect for the classification task. By adding the LSTM, for each attribute in the data, we could generate one 10-dimension vector to represent it in following Casper net. After training, these vectors are optimized for better performance on Caper. Thus, the LSTM improved the quality of encoding, the performance of network should be better.

The accuracy score of Casper algorithm is always higher than the feedforward neural network with 1000 epochs. That's because the topology of Casper is more complicated than normal neural network (multiple layers vs one layer). With the adding of new nodes, more specific features could be learned. Also, the Casper used the RPROP which could adjust the learning rate during the training. So, Casper will get the minimum (at least local minimum) in less epoch. Another detail we could see in the figure is the with the growing of instance the accuracy of Casper increasing more smoothly than feed forward neural network. That's is also due the using of RPROP, so that Casper could reach the minimum point faster. The unstable of normal neural network is that it is not trained well with that amount of training epoch. To compare with the training speed of two network. Table 2 is the training epoch needed for them to get the 84% accuracy on the classification of adult. They use the same training and testing dataset.

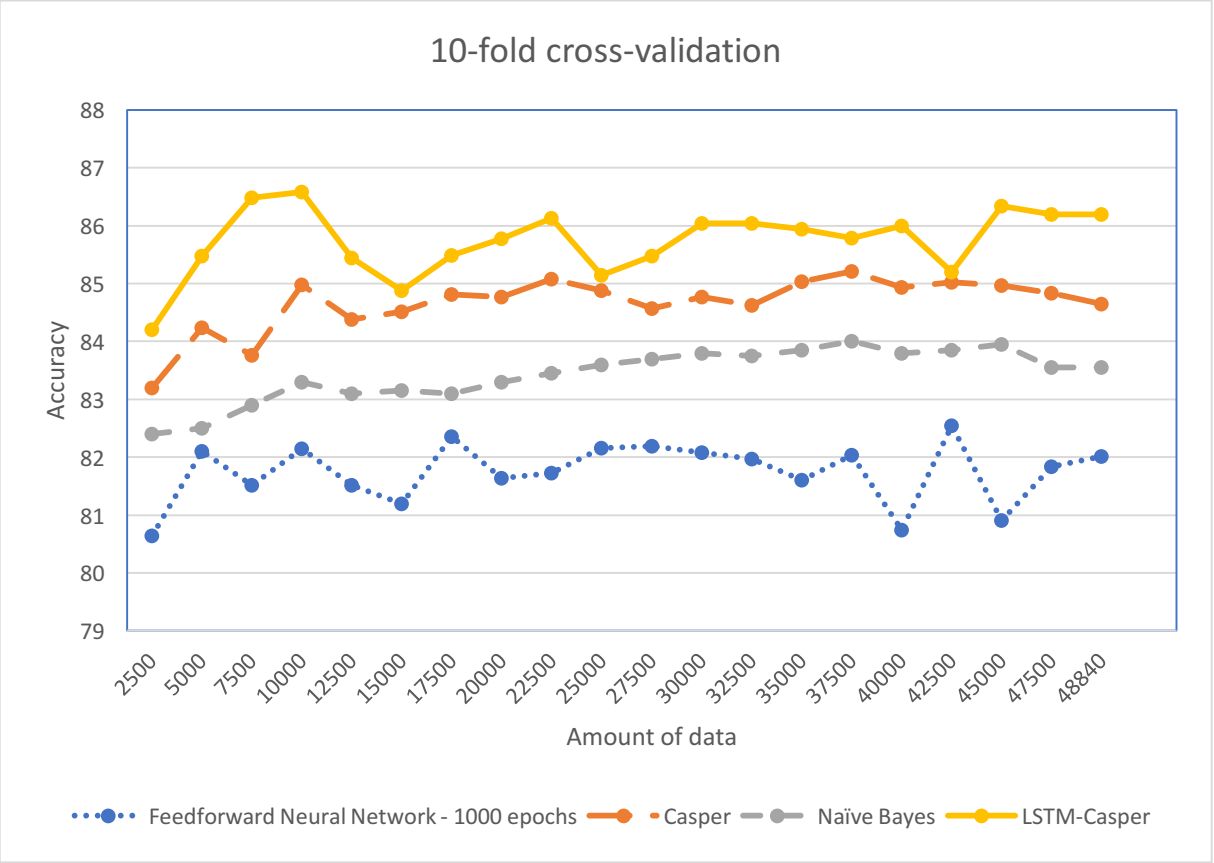

Figure 5. 10-fold cross-validation of Casper, Feedforward Neural Network, Naïve Bayes

Table 2. The number of epoch to getting the accuracy of 84%

The network	Number of epochs/ 1st test	Number of epochs/ 2nd test	Number of epochs/ 3rd test
Casper	301	535	372
Normal Neural Network	25603	26435	24012

The epoch needed for normal neural network is much more than the epoch needed for Casper. Also, the topology of Casper is simpler at the beginning epochs, the training one epoch should also be faster. Thus, the training of Casper is faster than the training of normal neural network.

The accuracy of LSTM-Casper and Casper are always higher than Naïve Bayes Model which is a classic machine learning algorithm (Ng & Jordan 2002). It is a supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of independence between every pair of features (MaCallum & Nigam 1998). Given a class y and a dependent feature x_1 to x_n. Based on Bayes rules, we could use the value of $x_1, \dots, x_n$ to predict the value of y (MaCallum & Nigam 1998).

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n|y)}{P(x_1, \dots, x_n)} \qquad (8)$$

The reason for Naïve Bayes performs worse than Casper is the assumption of the independence between different attributes. Naïve Bayes assume that all of the attributes are independent to each other (Ng & Jordan 2002). But based on the data analyzation finished before. The hidden connections between different connections exits. For example, the sex and relationship attributes. If the sex is female, the attribute of relationship could not be Husband. So, neural network strategy is more suitable on the classification task on adult. And the performance of Casper algorithm on adult dataset should be better than Naïve Bayes.

3.2 Area of the Receiver Operating Characteristic

The amount of training data is 4000, the amount of testing data is 35222. They are same with the training and testing data used by Rich and Alexandru's work on adult dataset (Rich & Alexandru 2004). Table 3 shows the AUC value of different model. The LSTM-Casper is worse than Casper and traditional machine learning strategy, DT and SVM, by using AUC evaluation.

Table 3. AUC of different model

	LSTM-Casper	Casper	DT	SVM
AUC	0.8775	0.8913	0.8901	0.8980

There are several reasons for it. First, the distribution of Adult dataset is uneven, the percentage of "<=50k" data is 75.22% (Dua & Karra 2017). Since the training of network is based on the improving the accuracy, it is reasonable to get different performance on AUC evaluation and accuracy evaluation. Because in this case, the accuracy of "<=50k" class might be higher than the accuracy ">50k" class, which could lead to better accuracy and worse AUC. Second, the amount of training data is 4000, which might be too small for LSTM-Casper, it has the risk of overfitting, but Casper net is simpler, so the overfitting problem might not happen on Casper. Third, in Rich's work, they choose the model with best performance in a group of models (Rich & Alexandru 2004). By changing the hyper-parameters of LSTM-Capser and training a group of models, the performance of LSTM-Casper should have space for improvement.

4 Conclusion and future work

In this paper, we have shown that LSTM-Casper performs well over all of other chosen models by using 10-fold cross-validation. It has reached the accuracy of 86.19% by using all of the data in UCI Adult dataset. We have proved that LSTM in LSTM-Casper could improve the quality of the original encoded data. So, this model could be applied on other tasks which need complex encoding strategy. Also, training of Casper is much faster than training general neural network. It could be applied on some task whose training time is limited.

Evaluating by AUC, the trained LSTM-Casper model is not performing very well. That is because the model is trained for improving the accuracy. So, if we could modify the network, using AUC to do the back propagation. We should get higher AUC value. Since the distribution of adult dataset is uneven, the model trained by using AUC should get better prediction on the class with less amount of data. Which is useful in some case.

So far, all of the experiments are based on adult dataset, if we could test the models on different datasets, some new discoveries might be found. Also, there are a lot of hyper-parameters in LSTM-Casper net, by using genetic algorithm, we could find a better combination of these hyper-parameters.

Reference

1. Adams, A., & Waugh, S. (1995, November). Function evaluation and the cascade-correlation architecture. In *IEEE International Conference on Neural Networks* (pp. 942-946).

2. Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, *22*(5), 717-727.

3. Caruana, R., & Niculescu-Mizil, A. (2004). An Empirical Evaluation of Supervised Learning for ROC Area. In *ROCAI*(pp. 1-8).

4. Colah (2015). *Understanding LSTM Neural Networks*. [online] Colah's Blog. Available at: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ [Accessed 30 May. 2018]

5. Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

6. Fahlman, S. E., & Lebiere, C. (1990). The cascade-correlation learning architecture. In *Advances in neural information processing systems* (pp. 524-532).

7. Hall, P. (2013). [online] Available at: https://www.quora.com/What-is-the-meaning-of-capital-gain-capital-loss-and-fnlwgt-in-adult-dataset-from-UCI?utm_medium=organic&utm_source=google_rich_qa&utm_campaign=google_rich_qa [Accessed 29 Apr. 2018].

8. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

9. Kohavi, R. (1995, August). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*(Vol. 14, No. 2, pp. 1137-1145).

10. Kohavi, R. (1996, August). Scaling up the accuracy of Naive-Bayes classifiers: a decision-tree hybrid. In *KDD* (Vol. 96, pp. 202-207).

11. McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization* (Vol. 752, No. 1, pp. 41-48).

12. Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems* (pp. 841-848).

13. Riedmiller, M., & Rprop, I. (1994). Rprop-description and implementation details.

14. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533.

15. Scikit-Learn 2018. *Scikit-Learn Machine Learning in Python*. [online] Scikit-Learn. Available at: http://scikit-learn.org/stable/ [Accessed 30 May. 2018]

16. Treadgold, N. K., & Gedeon, T. D. (1997, June). A cascade network algorithm employing progressive RPROP. In *International Work-Conference on Artificial Neural Networks* (pp. 733-742). Springer, Berlin, Heidelberg.

17. Valentin, M. (2015). *Adult Income Data Set Analysis with IPython*. [online] Valentin Mihov's Blog. Available at: https://www.valentinmihov.com/2015/04/17/adult-income-data-set/ [Accessed 29 Apr. 2018].

18. ujjwalkarn. (2016). *A Quick Introduction to Neural Networks*. [online] Available at: https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/ [Accessed 29 Apr. 2018].

Appendix

1. The explanation of attributes

age is the age of informant, it is continuous number. **workclass** is the working state of the person. It has 8 types, which are Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked. **fnlwgt** is the third attribute. Since the adult dataset is only one sample survey. It couldn't cover all of the people in each country. fnlwgt means the amount of people that one sample could represents in their country (Hall 2013), it is continuous number. **education** stands for the highest degree the person has finished. It has 16 different types, they are Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool. **education-num** is from 1 to 16, it is equal to education attribute. **Marital-status** attribute has 7 types, which are Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse. **occupation** has 14 different types, which are Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces. **relationship** is the relationship between the recorded people and their families. Which includes Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried. **race** has 5 different types, which are White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black. **sex** attribute could be Male or Female. The attribute **capital-gain** is continuous number which stands for income from investment sources, apart from salary (Hall 2013). The **capital-loss** is continuous number which stands for losses from investment sources, apart from salary (Hall 2013). **hours-per-week** is the weekly working hours of that person. It is continuous number. The attribute **native-country** has 40 different types, they are United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad & Tobago, Peru, Hong, Holand-Netherlands. The last attribute is the class, it is the information we want to predict, whether >50K or <=50K.

2. Encoding Startegy

These tables are the encoding strategy for attributed not mention in data analysis section. They are workclass, marital-status, occupation, race, sex and native-country

Table 4. workclass encoding

Attribute	Encoded number	Attribute	Encoded number
Private	1	State-gov	6
Self-emp-not-inc	2	Without-pay	7
Self-emp-inc	3	Never-worked	8
Federal-gov	4	?	9
Local-gov	5		

Table 5. marital-status encoding

Attribute	Encoded number	Attribute	Encoded number
Married-civ-spouse	1	Widowed	5
Divorced	2	Married-spouse-absent	6
Never-married	3	Married-AF-spouse	7
Separated	4		

Table 6. occupation encoding

Attribute	Encoded number	Attribute	Encoded number
Tech-support	1	Adm-clerical	9
Craft-repair	2	Farming-fishing	10
Other-service	3	Transport-moving	11
Sales	4	Priv-house-serv	12
Exec-managerial	5	Protective-serv	13
Prof-specialty	6	Armed-Forces	14
Handlers-cleaners	7	?	15
Machine-op-inspct	8		

Table 7. race encoding

Attribute	Encoded number	Attribute	Encoded number
White	1	Other	4
Asian-Pac-Islander	2	Black	5
Amer-Indian-Eskimo	3	Black	5

Table 8. sex encoding

Attribute	Encoded number	Attribute	Encoded number
Male	1	Female	2

Table 9. native-country encoding

Attribute	Encoded number	Attribute	Encoded number
United-States	1	Ireland	23
Cambodia	2	France	24
England	3	Dominican-Republic	25
Puerto-Rico	4	Laos	26
Canada	5	Ecuador	27
Germany	6	Taiwan	28
Outlying-US(Guam-USVI-etc)	7	Haiti	29
India	8	Columbia	30
Japan	9	Hungary	31
Greece	10	Guatemala	32
South	11	Nicaragua	33
China	12	Scotland	34
Cuba	13	Thailand	35
Iran	14	Yugoslavia	36
Honduras	15	El-Salvador	37
Philippines	16	Trinadad&Tobago	38
Italy	17	Peru	39
Poland	18	Hong	40
Jamaica	19	Holand-Netherlands	41
Vietnam	20	?	42
Mexico	21		
Portugal	22		