***Abstract*—**

I. Results

The following section details three experiments using the PTCN to predict the sentiment of the text on a binary label of positive and negative.

A. Movie Script Sentiment

The movie script dataset is contained individual /.txt files which are imported as a list of data frames resulting in 8280 samples. Each data frame includes the movie title and movie script text. Both variables are character types. The original data frames are difficult to process for further modeling. Merging the individual data frames into a single data frame insure ease of processing later on. The movie script text variable labeled *"text"* is processed to remove noise current in the original sample texts. After performing a conversion of all lettering to lower casing, the removal of stop words, special characters, NA values, punctuation, numbers and whitespace from the texts. After the text is pre-processed the sentiment of each movie script needs to be measured. Each document using the pre-processed form is set to capture each individual words polarity score and than the documents overall polarity measurement. The polarity measures are conditionally converted into a binary labels.

Since DL models are susceptible to forming bias when trained on data that is not equally distributed across sample data for modeling requires a reduction in the sample size. The initial count of positive and negative sentiment labels is shown in Table I. An unequal distribution of the classes will hinder the PTCN in recognizing the most signifcant features. Balancing the number of positive samples and negative samples is key for the PTCN to recognize features with mitigated bias.

TABLE I

Number of Positive and Negative sentiment Labels   Movie Scripts

0	3828
1	4452

The new count of positive and negative sentiment labels results in Table II. A equal distribution of classes helps on its own accord in controlling the bias the PTCN results in for either class. Ensuring the balance of the labels in the training data mitigates bias from forming during the training of the C-LSTM. The now equally distributed samples are further processed for modeling. The texts are tokenized resulting in a list of word vector spaces

The inputs are not the same shape due to the various length sizes of the text representations. To help reduce further overfitting issues the tokenized word vector spaces are padded. The texts are padded to ensure that all inputs are set to the same length. Now the next process changes for the hyper-tune modeling and the best model including the cross-validation implementation. The hyper-tuning model splits the training, validation, and evaluation into fixed splits with no randomization. For instance, the training samples consisted of 7500 tokenized text samples and evaluation consisted of 100 tokenized text samples. The labels were split the same fashion. The text inputs and the label outputs are further transformed into matrices as depicted below. After, hyper-tuning sessions were enacted and the best parameters are identified the final model is captured.

TABLE II

Equal Number of Positive and Negative sentiment Labels Movie Scripts

0	3800
1	3800

The implementation of the model using 10-fold cross validation averaged 76.711% evaluation accuracy with a standard deviation of 10.27. The best model performed at 83.82% evaluation on fold 5 as depicted below in in Figure 1. The minimum evaluation accuracy is reported at 50% on fold 10. The remaining folds performance mesasures are depicted in III. The learning curve in the figure above suggests that the parameters are well trained.

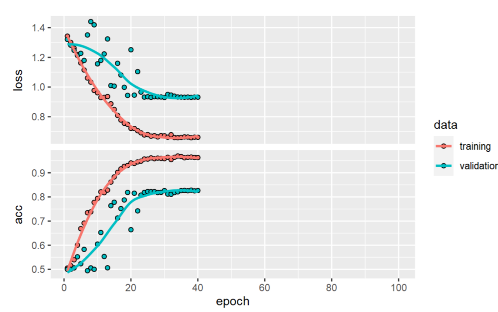

Fig. 1: PTCN performance on fold 3 ~ Movie Scripts

The training/validation plots indicate that each follows a similar pattern which suggests that the problem of overfitting could easily be handled. Even, though the evaluation accuracy shows significant variance across the folds it provides evidence of the robustness of the algorithm. No matter the input of movie script texts in the training, validation, and evaluation datasets the algorithm performed well. To compare the performance of the PTCN to other approaches three experiments will be performed with the Twitter Airline dataset, Sentiment 140 dataset, and IMBD Movie Review dataset.

TABLE III

PTCN evaluation of all folds Movie Scripts

	accuracy	precision	recall	F1
fold1	83.16	83.25	83.03	83.14
fold2	72.76	73.57	71.05	72.29
fold3	82.69	82.74	82.63	82.69
fold4	80.46	80.50	80.39	80.45
fold5	83.82	83.82	83.82	83.82
fold6	82.11	82.19	81.97	82.08
fold7	74.54	74.97	73.68	74.32
fold8	74.54	74.00	75.66	74.82
fold9	83.03	83.03	83.03	83.03
fold10	50.00	50.00	50.00	50.00

B. Twitter Airline Sentiment

The sample of Tweets contained in the Twitter Airline /.csv file are imported as a data frame shown below in Figure. The Tweets/text are pre-processed and re-combined with the classes in a dataframe depicted in Figure. The same sentiment process is not performed on the Tweets as the data provided comes with the sentiment classified. The sample size contains a total … tweets from various Twitter users. Each sample tweets are pre-processed using the aforementioned techniques discussed in **??**. The baseline model is the RNN at 90.45% and a close second the CNN at 90.37% accuracy rate [@Dang2020SentimentAB]. The PTCN's comparable performance is shown in Table IV.

TABLE IV

PTCN evaluation of all folds Twitter Airline

	accuracy	precision	recall	F1
fold1	52.97	53.02	52.12	52.66
fold2	88.56	88.72	88.35	88.54
fold3	84.22	84.29	84.11	84.20
fold4	88.98	88.82	89.19	89.01
fold5	87.92	88.09	87.71	87.90
fold6	89.72	89.81	89.62	89.71
fold7	87.71	87.55	87.92	87.74
fold8	89.09	89.17	88.98	89.08
fold9	90.89	90.89	90.89	90.89
fold10	62.39	62.53	61.86	62.19

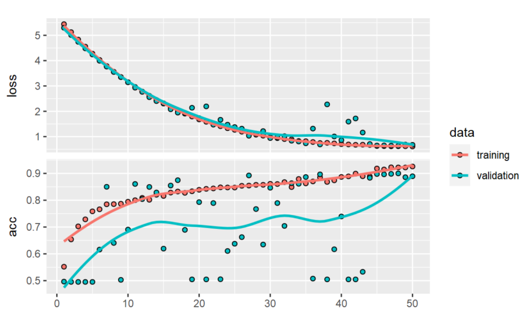

Fig. 2: PTCN performance on fold 9 ~ Twitter Airline

The implementation of the model using 10-fold cross validation averaged 82.245% evaluation accuracy with a standard deviation of 13.25. The best model performed at 90.89% evaluation on fold 9 as depicted below in in Figure 2. The minimum evaluation accuracy is reported at 52.97% on fold 1. The remaining folds performance mesasures are depicted in IV. The learning curve in the figure above suggests that the parameters are well trained.

C. Sentiment 140

The sample of Tweets contained in the Sentiment 140 /.csv file are imported as a data frame shown below in Figure.

D. IMBD Sentiment

To test the comparable performance of the PTCN performing sentiment analysis, the model is trained on the IMBD movie review task. The baseline model for the IMBD is 89.37% established by [@Zulqarnain2020AnID].