CCU Spring Semester Report

Dhaarna Maheshwari
dm3792@columbia.edu

Introduction

The Columbia DARPA Computational Cultural Understanding project aims to understand how to predict communication change or failure given data from multiple modalities and featuring an arbitrary natural language. We perform our research on languages such as Mandarin, Chinese and other non-English languages. We aim to develop accurate models for predicting conversational outcomes that are language agnostic.

Work done this semester

This report is a summary of all the work done by me during the spring semester 2023. Our main aim was to detect changepoints (abrupt changes in emotion) in a conversation. I have tried to do so by first establishing the validity of the hypothesis that change points in communications might be closely related to norm violations and adherences in the nearby segments and then building an XLM-Roberta for detecting change points using norms. From the data analysis, we can see that our hypothesis was indeed very true and hence we proceeded with building models for changepoint detection using norm violations and adherences in the nearby segment along with the text in the segment as our input. We have a lot of scope for future work where we train more models, compare them and their different variants.

PART1 : Data Analysis

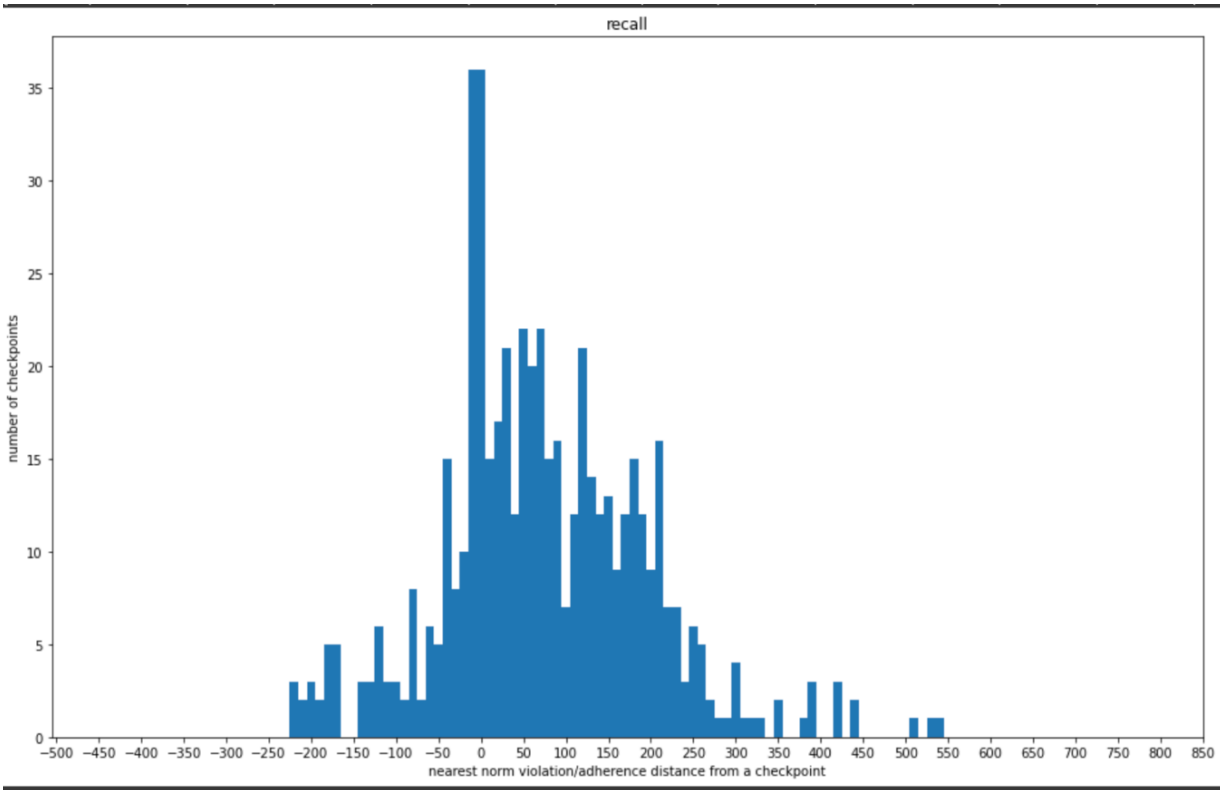
We had the LDC andUIUC data for detected changepoints and norm violations/adherences in segments of files.To check the correlation between the norm violations/adherences and changepoints we did a bit of data analysis. The main results of the data analysis is that it supports our hypothesis that a changepoint occurs in close vicinity to norm violations/ adherences.

All the different graphs that we plotted to analyse the correlation have been included in the mid sem report which is also present in the github repo link mentioned towards the end of this report.. We tested different document types, norm types, violations and adherences separately to make sure the hypothesis holds in all these cases.

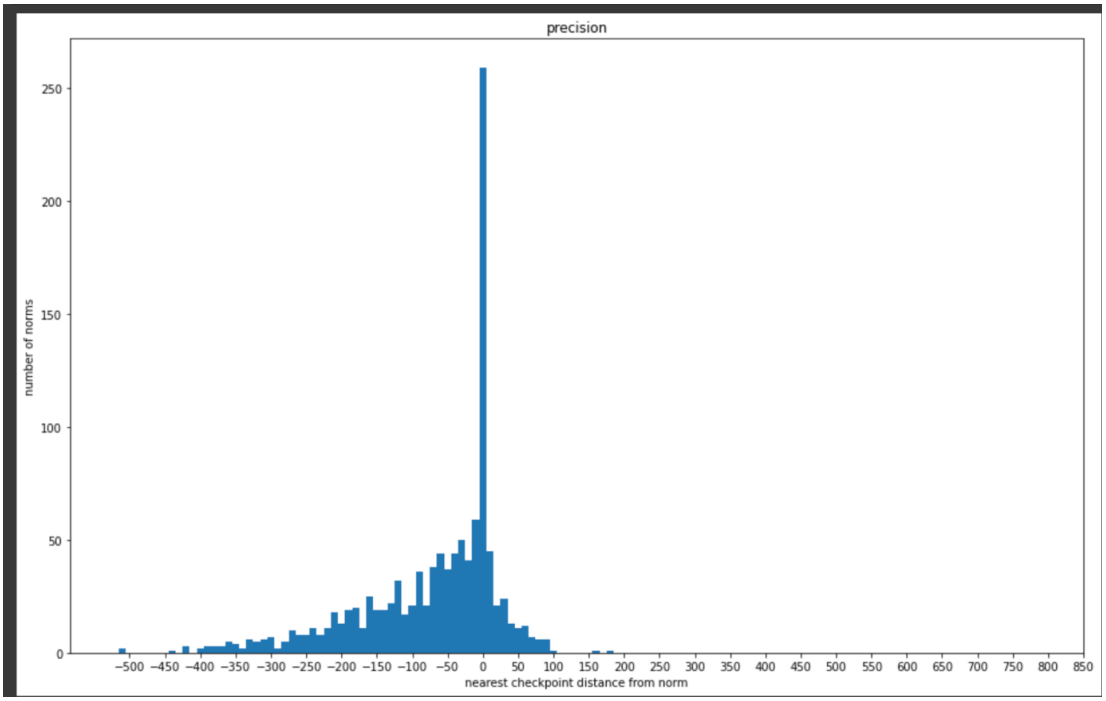
Steps done:

0. Plotted change points and segments with norm violations/adherences to get a basic idea.

1. Recall curve for the entire LDC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.

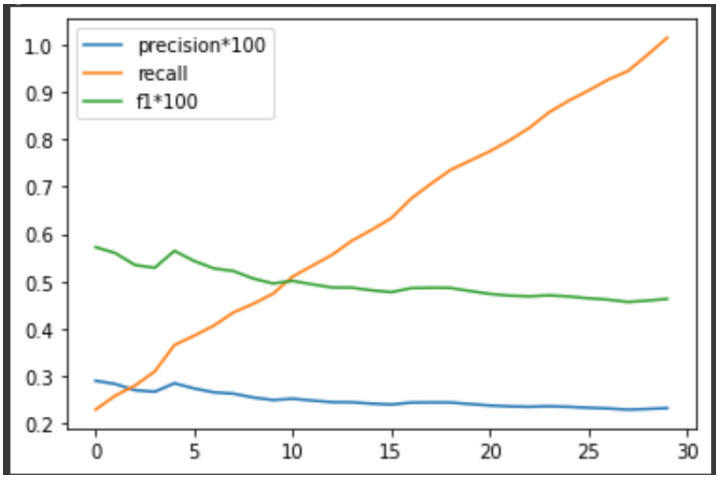


2. Precision curve for the entire LDC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.

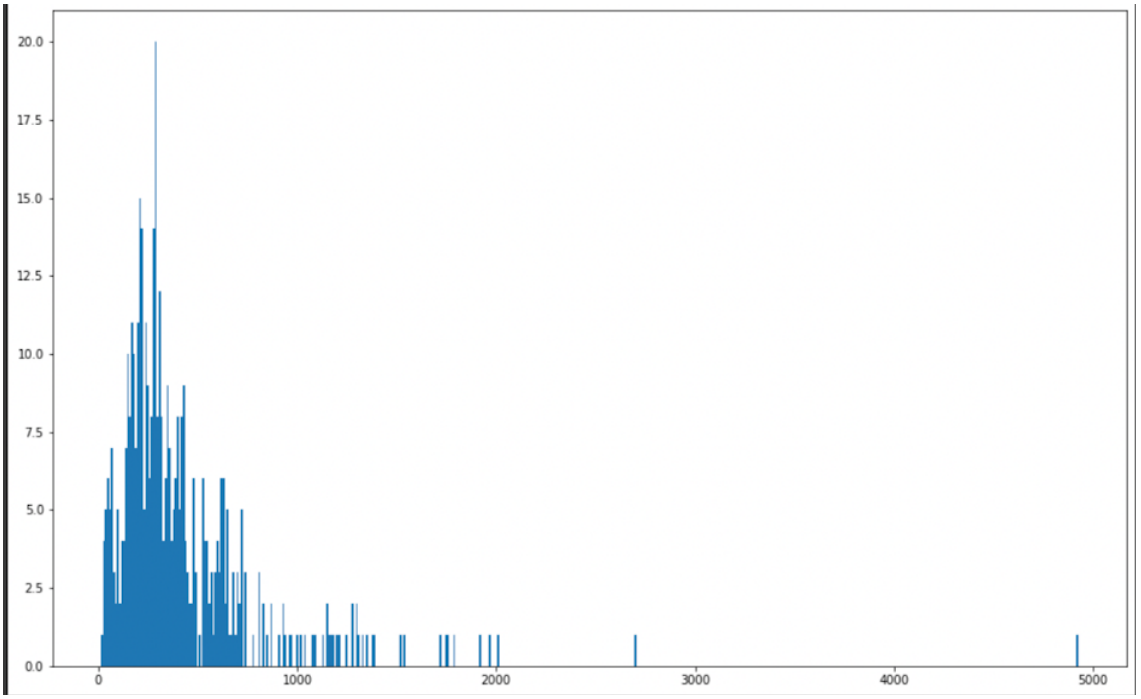


3. Plotting precision, recall and f1 score for different values of d where we predict

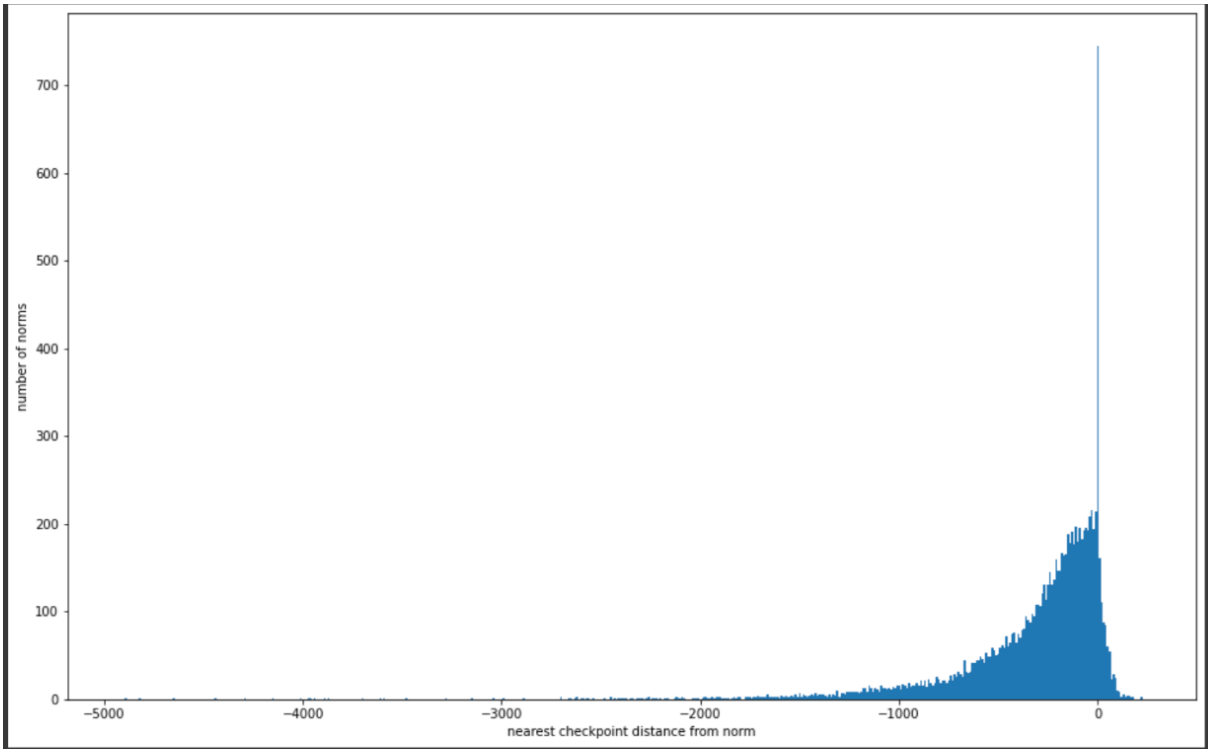
timestamps in the vicinity(+-d) of a norm violation/adherence as a changepoint.



4. Recall curve for the entire UIUC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints. [nearest norm violation distance vs. number of checkpoints]



5. Precision curve for the entire UIUC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.



6. For both LDC and UIUC data: Precision, Recall and PR- F1 score curves for different file types : flac.ddc,.psm.xml,.mp4.ldcc
7. For both LDC and UIUC dataPrecision, Recall and PR- F1 score curves for norm violations and adherences separately.
8. For both LDC and UIUC data Precision, Recall and PR- F1 score curves for different norm types. 101 : apology,102 : criticism,103 : greeting,104 : request,105 : persuasion,106 : thanks,107 : leave and 50 other norms

Conclusion from Data Analysis and baseline model

After observing the graphs above, we can see that norm changes and violations occur in close vicinity to changepoints.We also tested the precision, recall and F1 score on a basic baseline model where we declared any point with a norm violation/adherence within +-d distance as a changepoint. We then calculated the precision, recall and F1 score for different values of d. For values near 0 we got a high F1 score again strengthening the validity of our hypothesis.

PART 2 : Training the model

Now, we know that our hypothesis that changepoints are closely related to norm violations/adherences holds true for different document types and norms so we will be building models with input as our segments data and norm violations/ adherences data and detect if there is a changepoint in the segment.

We then used a big pretrained language framework like RoBERTa for understanding the relation between the norm violation/adherence for different norm types and detecting the

changepoint. RoBERTa stands for Robustly Optimised BERT Pretraining Approach. While BERT provided an impressive performance boost across multiple tasks it is undertrained in comparison to XLM Roberta. We use the multilingual version of RoBERTa - XLM Roberta in our experiment as our utterances are in Chinese.

We have built an XLM RoBERTa model with the following properties:

- Uses the Adam Optimizer
- Generalised Cross Entropy loss
- Metrics, accuracy, precision, recall, average precision using the NSIT scorer
- Supports LR scheduling
- Checkpointing
- Downsampling
- Early stopping
- Configurable number of layers in the classifier
- Configurable number of utterances before/after to be included in the input
- Whether or not utterances are included in the input
- Regularisation
- Using norms with high confidence(llr) only

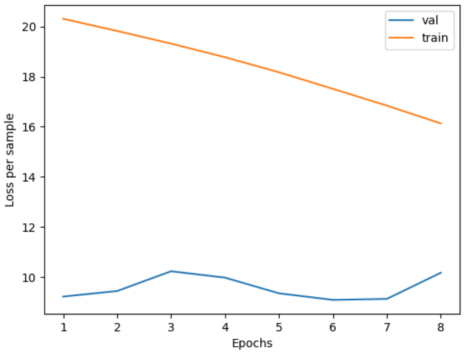
The input (generated by combining the LDC data we get from the data loader and UIUC norms) format for the model is as follows:

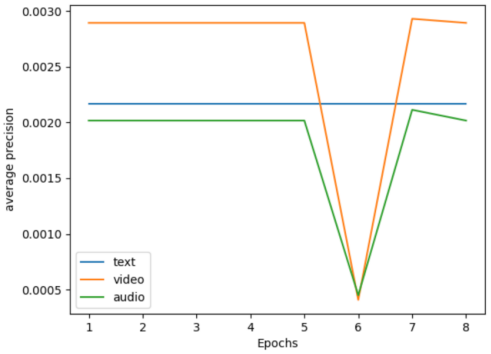
```
{
 "file_id": the source LDC file,
 "timestamp": , # the timestamp for the central utterance
 "utterance": (join the utterances before, the central utterance, and the utterances after), (string)
 "norms": (list of norm names that are adhered / violated across the utterances)
                Ex. "ADHERED:GREETING, VIOLATED:APOLOGY, etc" (string)
"label": whether or not there's a changepoint (integer, 0 or 1)
}
```

I trained different variations of the XLM Roberta model with input as the utterances and norm violations/adherence. I ran a grid search with various combinations of parameters (initial learning rate, utterances include/ not include, how many to include, regularisations, downsampling, with/without lr scheduler, number of classification layers, using confident norms only) to check the average precision of the model and choose the best variant.

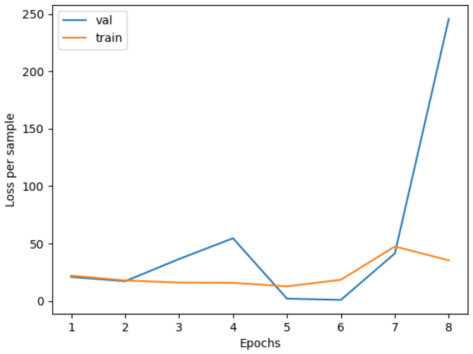
Results

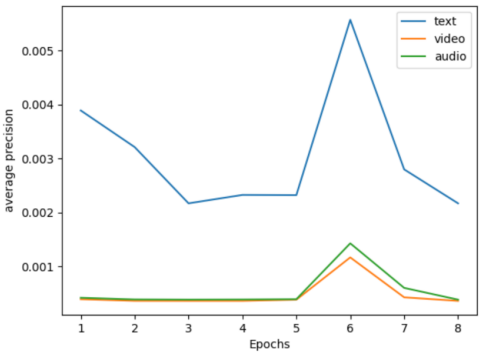
1. regularisation= L2, **learning rate = 1e-05**, **include utterance= False**, downsample= 2, lr scheduler= False, classifier layers= 1, confident only = False



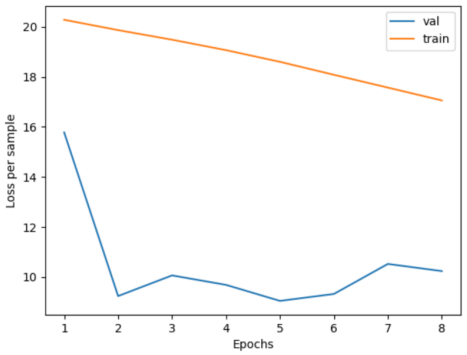


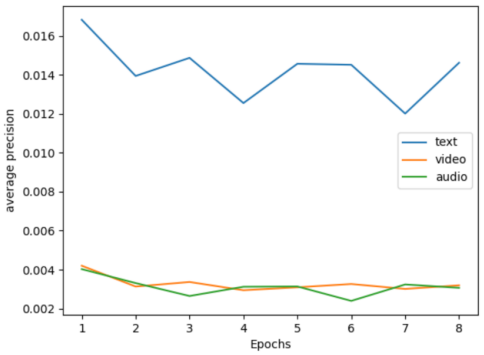
2. regularisation= L2, **learning rate = 0.1**, **include utterance= True**, downsample= 2, lr scheduler= False, classifier layers= 1, confident only = False



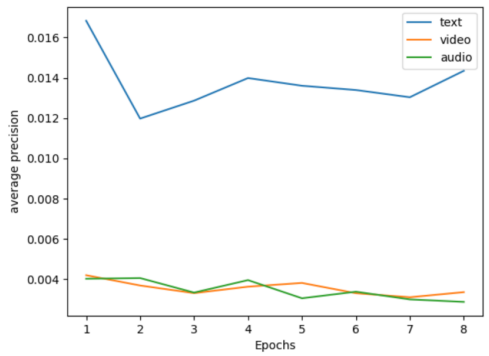


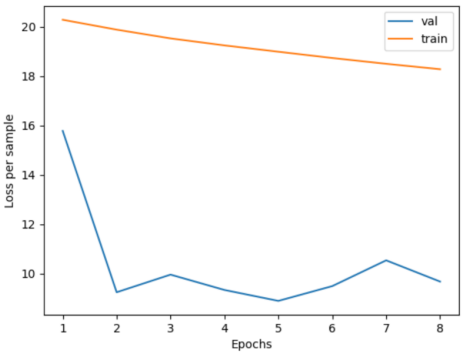
3. regularisation= L2, **learning rate = 1e-5**, **include utterance= True, downsample= 4,** lr scheduler= False, classifier layers= 1, confident only = False



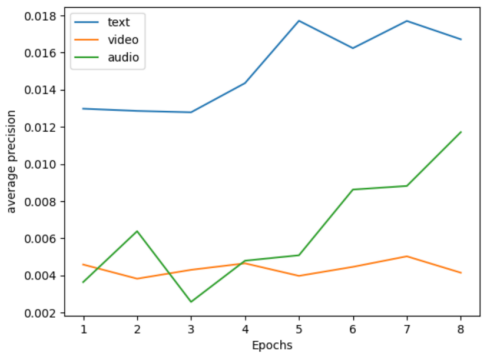


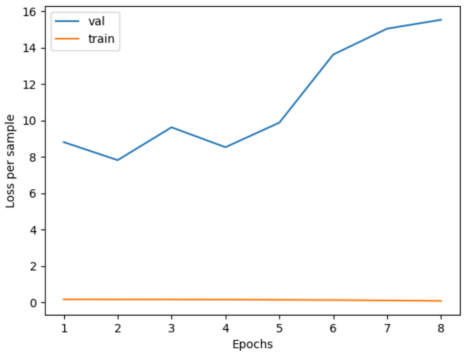
4. regularisation= L2, **learning rate = 1e-5**, **include utterance= True, downsample= 2,** lr scheduler= True, classifier layers= 1, confident only = False



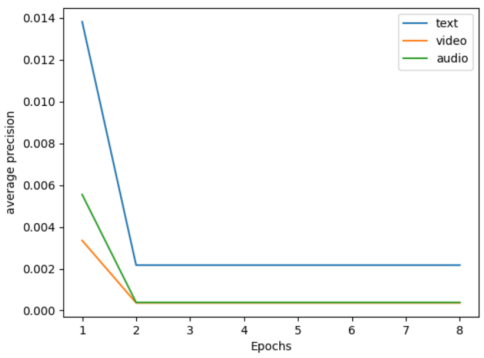


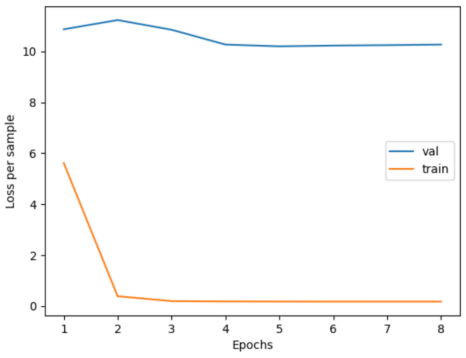
5. regularisation= dropout, **learning rate = 1e-5, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 1, confident only = False



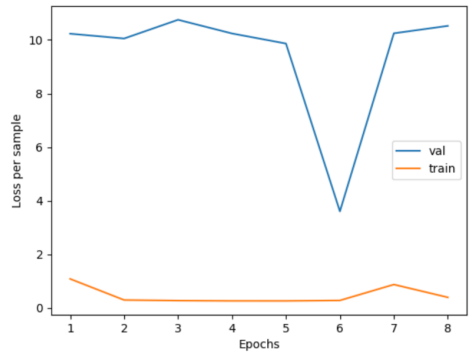


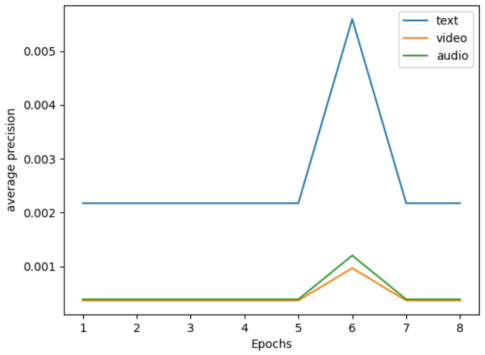
6.regularisation= L2, **learning rate = 0.001, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 1, confident only = False



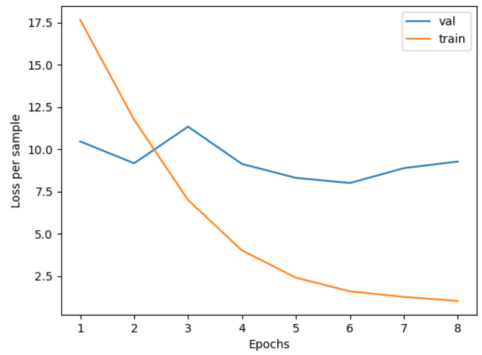


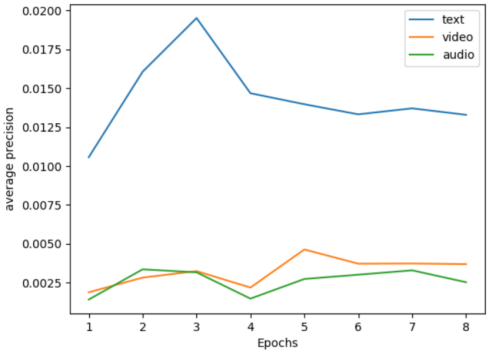
7.regularisation= L2, **learning rate = 0.01, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 1, confident only = False



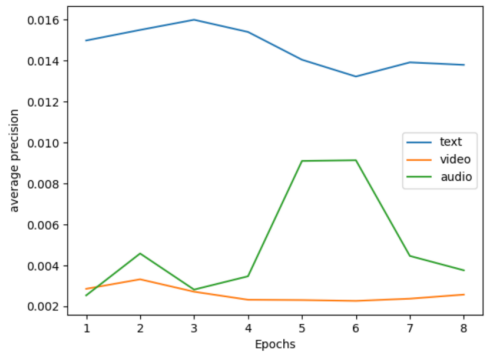


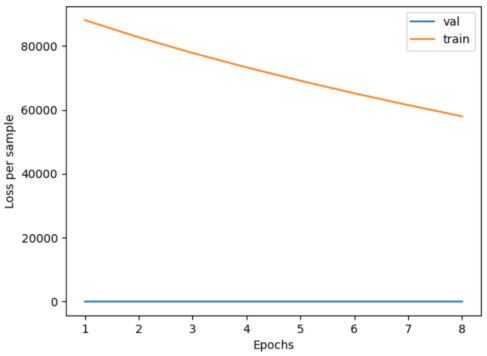
8.regularisation= L2, **learning rate = 0.0001, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 1, confident only = False



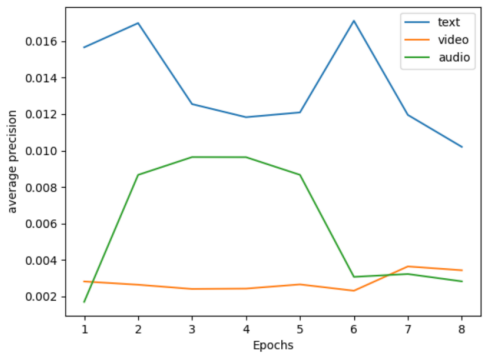


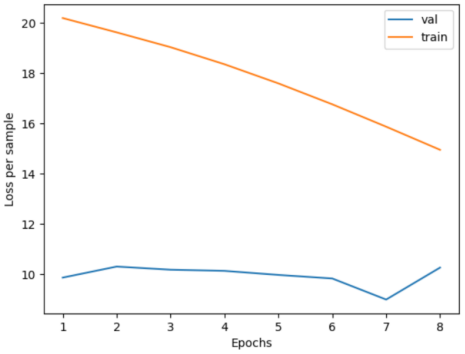
9.regularisation= L2, **learning rate = 1e-5, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 1, confident only = False





10.regularisation= L2, **learning rate = 1e-5, include utterance= True, downsample= 2,** lr scheduler= False, classifier layers= 2, confident only = False





Conclusion

We can see that dropout regularisation, low learning rate and including utterances is working out well for this model. But we can still see that the average precision is really low (max 0.02 for all modes text/audio/video) which means that either the hypothesis we made was faulty or we need to use different variants/ inputs/ features in the model.

To-dos for this model

- Using a subset of norms to find out if there is a set of norms that affects the changepoint detection more than others.
- Comparing this model to a simple norm counting logistic regression model.
- Checking out more variants in the grid search

Future work

These models can also be used to test the hypothesis in the future:

1. XLNet : The researchers from Carnegie Mellon University and Google have developed a new model, XLNet, for natural language processing (NLP) tasks such as reading comprehension, text classification, sentiment analysis, and others. XLNet is a generalised autoregressive pretraining method that leverages the best of both autoregressive language modelling (e.g., Transformer-XL) and autoencoding (e.g., BERT) while avoiding their limitations.
2. ALBERTA: The Google Research team addresses the problem of the continuously growing size of the pretrained language models, which results in memory limitations, longer training time, and sometimes unexpectedly degraded performance. Specifically, they introduce A Lite BERT (ALBERT) architecture that incorporates two parameter-reduction techniques: factorised embedding parameterization and cross-layer parameter sharing. In addition, the suggested approach includes a self-supervised loss for sentence-order prediction to improve inter-sentence coherence.

These two models have been known to perform well on sentiment analysis so we will try them out on our change point classification task and compare the results of all these models.

Git Hub link

References

1. [1907.11692] RoBERTa: A Robustly Optimized BERT Pretraining Approach
2. Using RoBERTA for text classification · Jesus Leal
3. A Review of Different Approaches for Detecting Emotion from Text - IOPscience.
4. [1906.08237] XLNet: Generalized Autoregressive Pretraining for Language Understanding
5. https://arxiv.org/abs/1909.11942v1
6. [1910.10683] Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer