CCU Spring Semester Report

Dhaarna Maheshwari
dm3792@columbia.edu

Abstract

This report is a summary of all the work done by me during the first half of the spring semester 2023. It primarily focuses on establishing the validity of the hypothesis that change points in communications might be closely related to norm violations and adherences in the nearby segments. This will help us further in building models that can detect change points in a conversation with a high accuracy. From the data analysis, we can see that our hypothesis was indeed very true and hence we proceed 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 our initial XML Roberta model as well as experiment with different models and parameters.

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.

Data Analysis

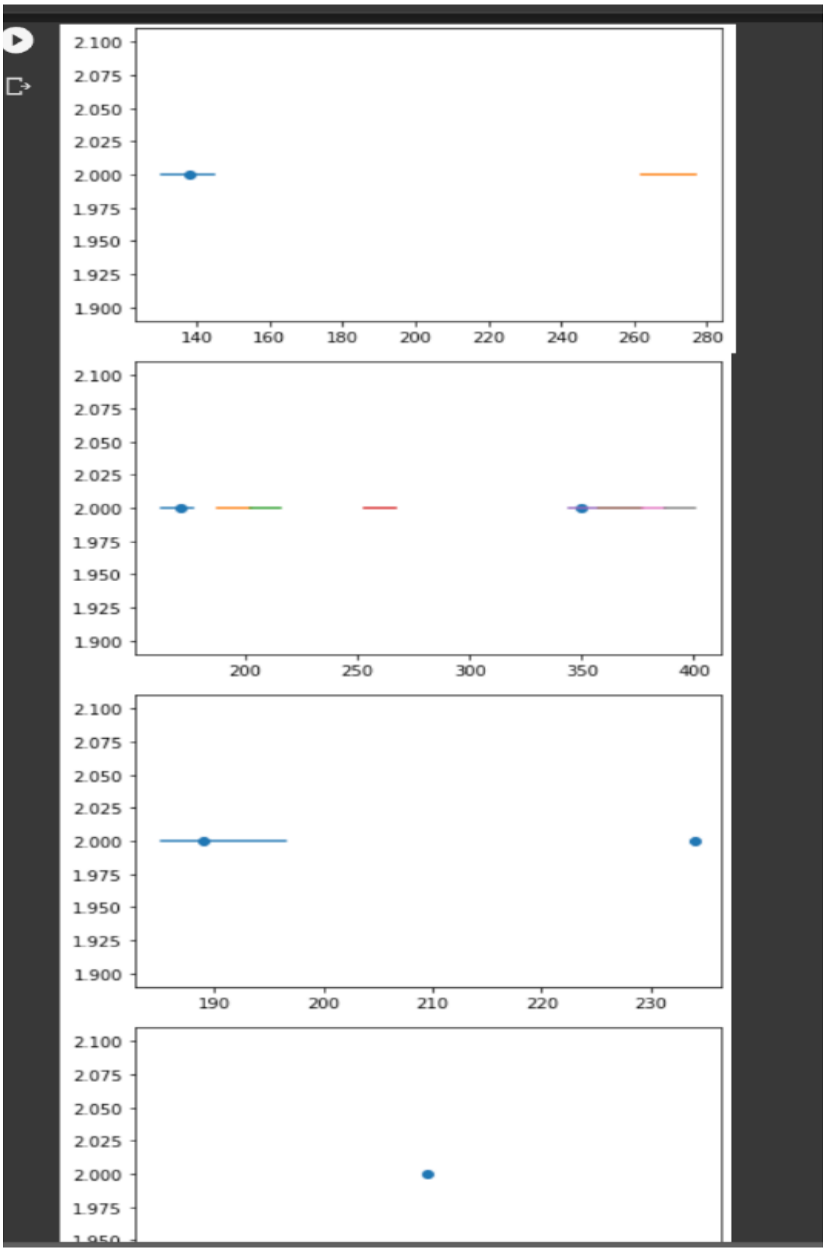
We had the LDC andUIUC data for detected changepoints and norm violations/adherences in segments of files.To check the correlation between the norm violations/adherences and changepoints we did a bit of data analysis. The main results of the data analysis is that it supports our hypothesis that a changepoint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

Github repo link with notebooks for the code for below data analysis - one for LDC one for UIUC.

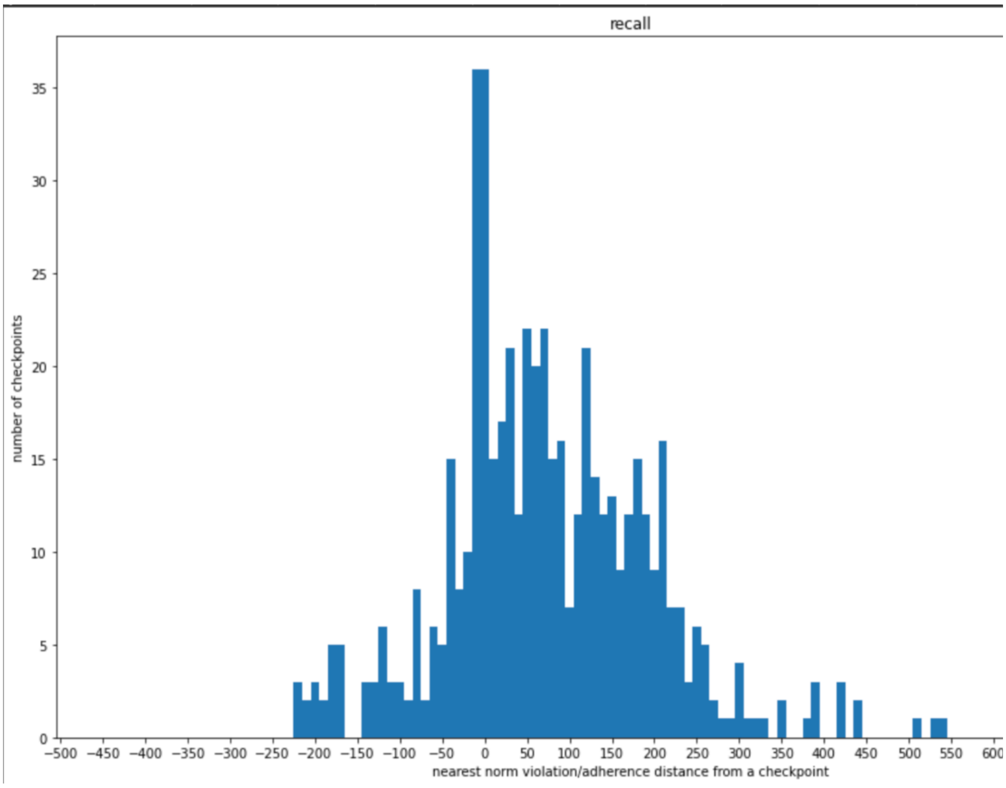
https://github.com/dm3792/ccu_Dhaarna/blob/main/LDC_data_analysis.ipynb
https://github.com/dm3792/ccu_Dhaarna/blob/main/LDC_data_analysis.ipynb

Below are the results of the different graphs we plotted to analyse the correlation. We even tested different document types, norm types, violations and adherences separately to make sure the hypothesis holds in all these cases.

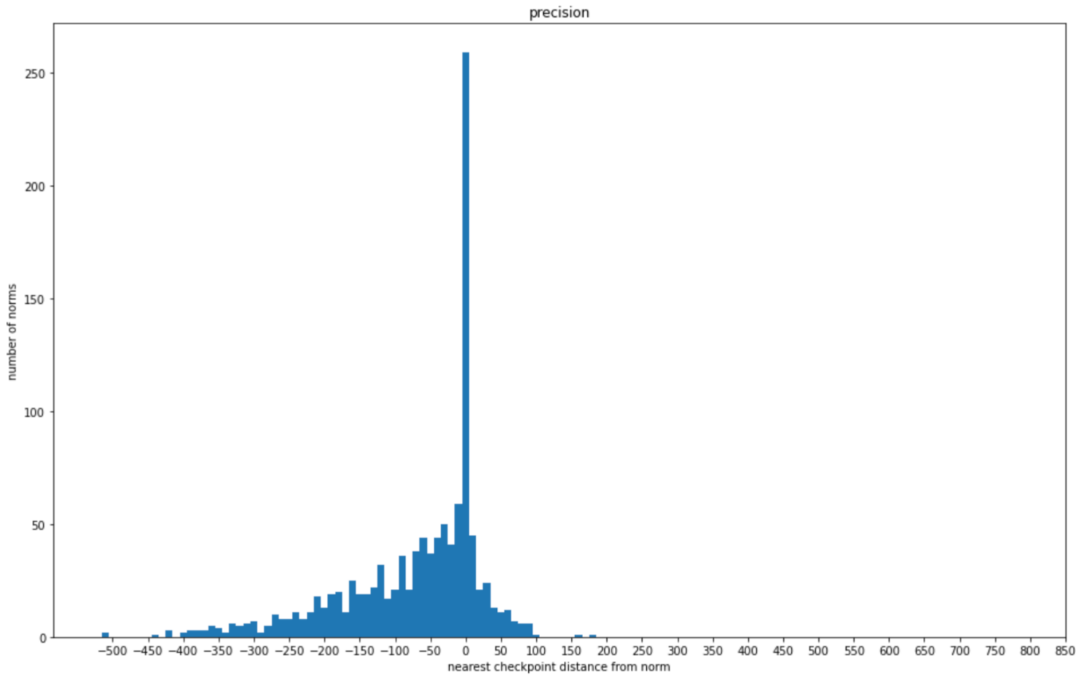
0. Plotted change points and segments with norm violations/adherences to get a basic idea.



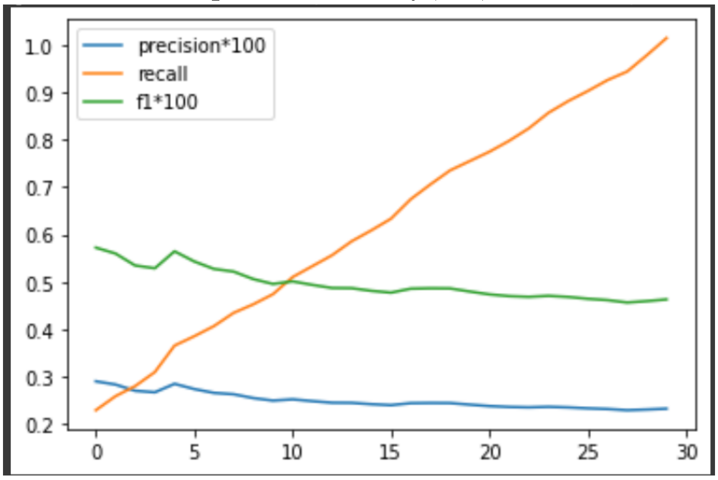
1. Recall curve for the entire LDC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.



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.

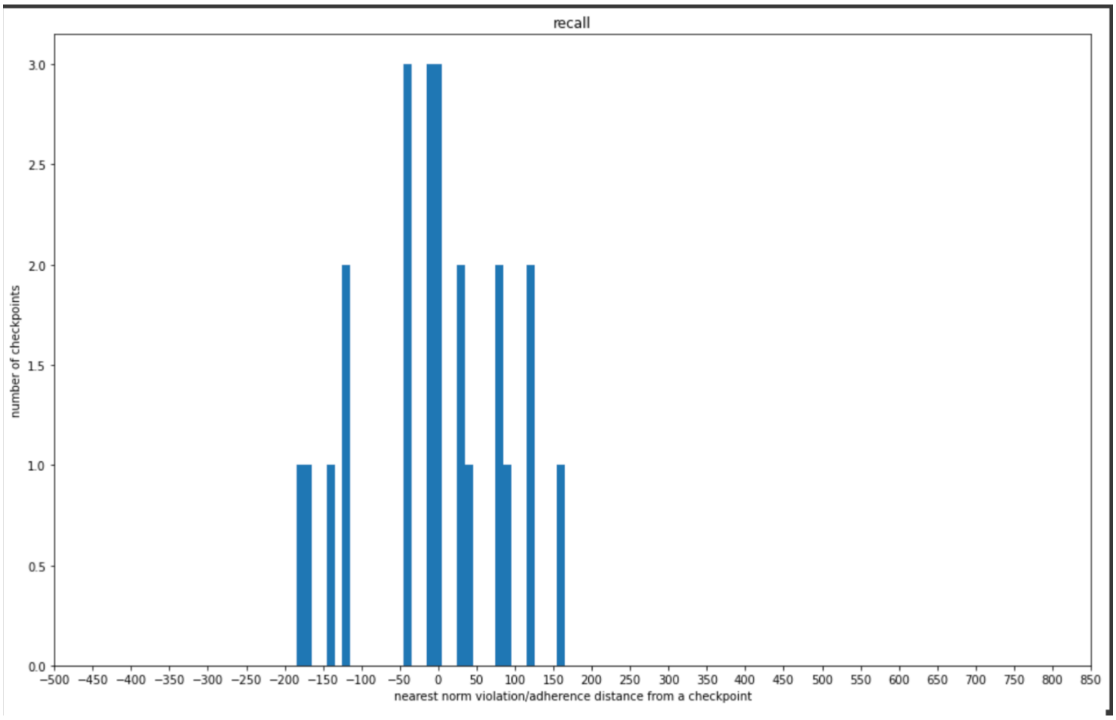


3. Plotting precision, recall and f1 score for different values of d where we predict timestamps in the vicinity(+-d) of a norm violation/adherence as a changepoint.

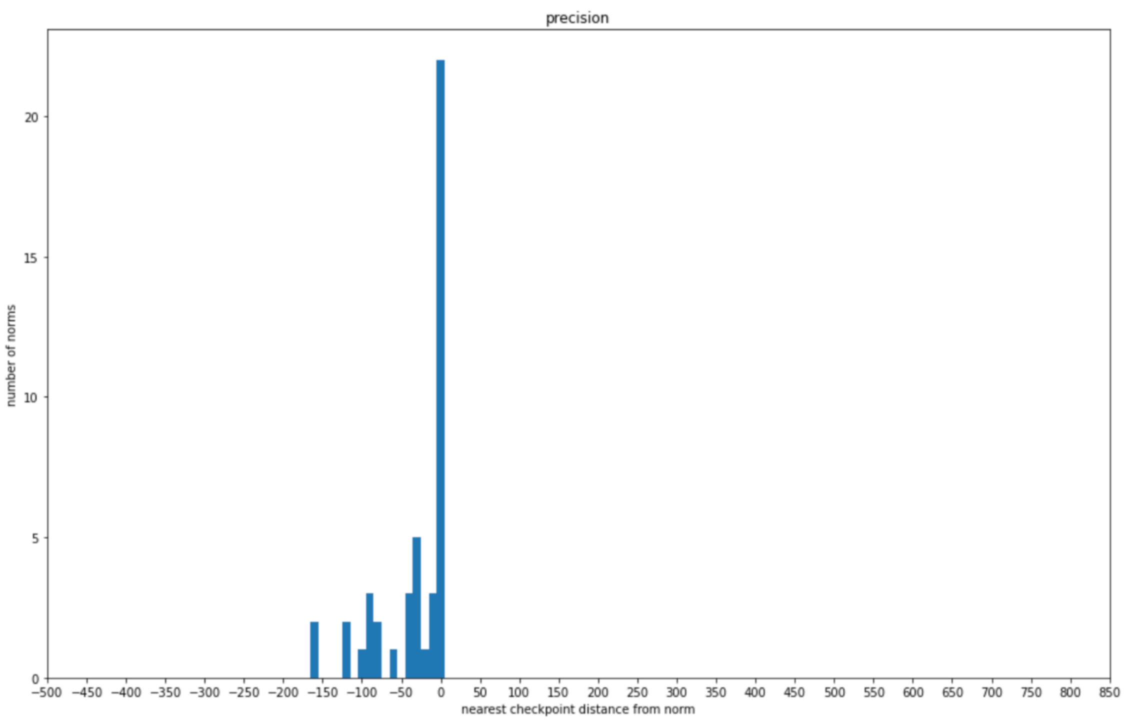


4. Precision, Recall and PR- F1 score curves for different file types.

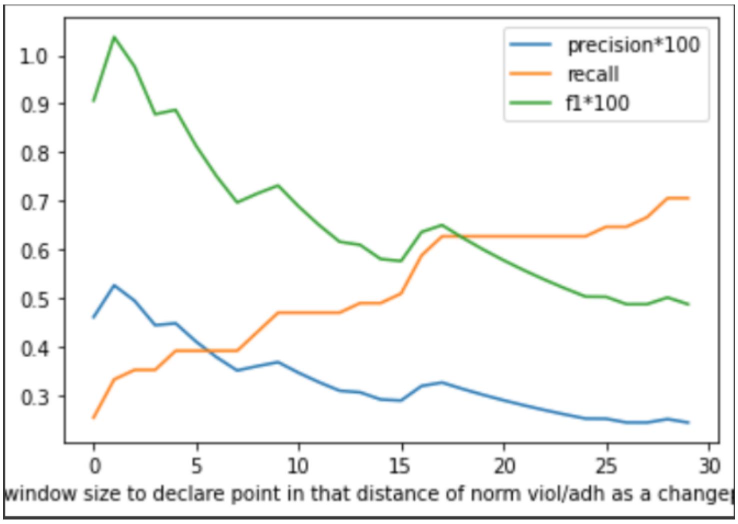
- .flac.ddc



recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

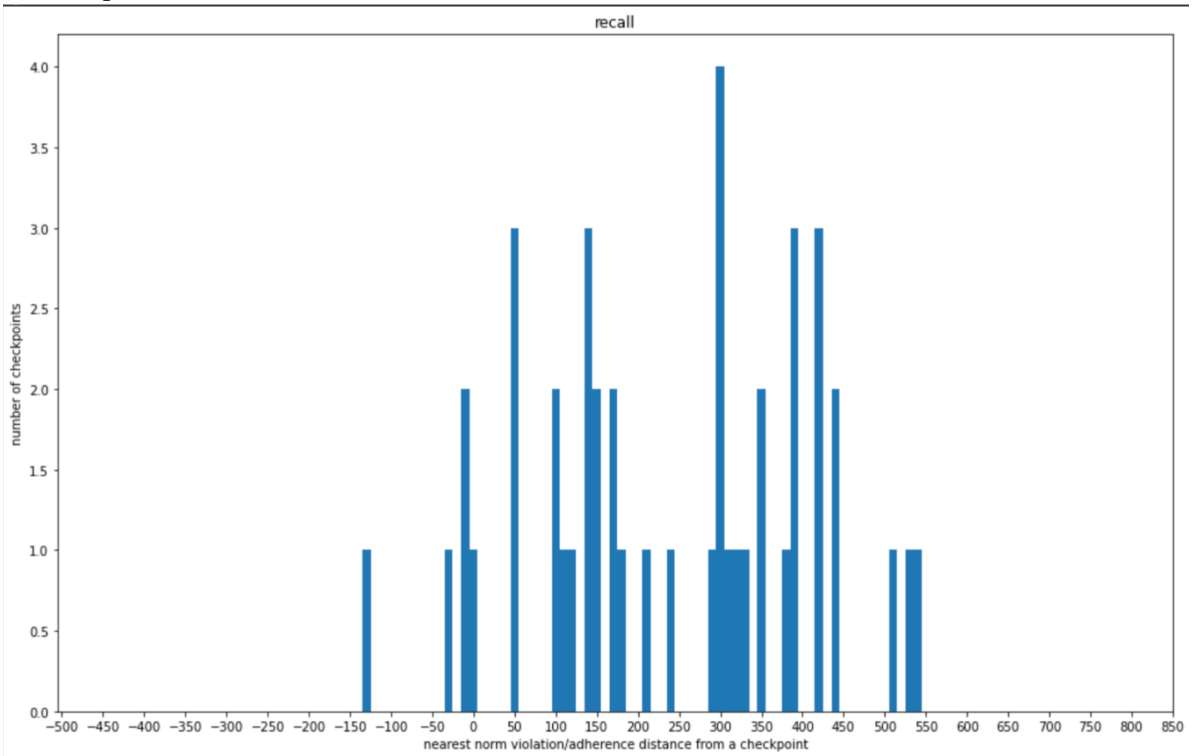


precision
number of norms
nearest checkpoint distance from norm

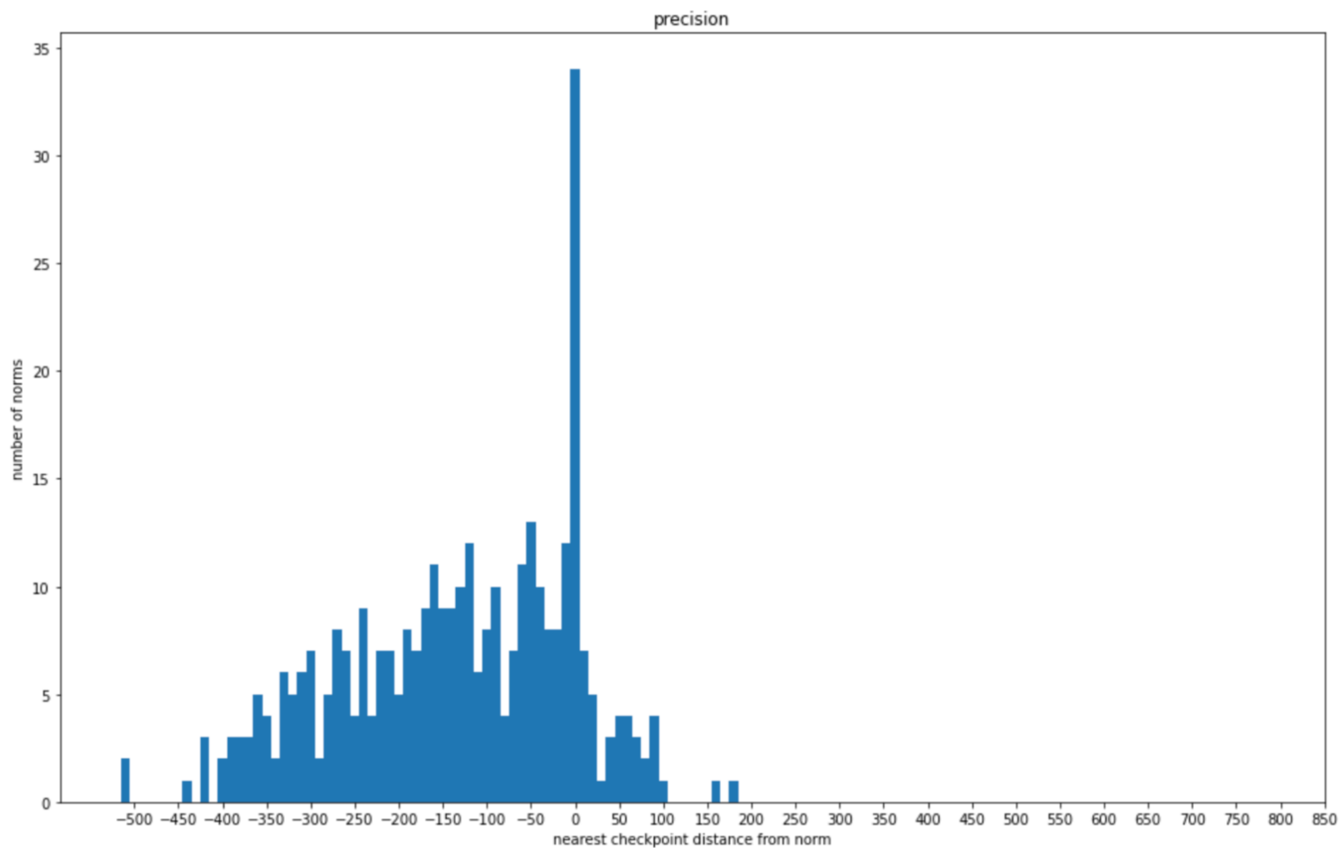


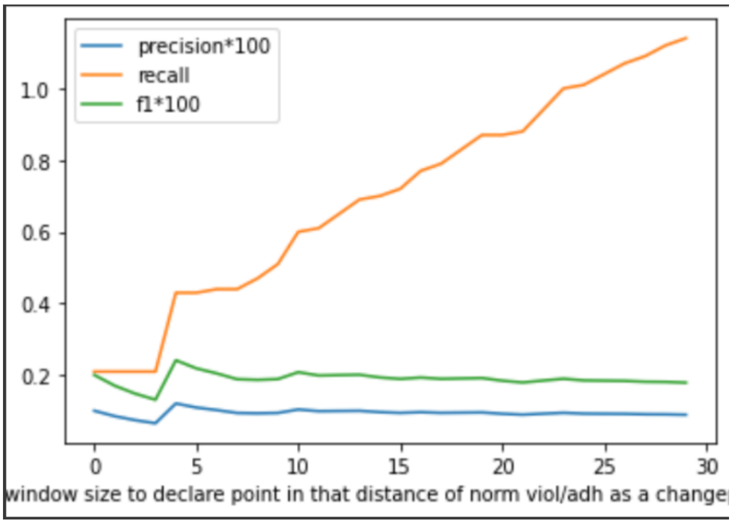
precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

- .psm.xml

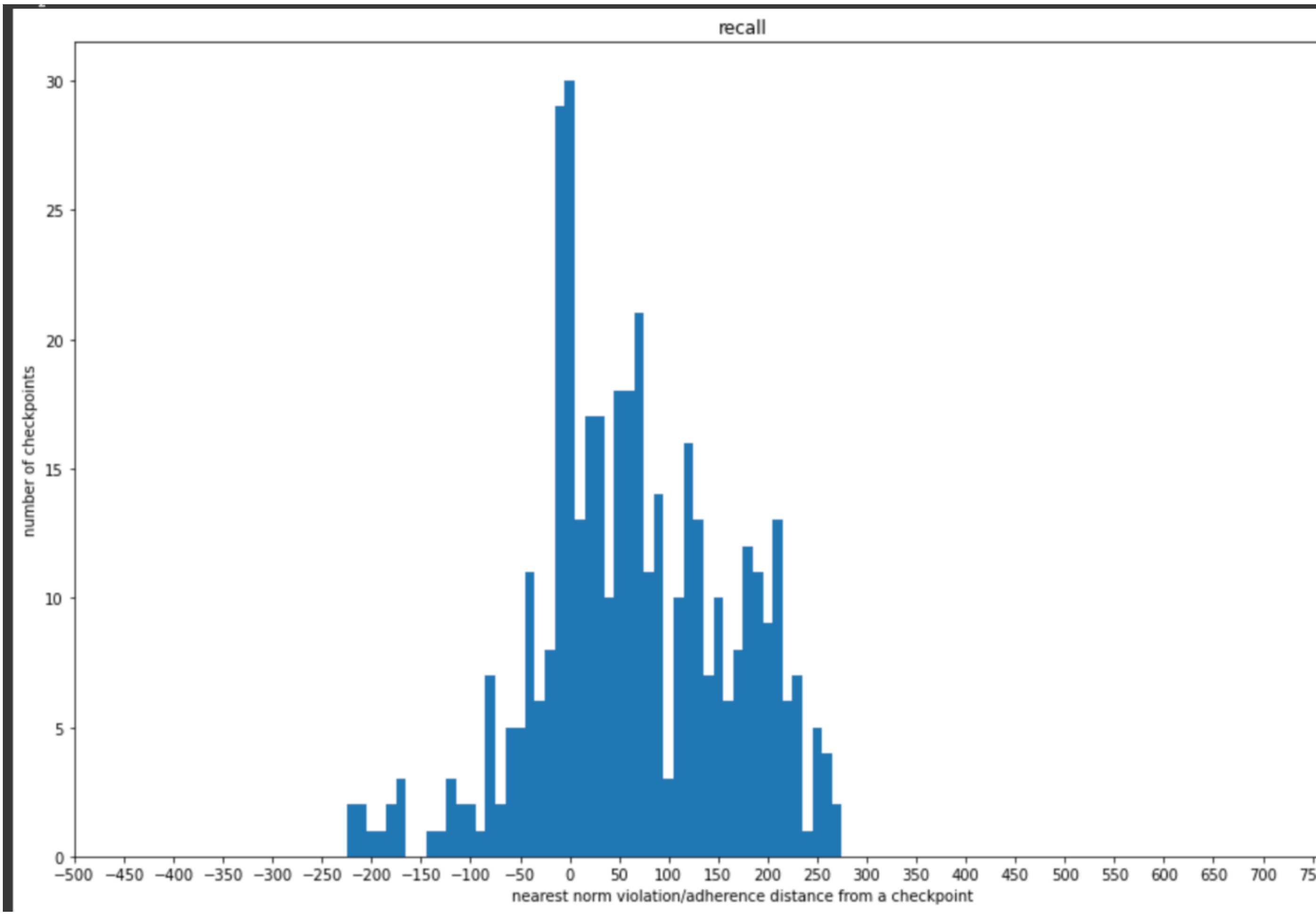


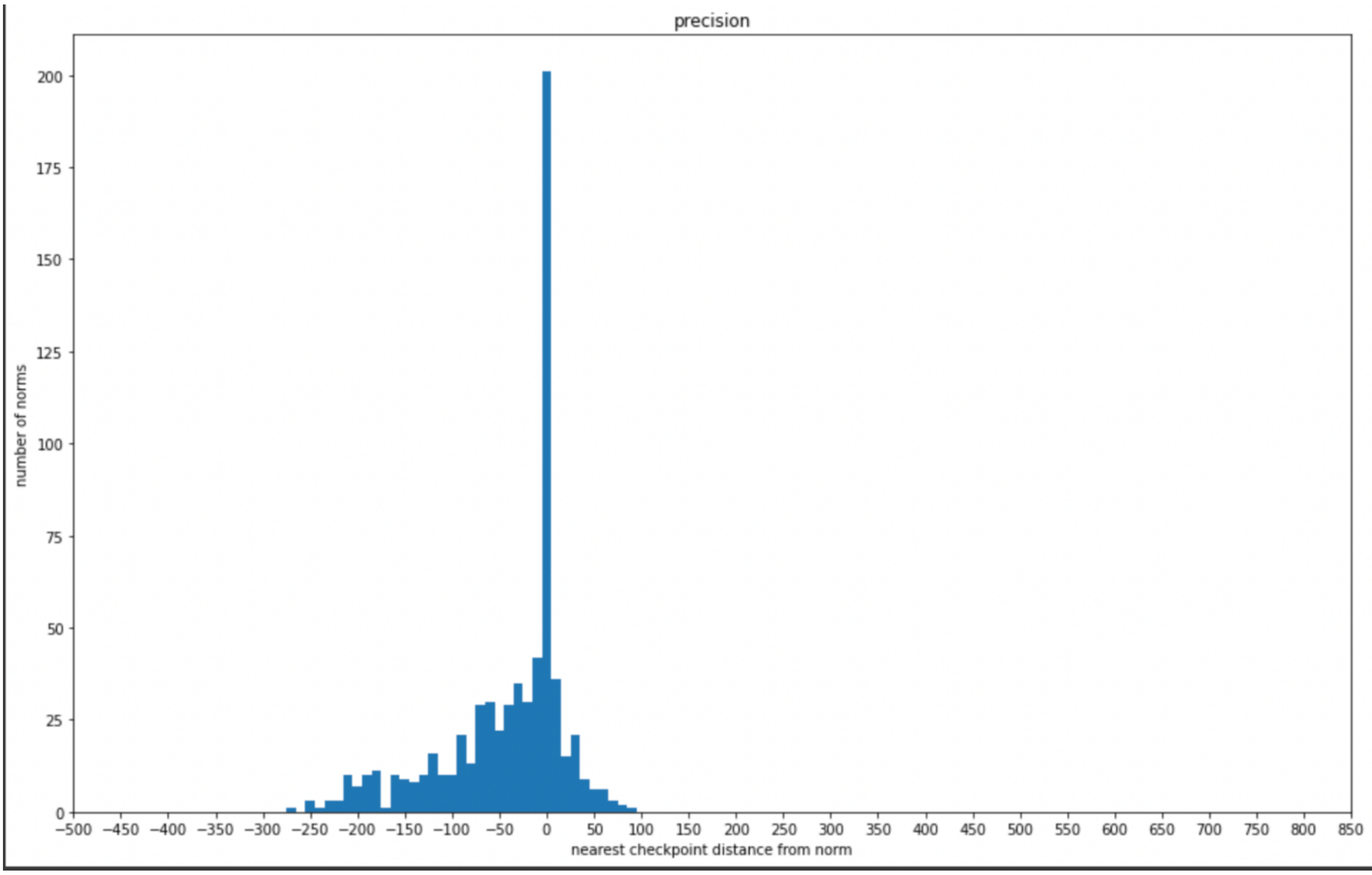
recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

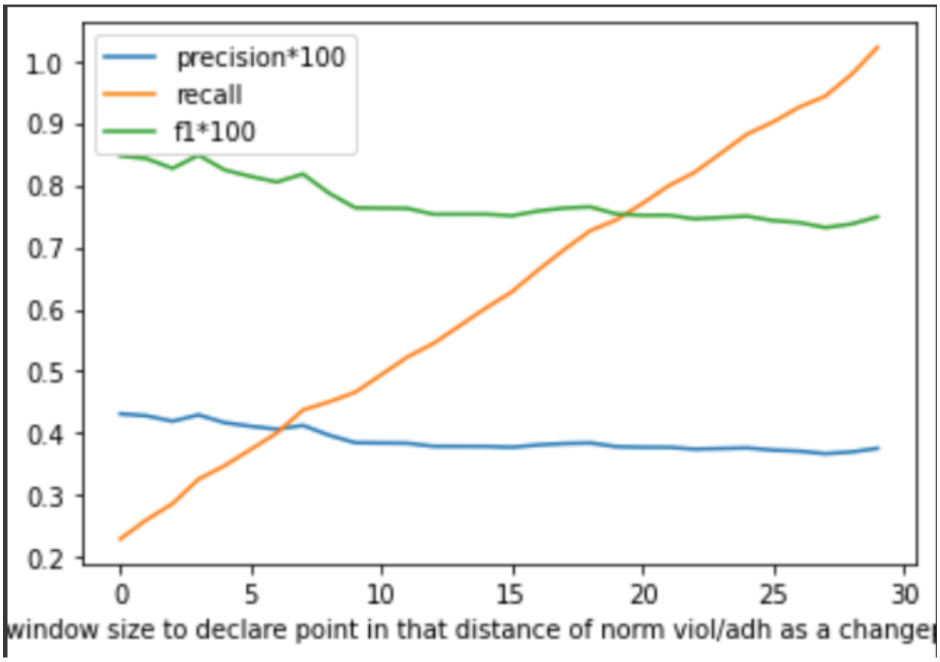

precision
number of norms
nearest checkpoint distance from norm


precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

- .mp4.ldcc

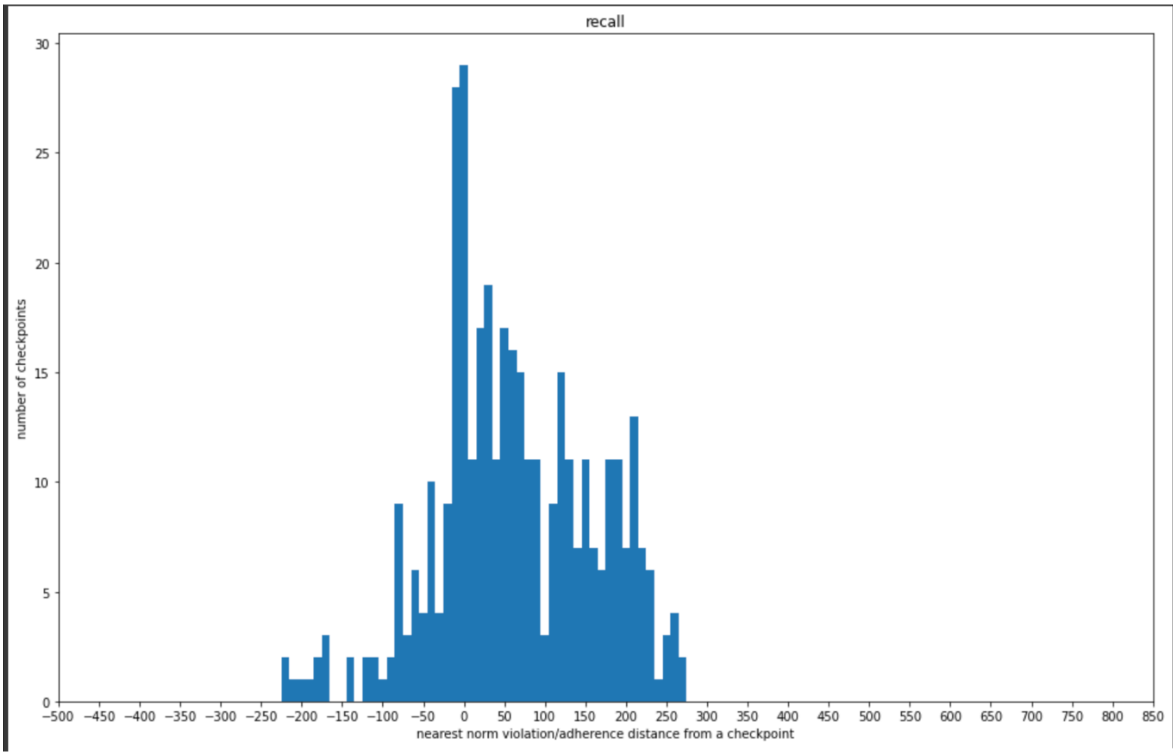

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

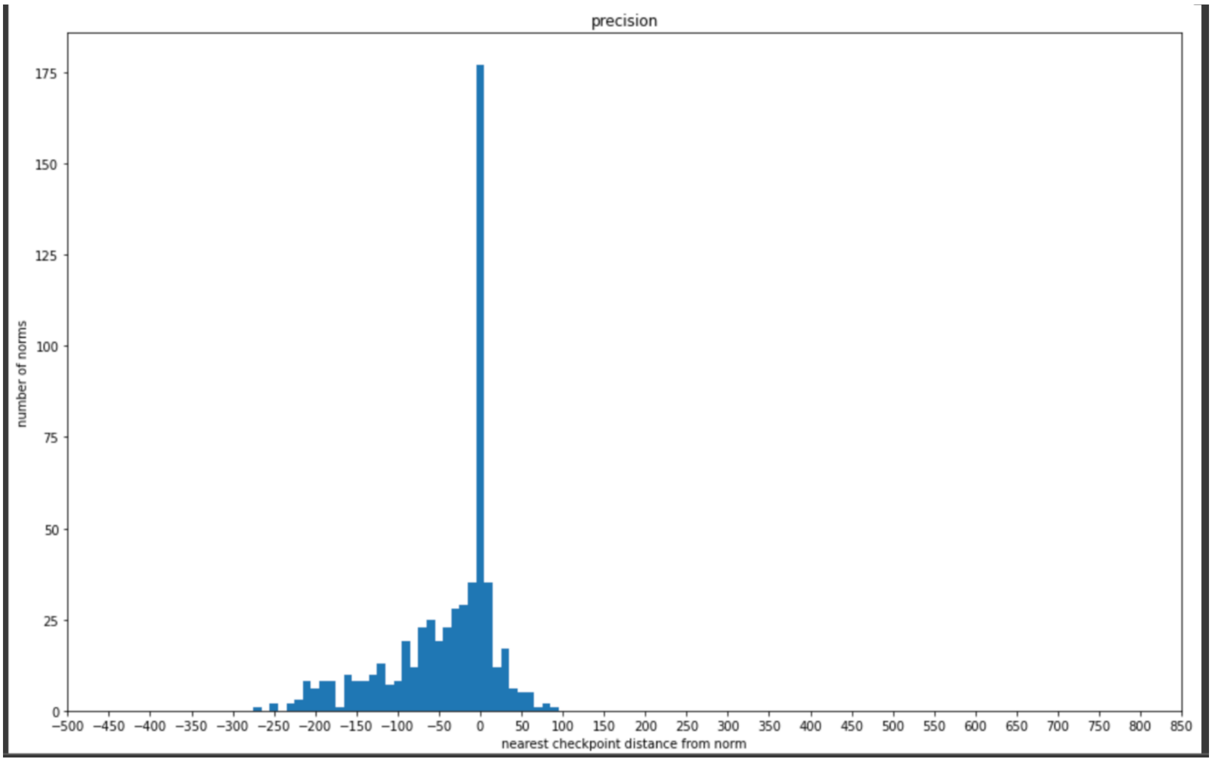


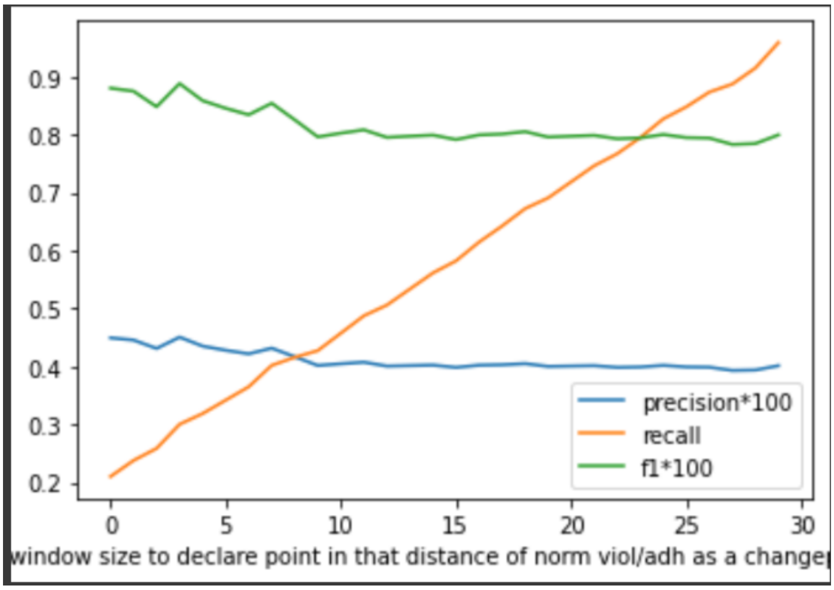


5. Precision, Recall and PR- F1 score curves for norm violations and adherences separately.

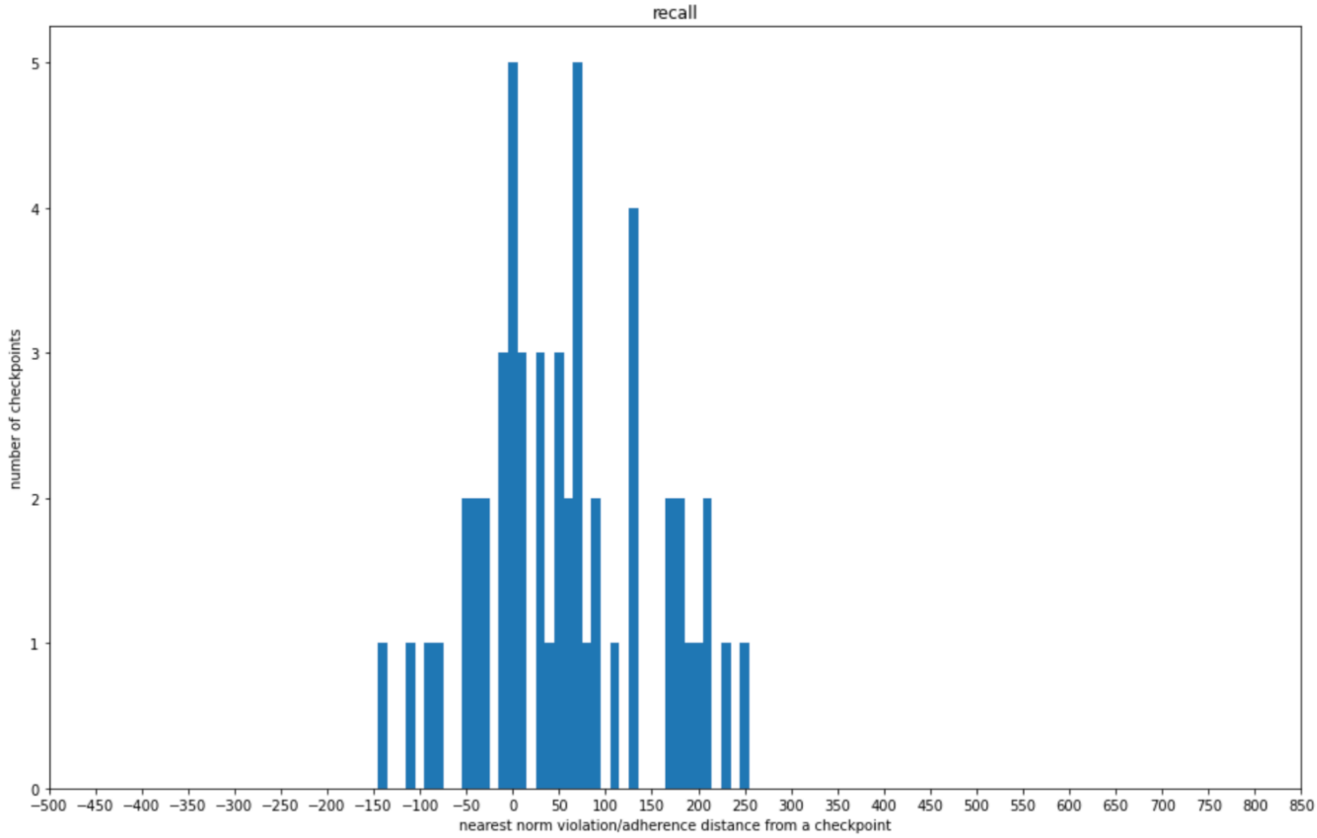
- Adhere

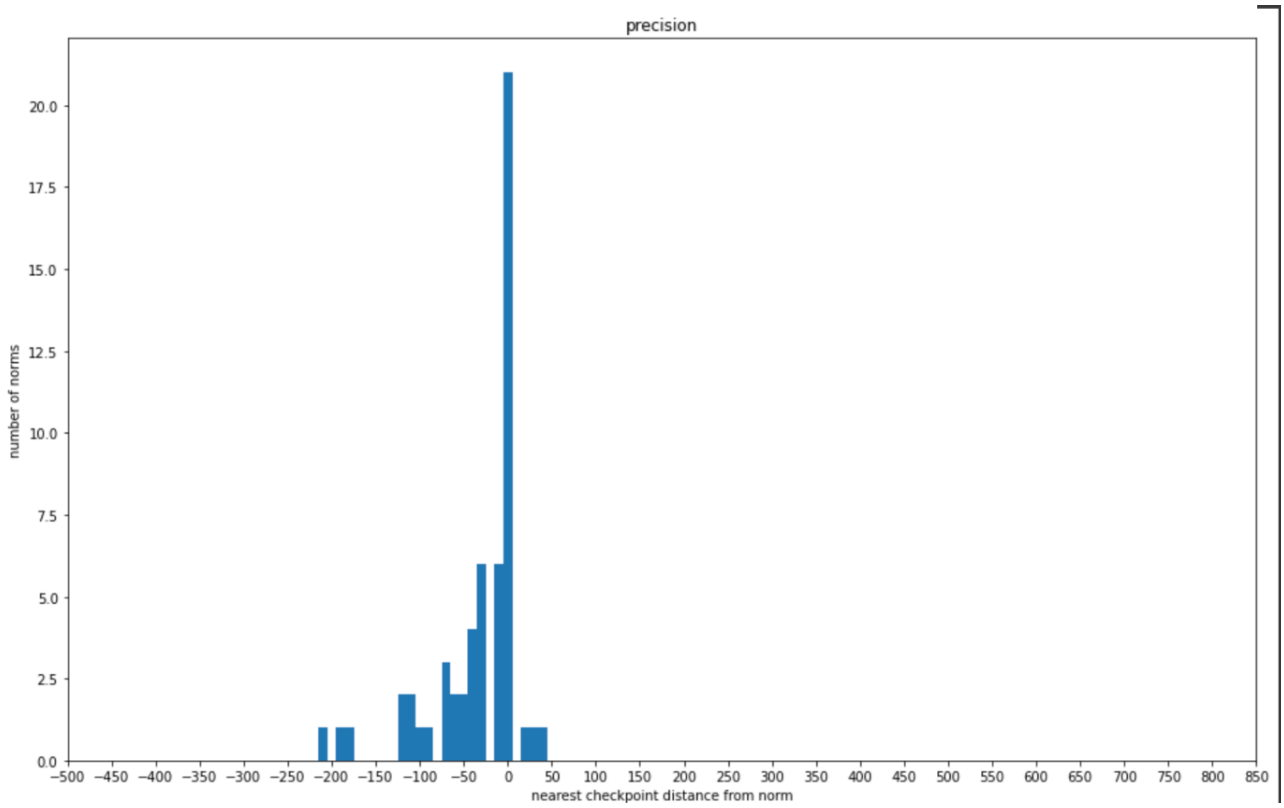

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

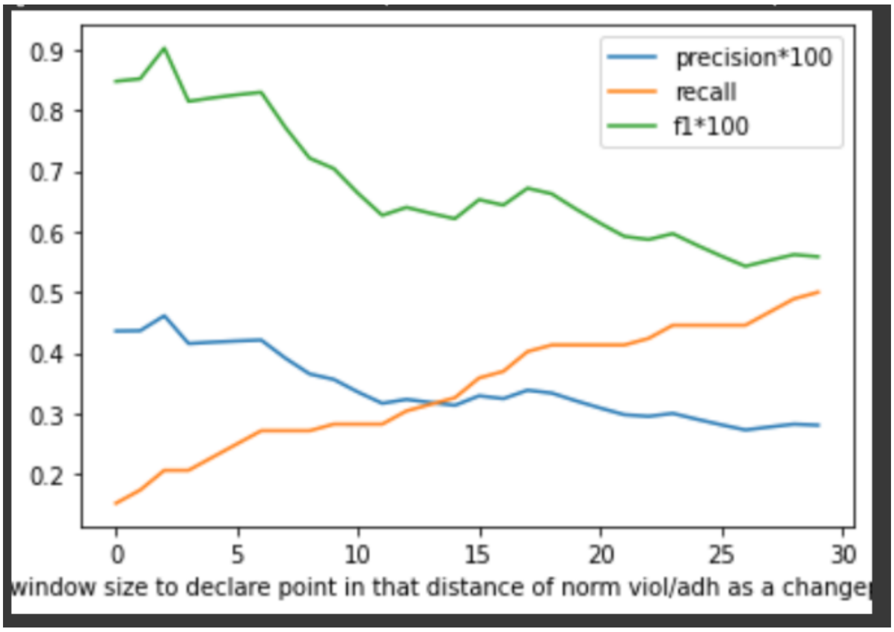

precision
number of norms
nearest checkpoint distance from norm


precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

- Violate

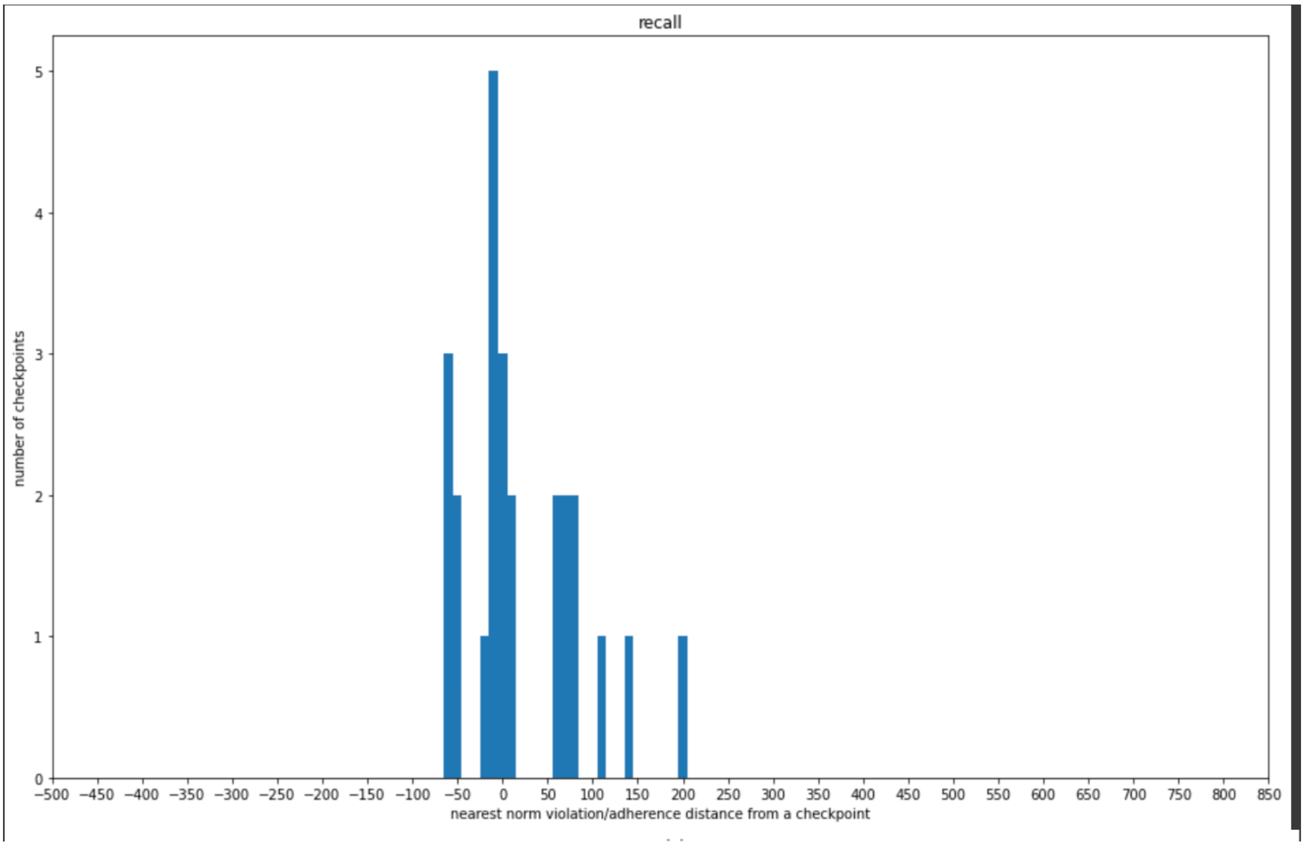

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

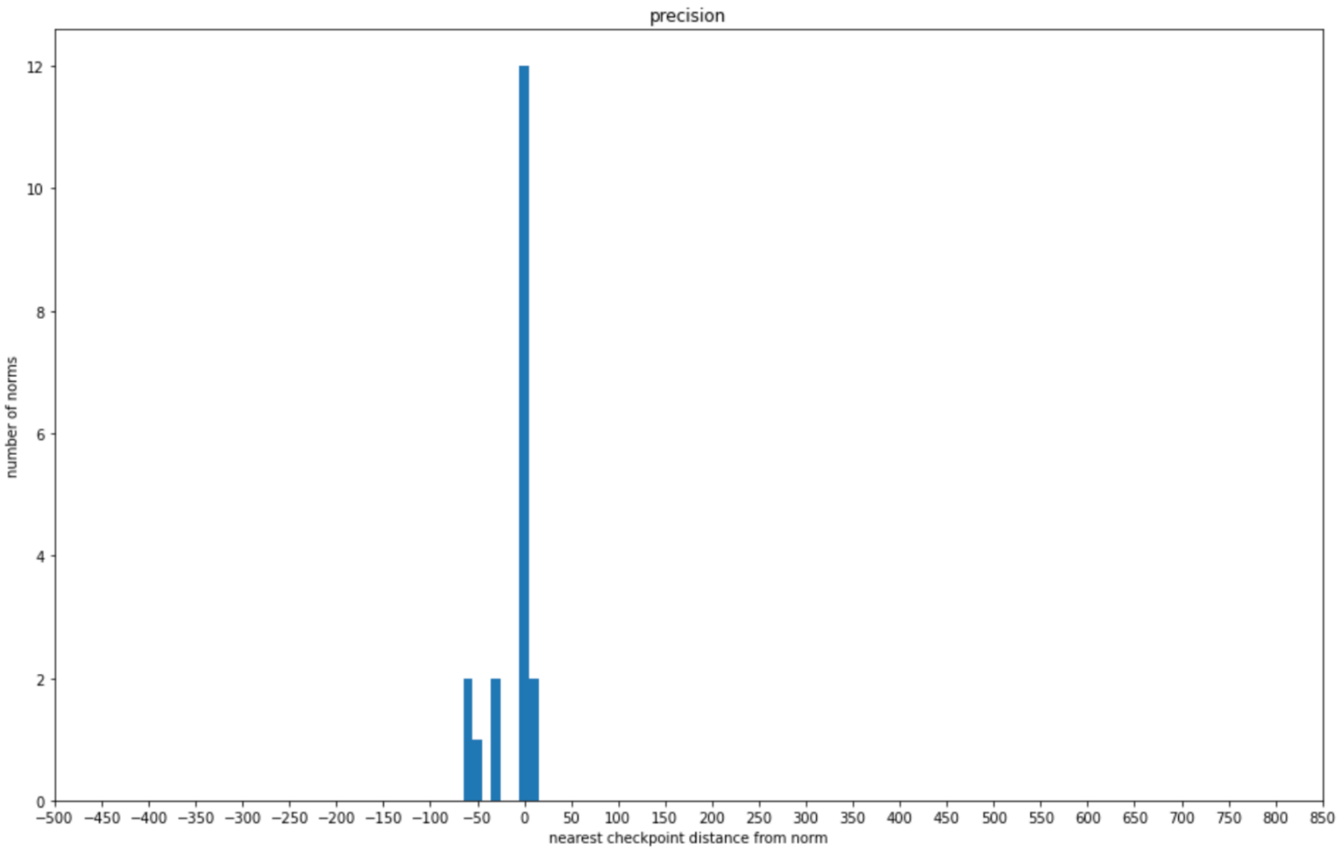


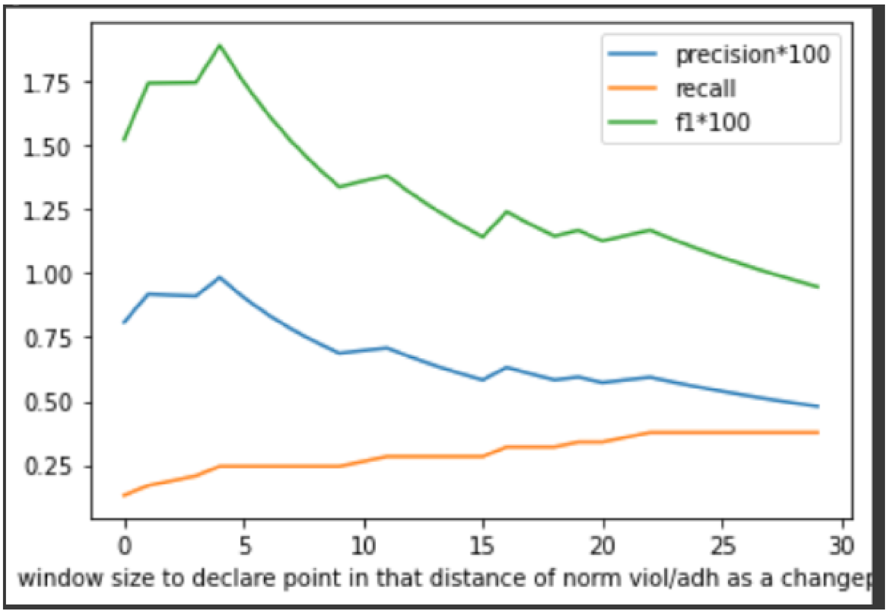


6. Precision, Recall and PR- F1 score curves for different norm types.

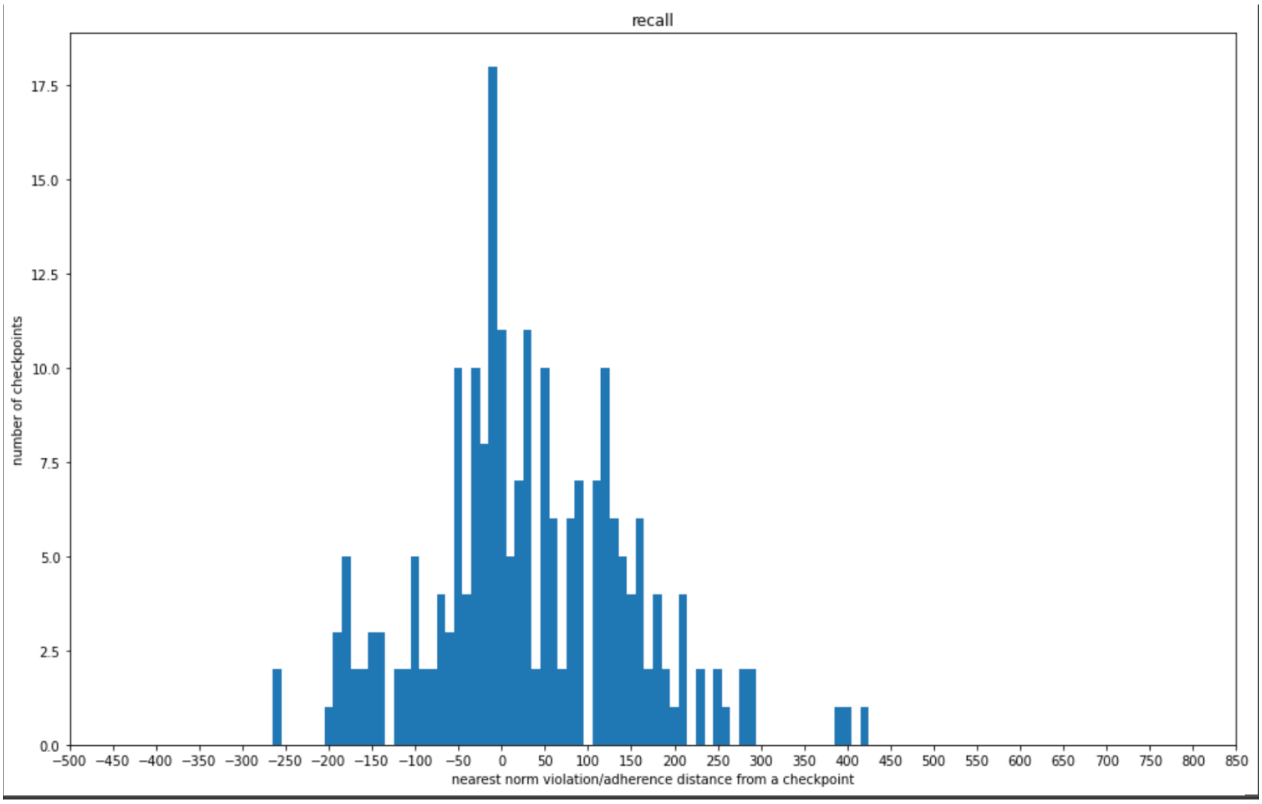
- 101 : apology

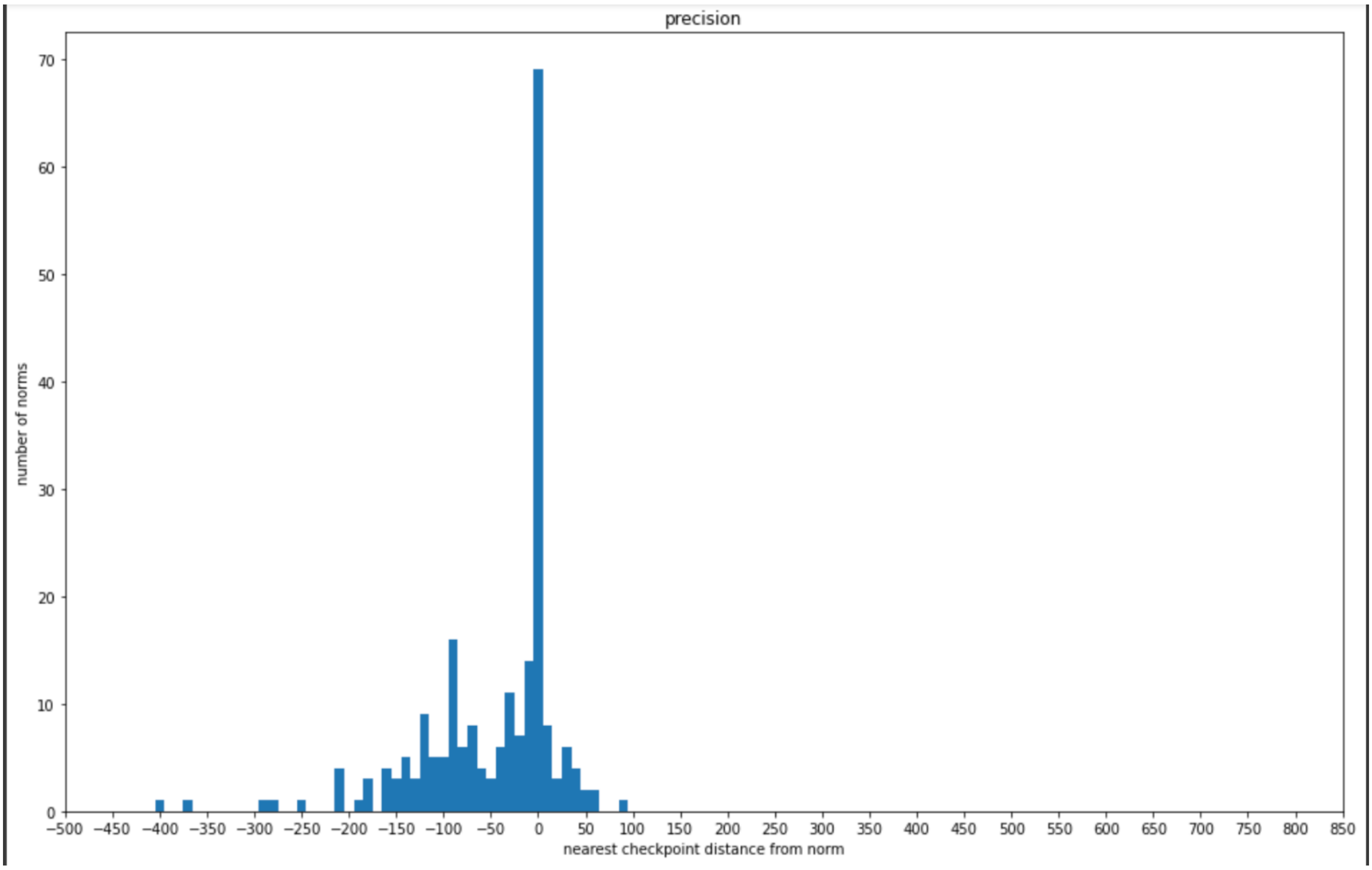

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

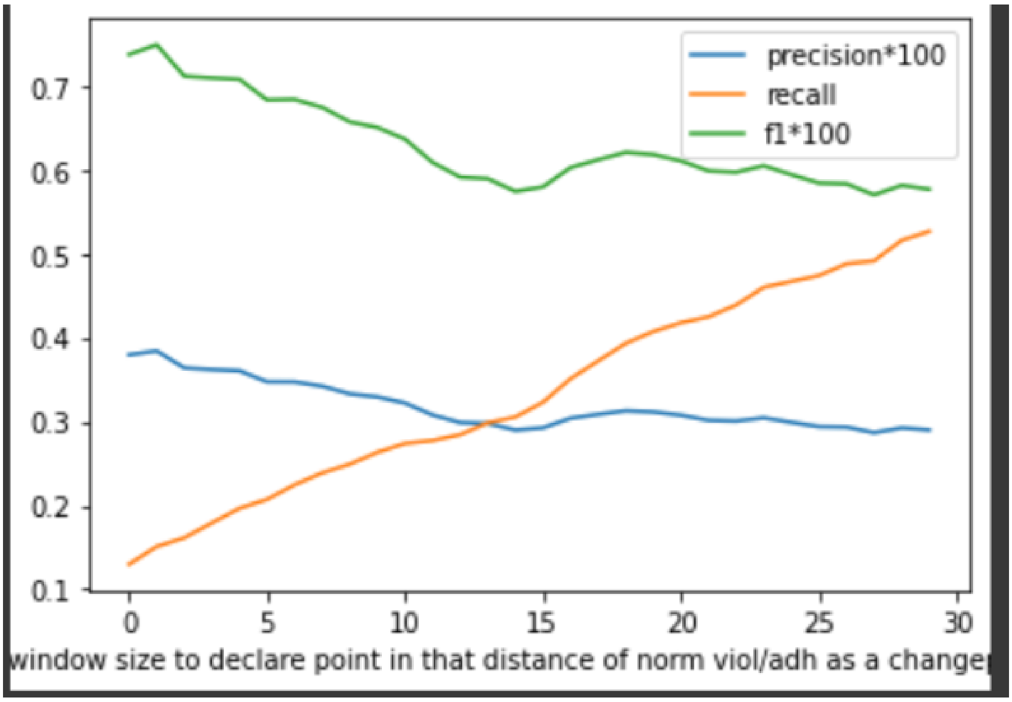

precision
number of norms
nearest checkpoint distance from norm



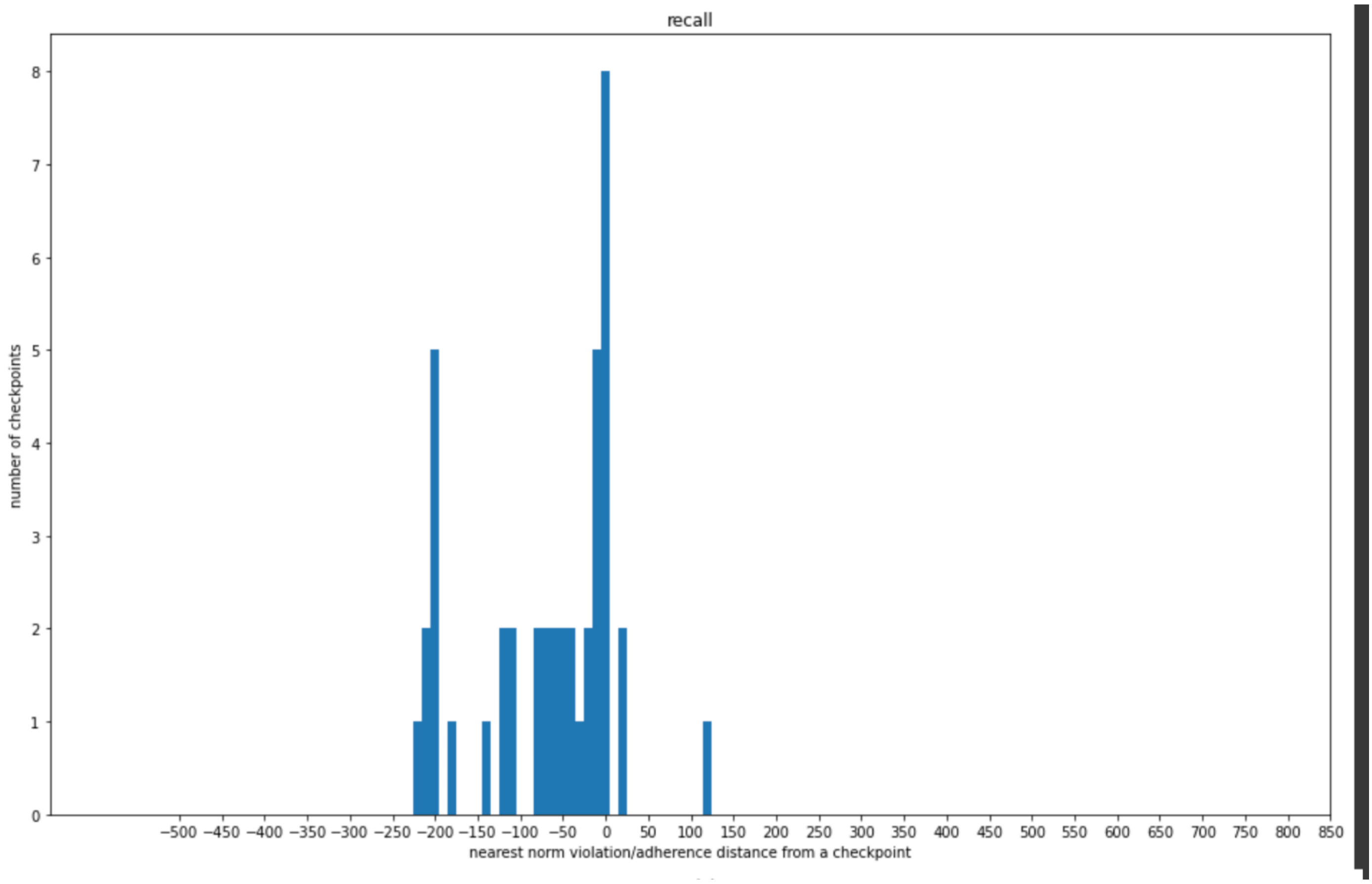
- 102 : criticism

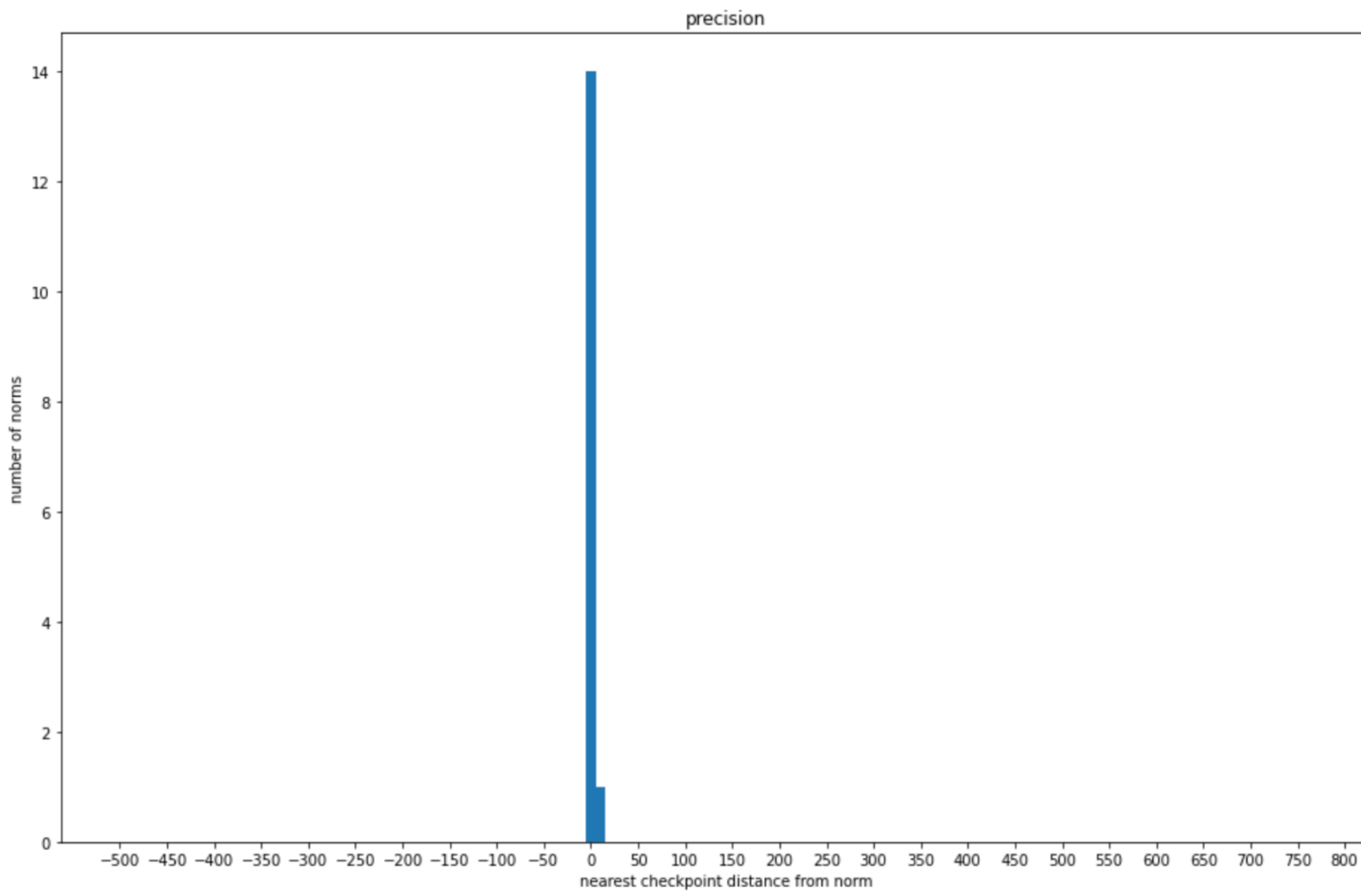


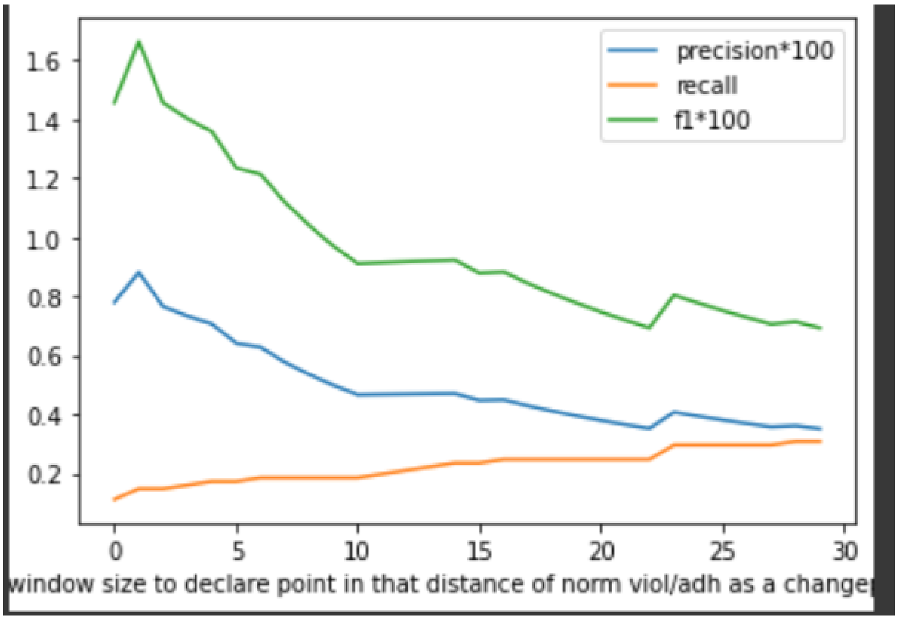




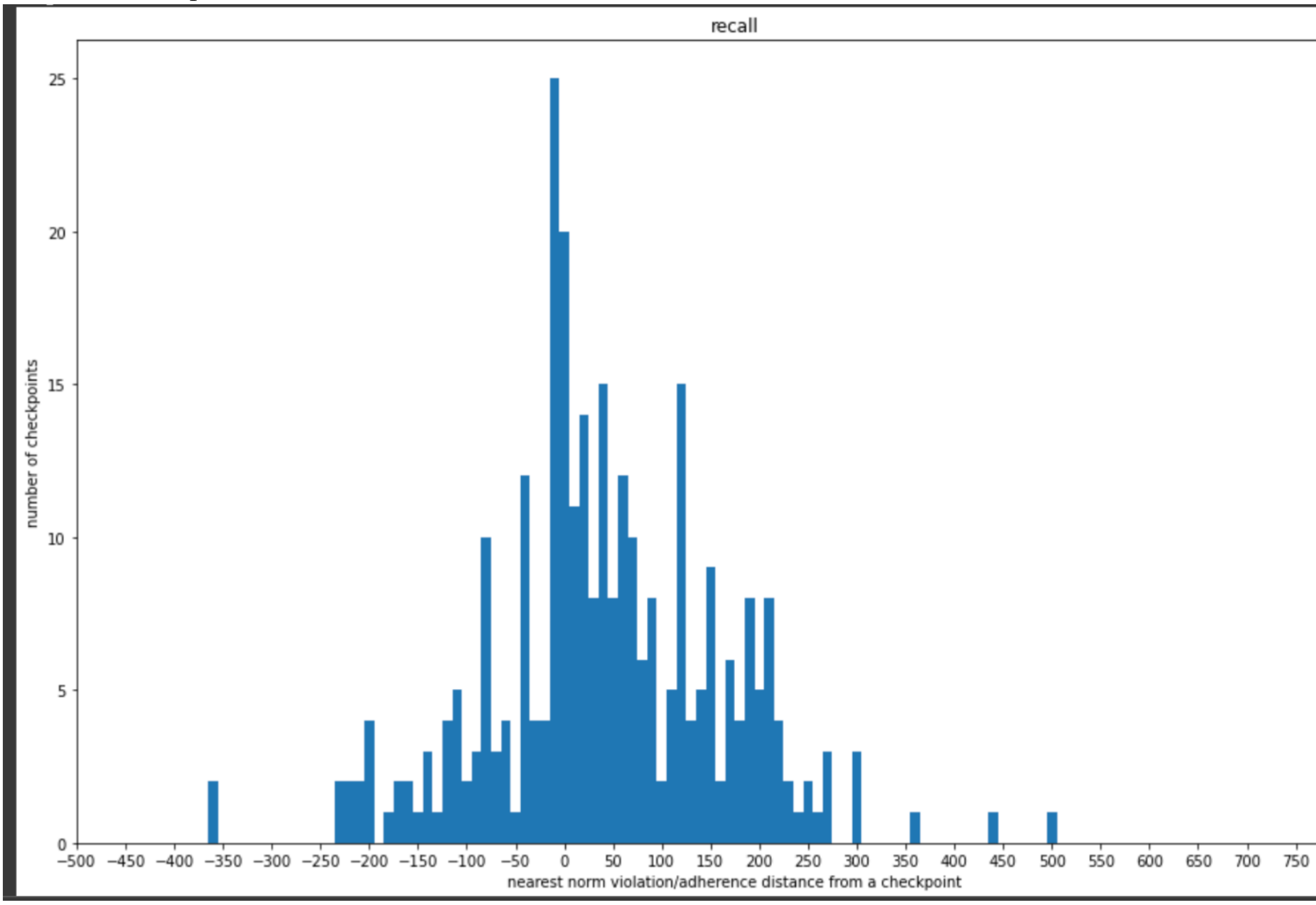
- 103 : greeting

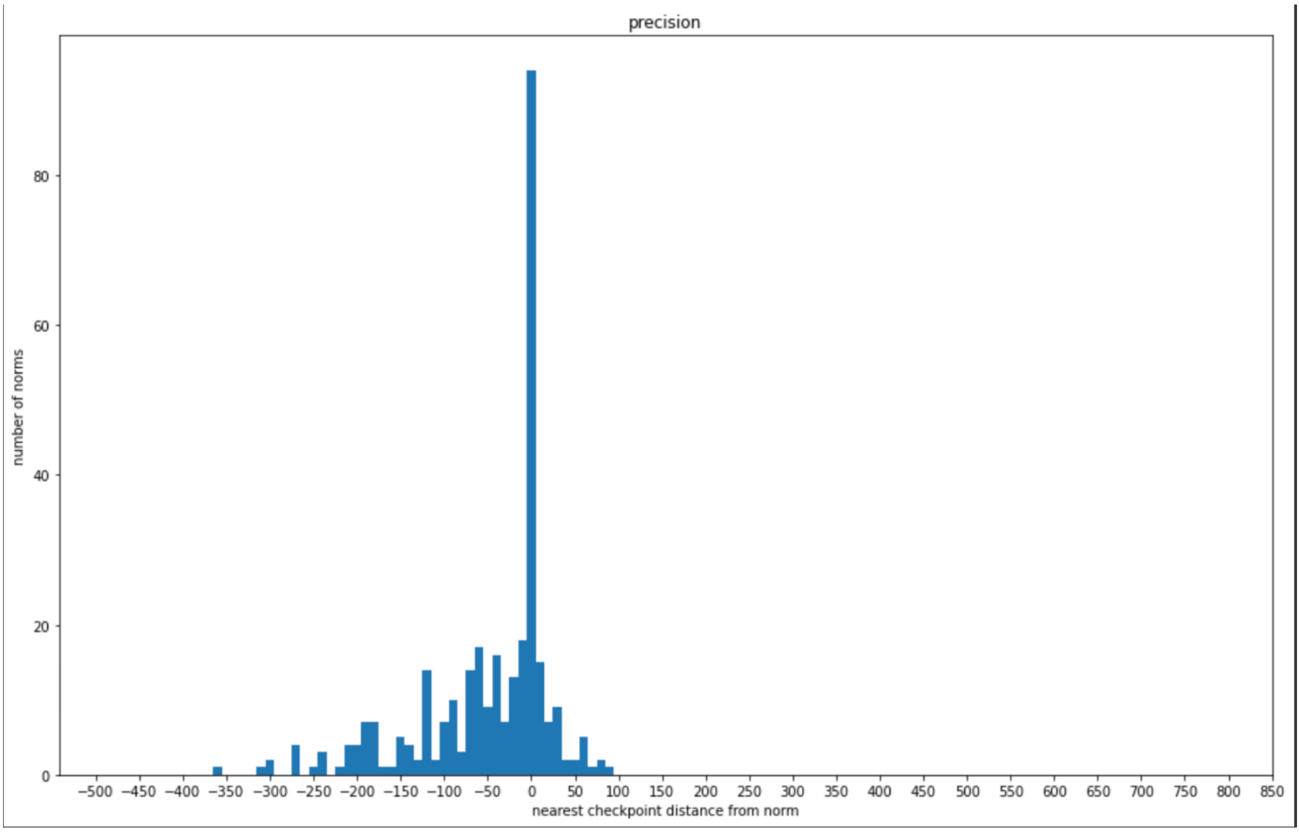

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint
0
1
2
3
4
5
6
7
8
−500 −450 −400 −350 −300 −250 −200 −150 −100 −50 0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850

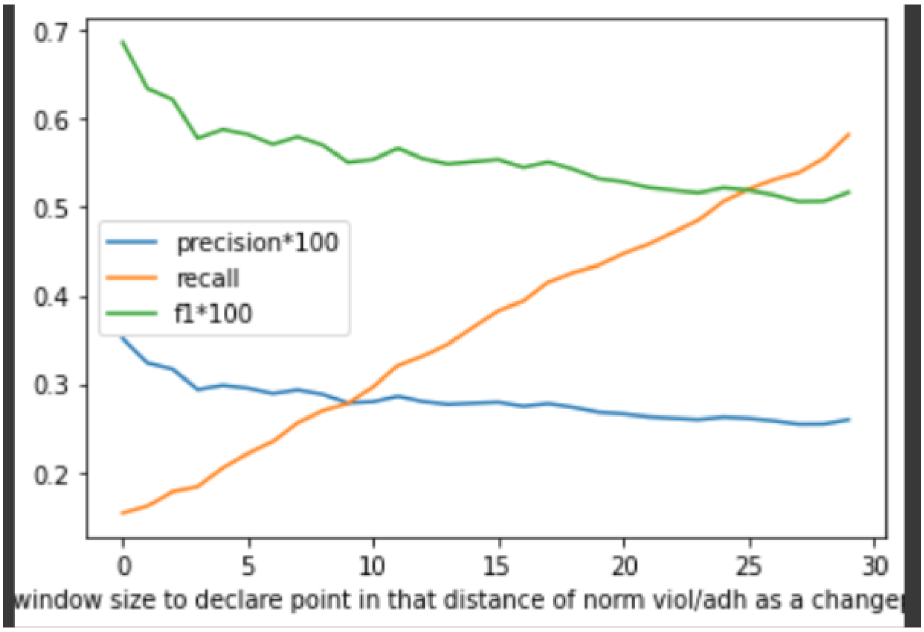

precision
number of norms
nearest checkpoint distance from norm
0
2
4
6
8
10
12
14
−500 −450 −400 −350 −300 −250 −200 −150 −100 −50 0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800


precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

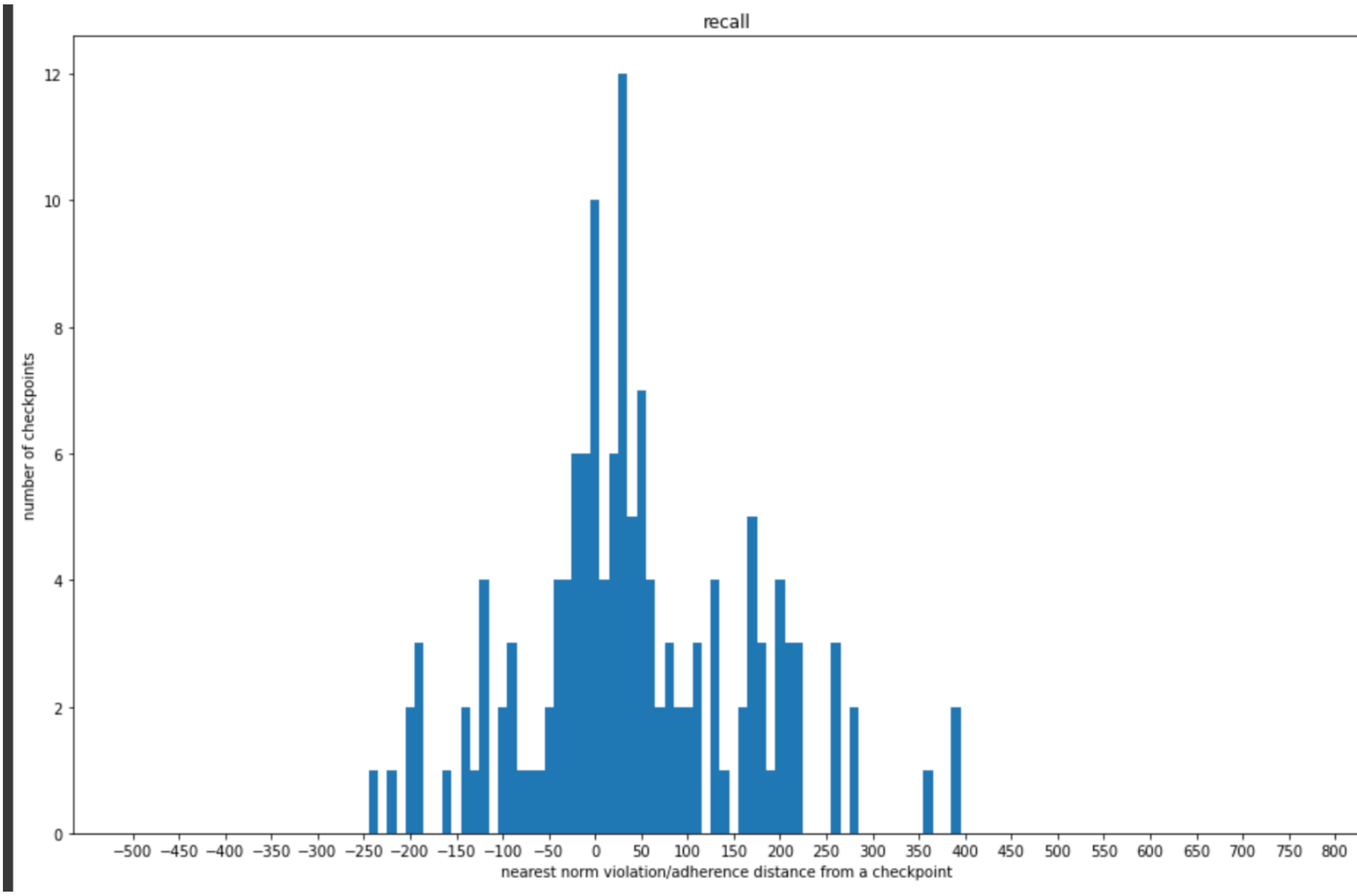
- 104 : request

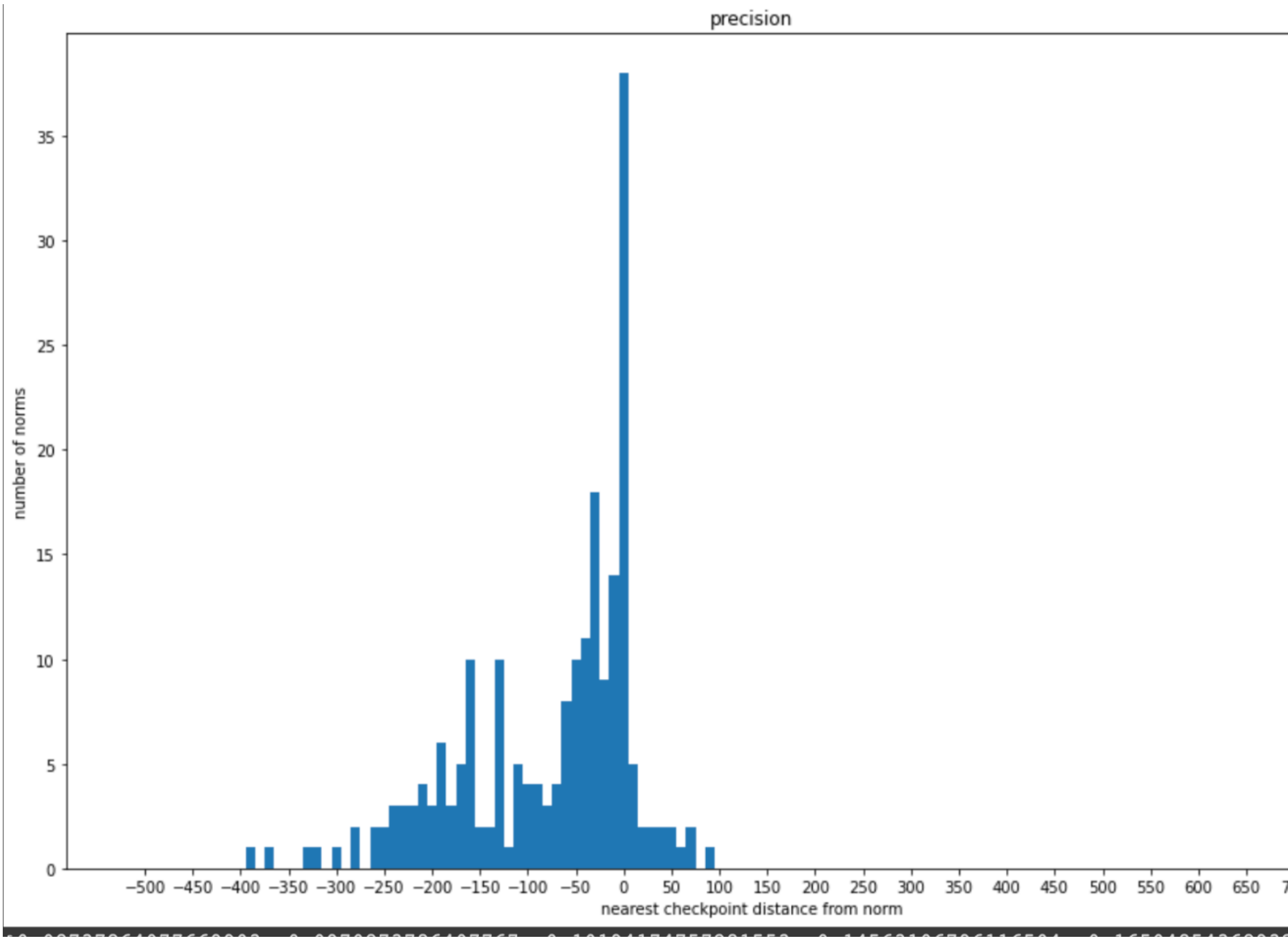

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

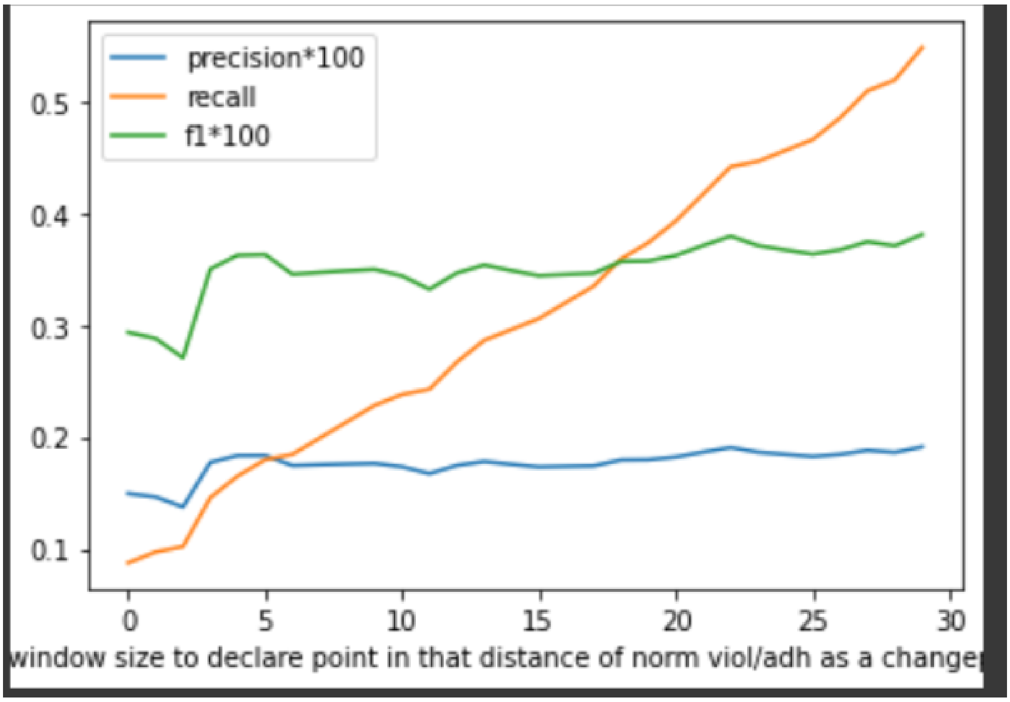




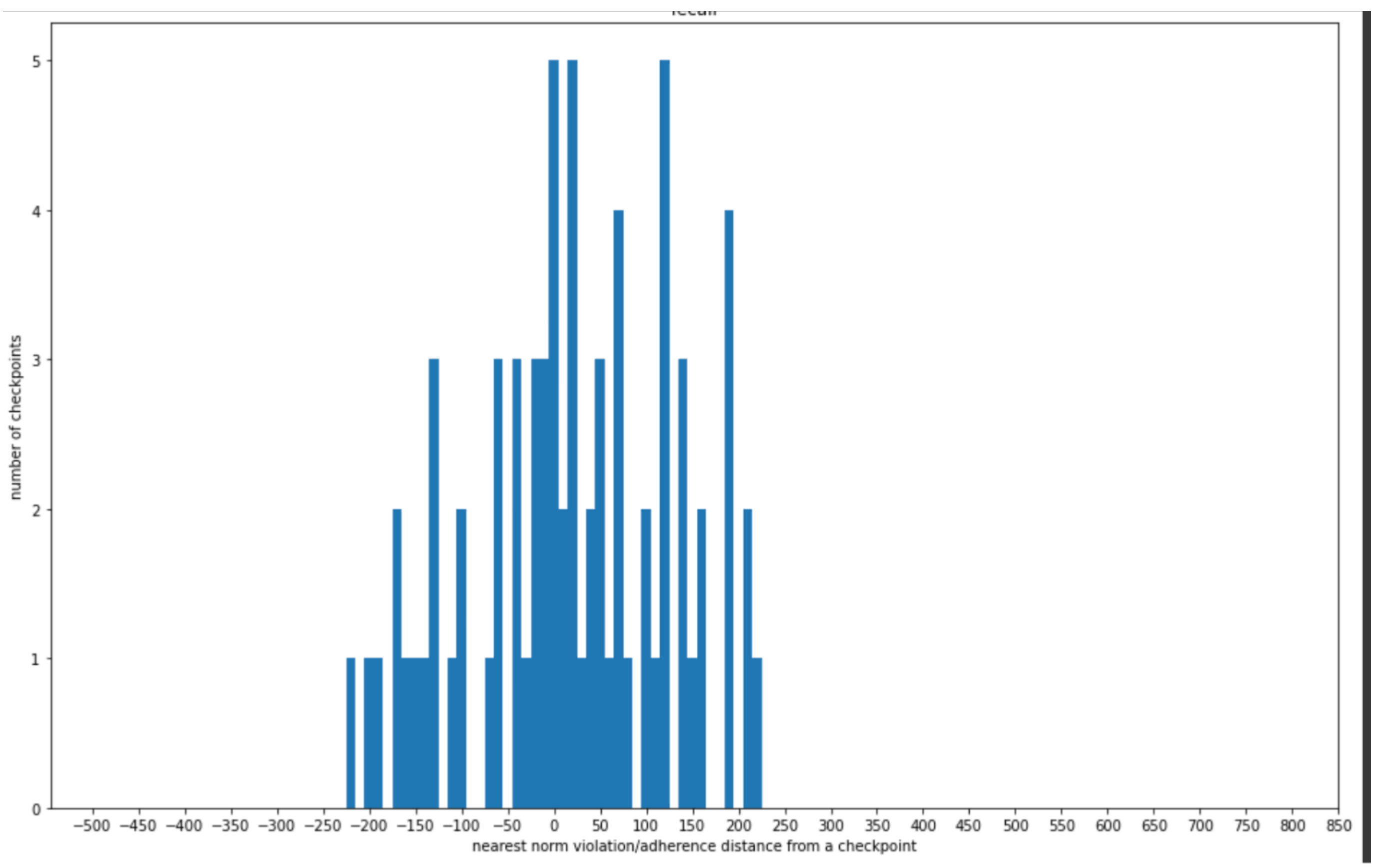
- 105 : persuasion

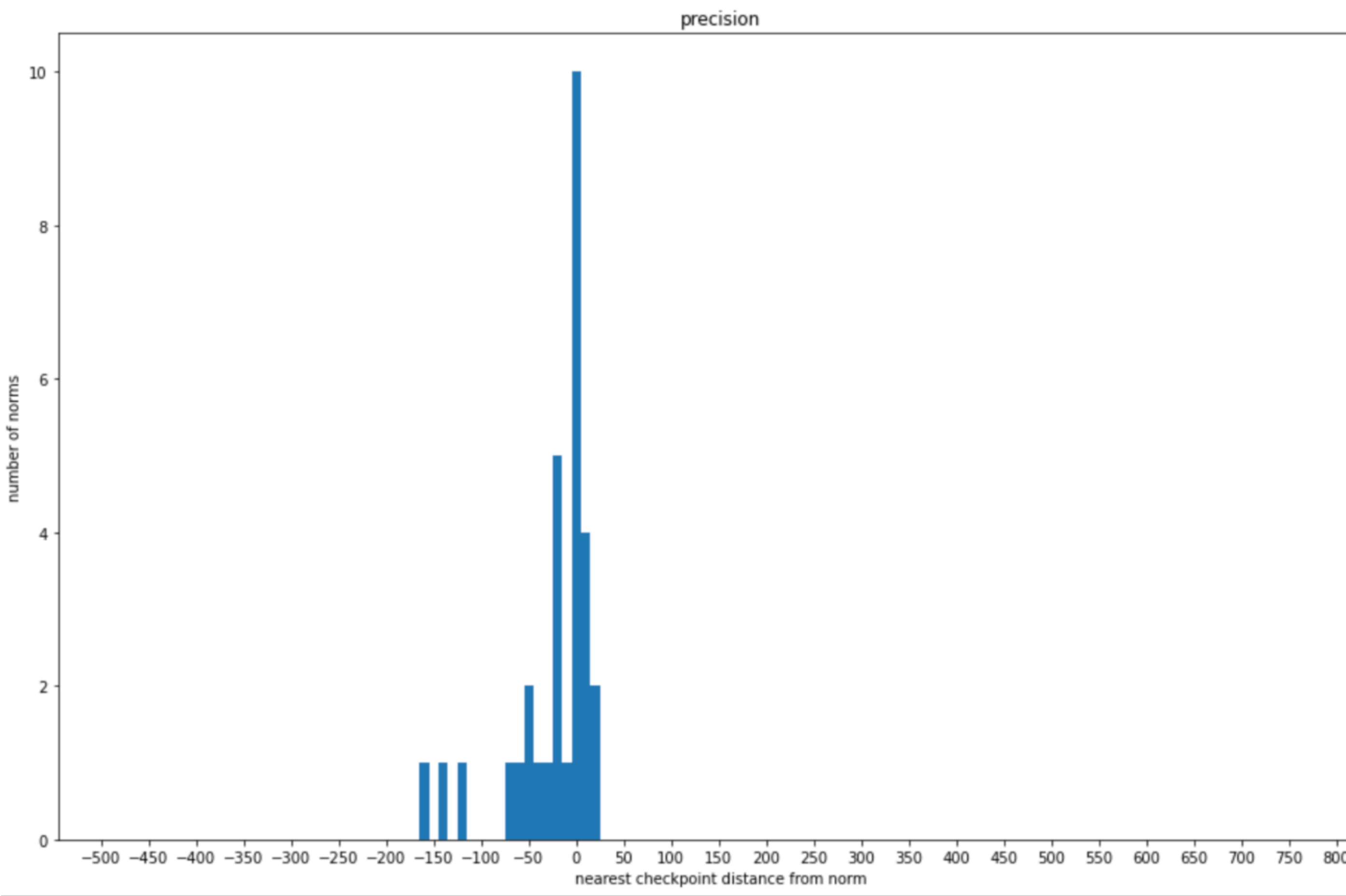

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

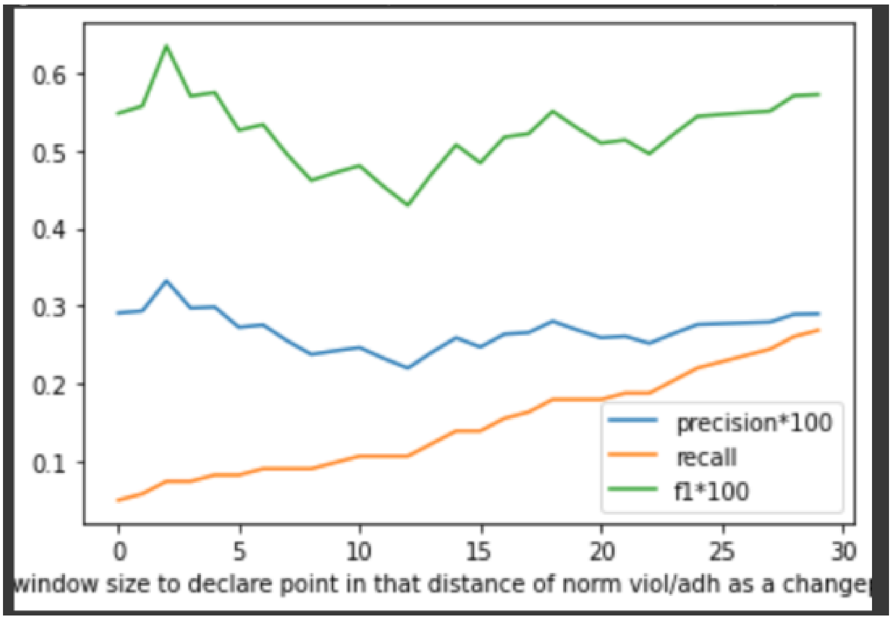

precision
number of norms
nearest checkpoint distance from norm


precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

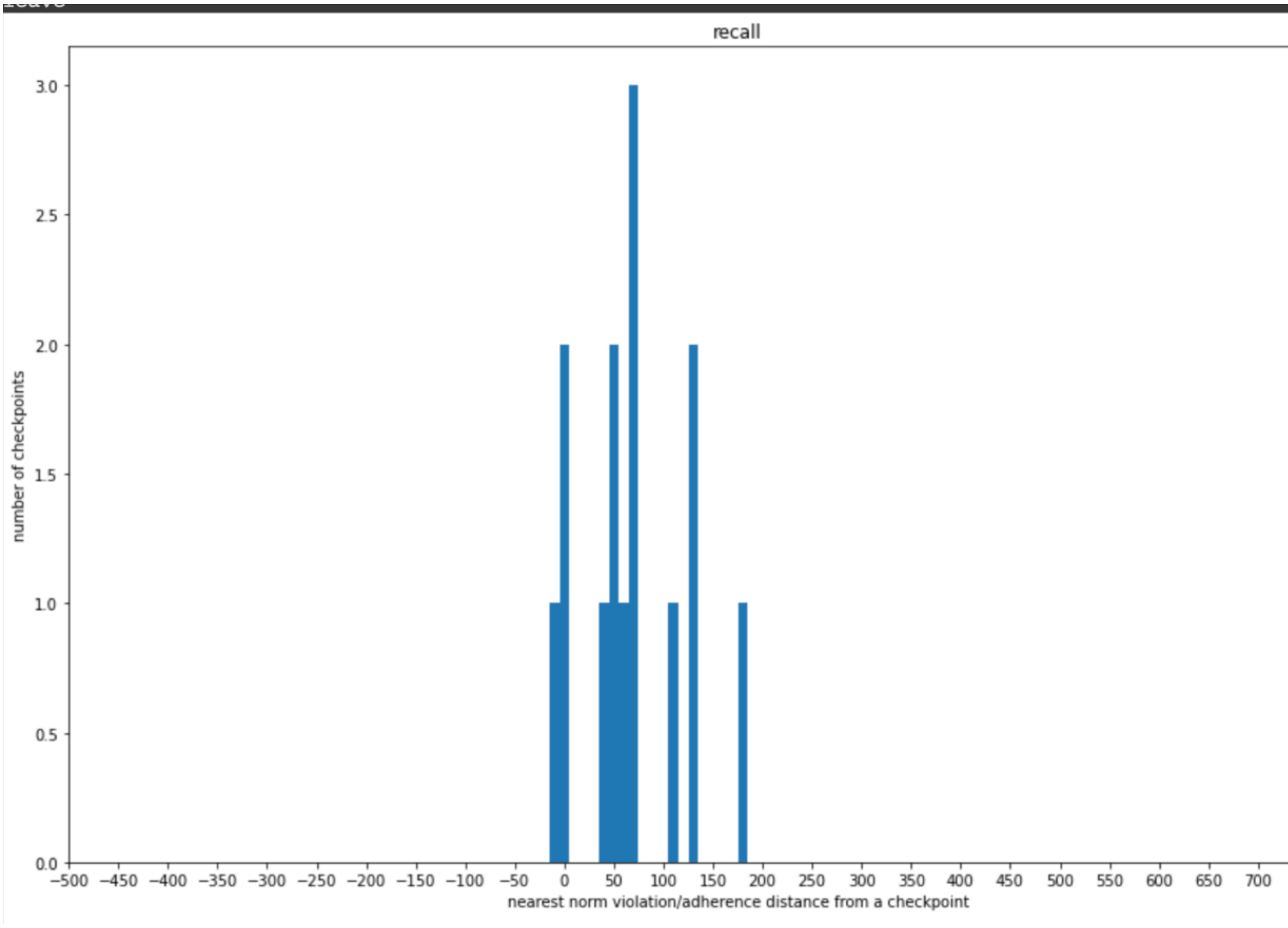
- 106 : thanks

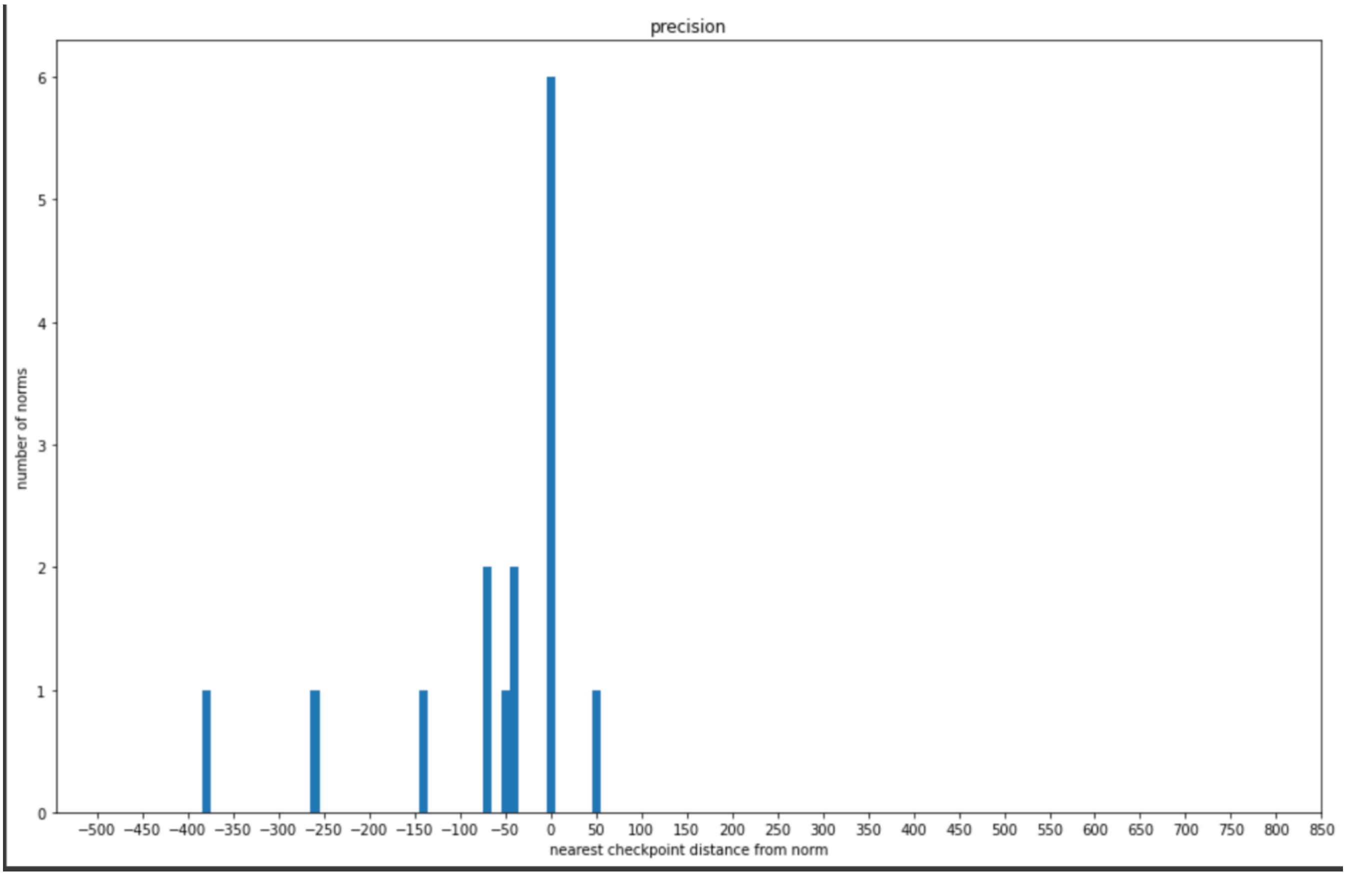

number of checkpoints
0
1
2
3
4
5
−500 −450 −400 −350 −300 −250 −200 −150 −100 −50 0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750 800 850
nearest norm violation/adherence distance from a checkpoint

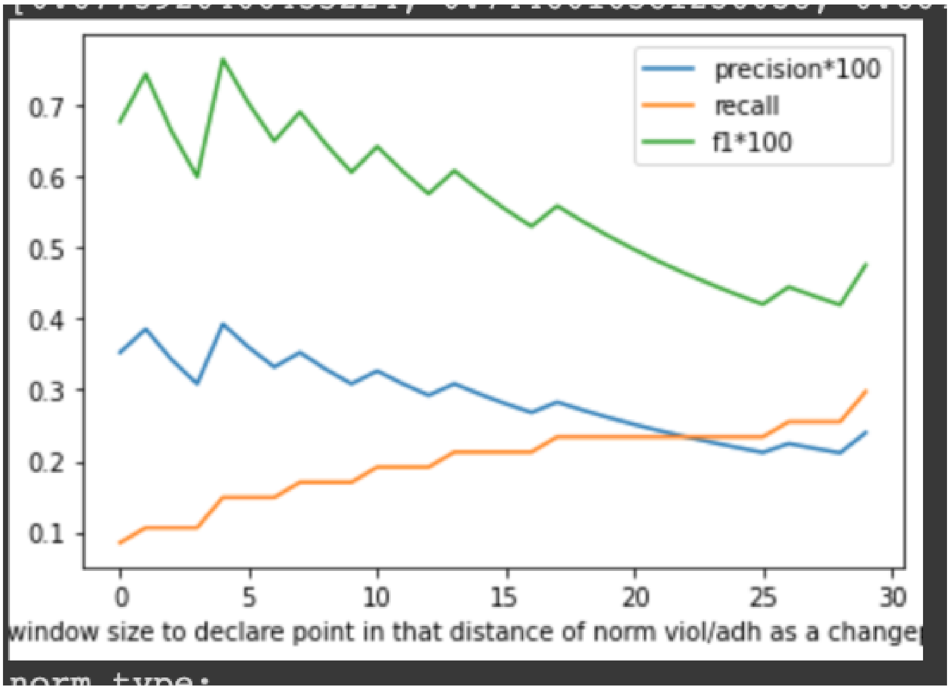

precision
number of norms
0
2
4
6
8
10
−500 −450 −400 −350 −300 −250 −200 −150 −100 −50 0 50 100 150 200 250 300 350 400 450 500 550 600 650 700 750
nearest checkpoint distance from norm


precision*100
recall
f1*100
window size to declare point in that distance of norm viol/adh as a change

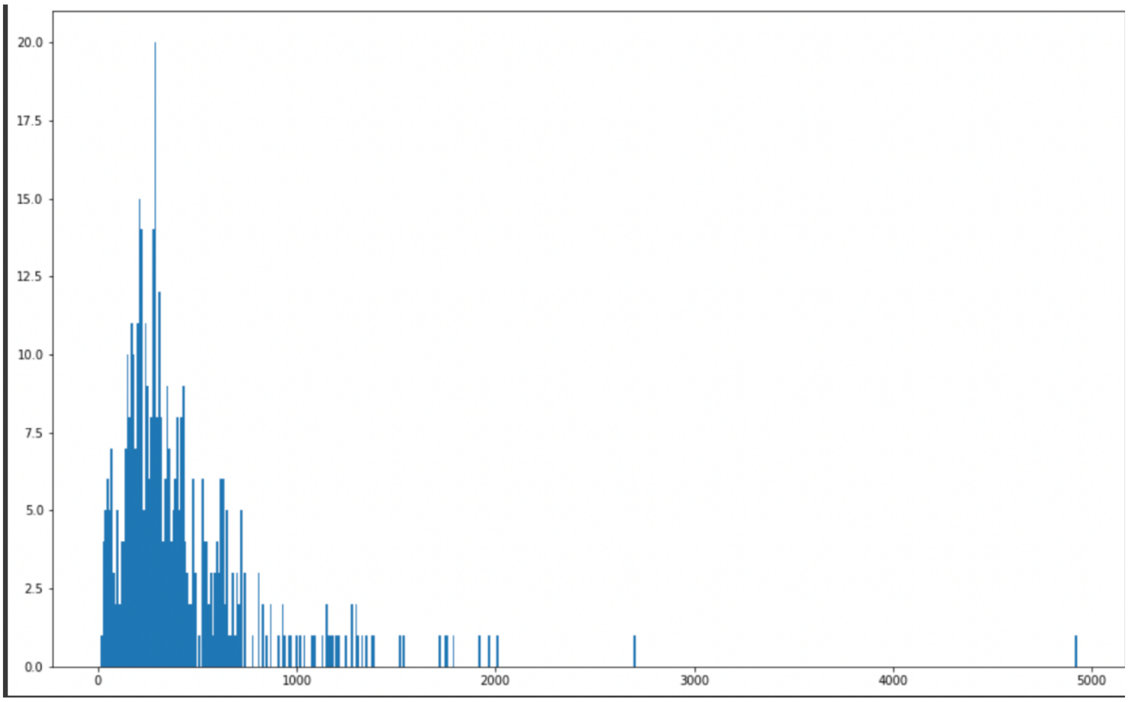
- 107 : leave


recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

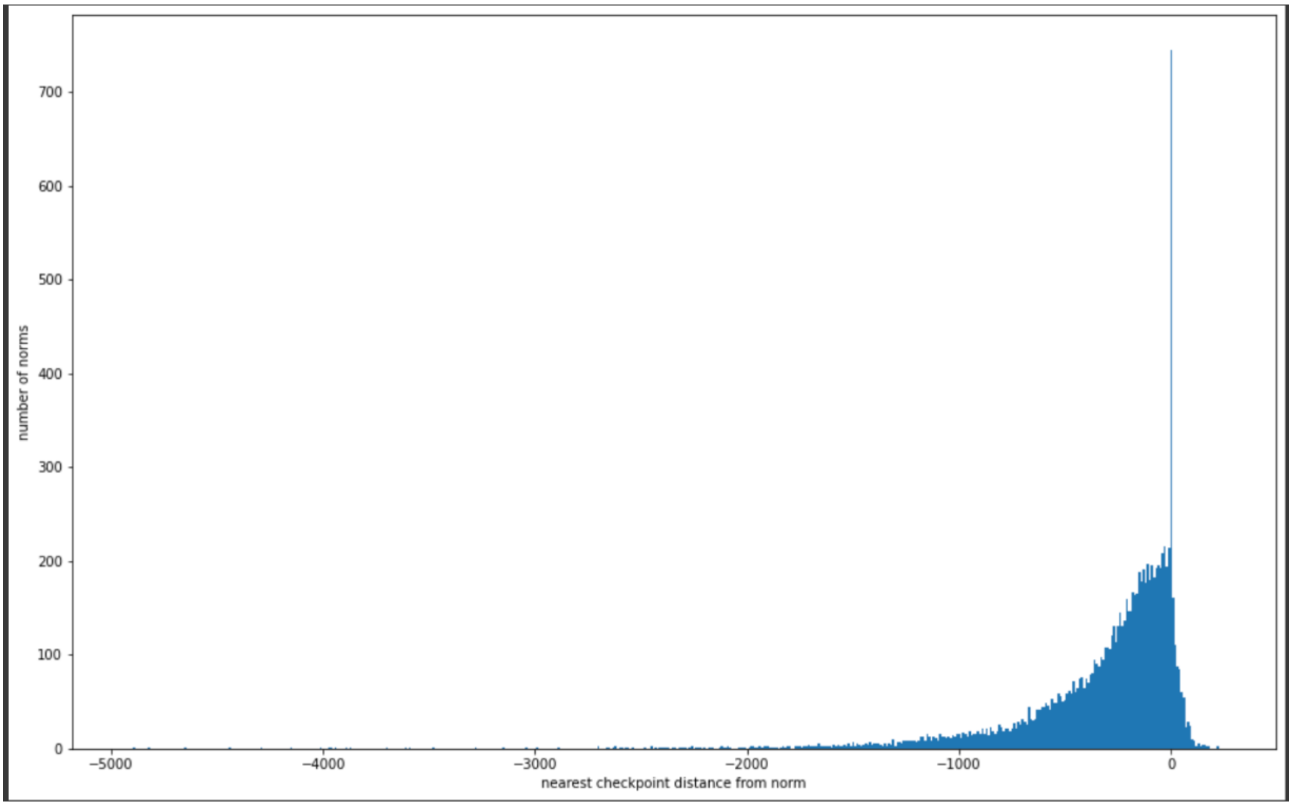




7. Recall curve for the entire UIUC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.

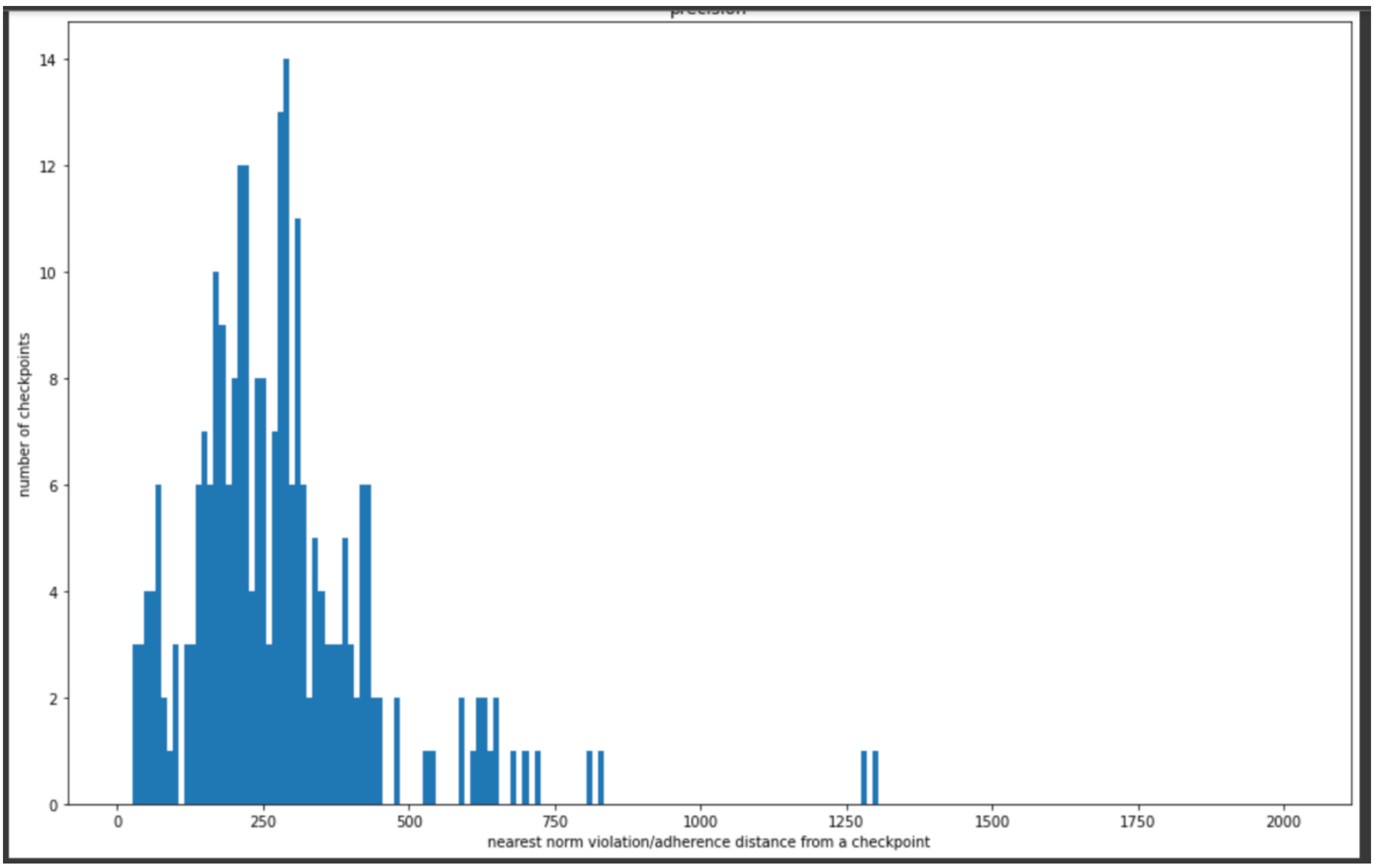


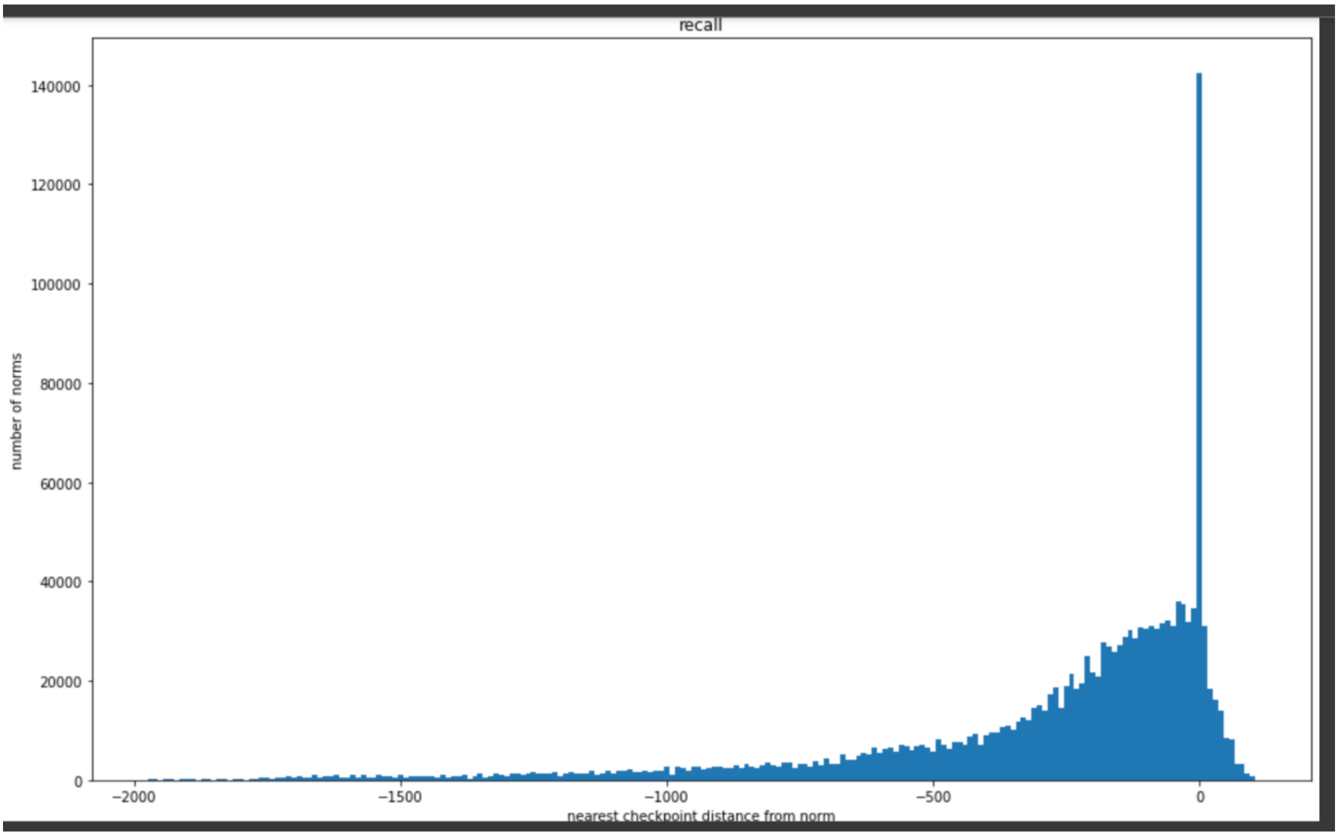
8. Precision curve for the entire UIUC data.Plotted the distance of the nearest norm violation/adherence from a checkpoint against the number of checkpoints.



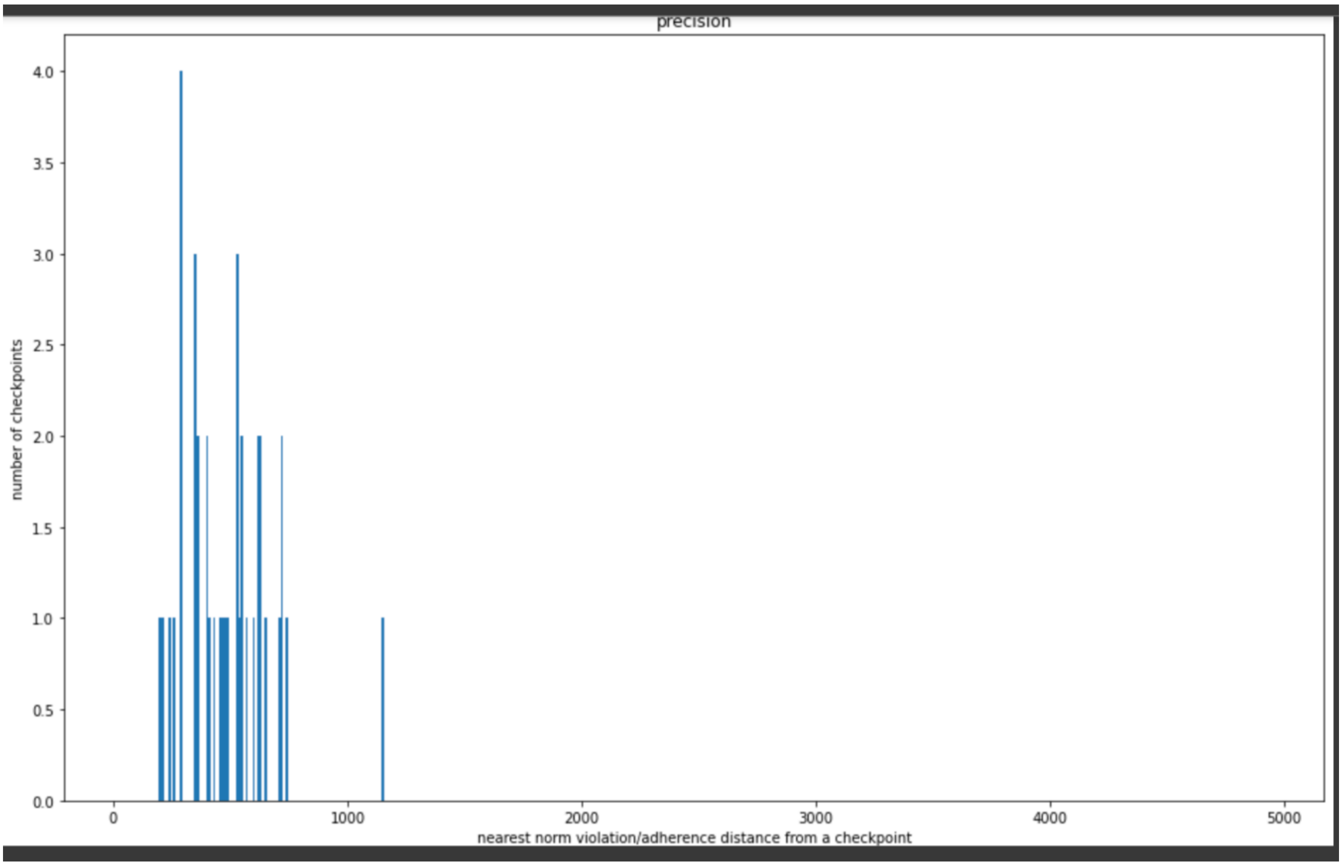
9. Recall and Precision curves for different file types.

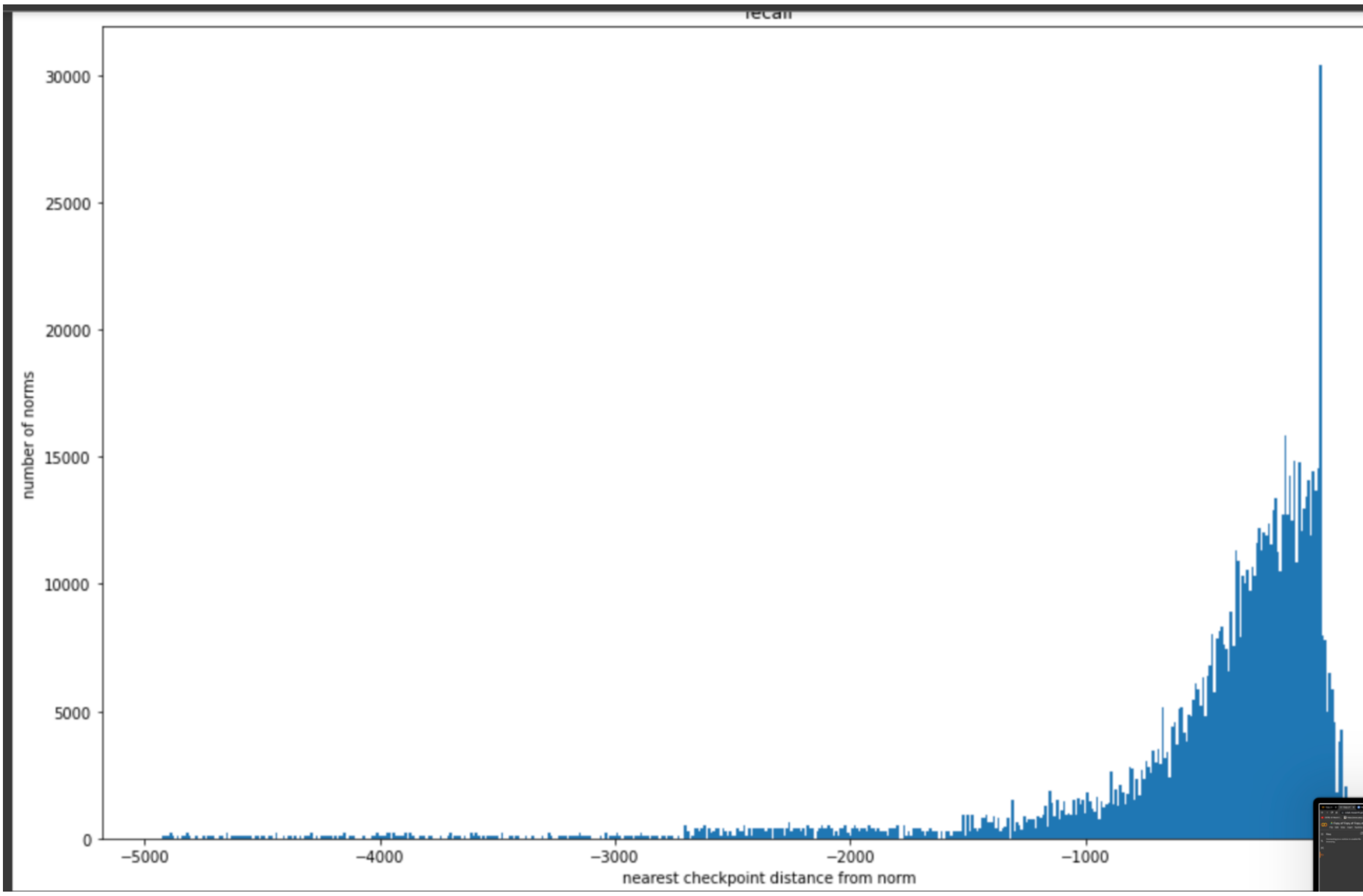
- .mp4.ldcc


precision
number of checkpoints
nearest norm violation/adherence distance from a checkpoint


recall
number of norms
nearest checkpoint distance from norm

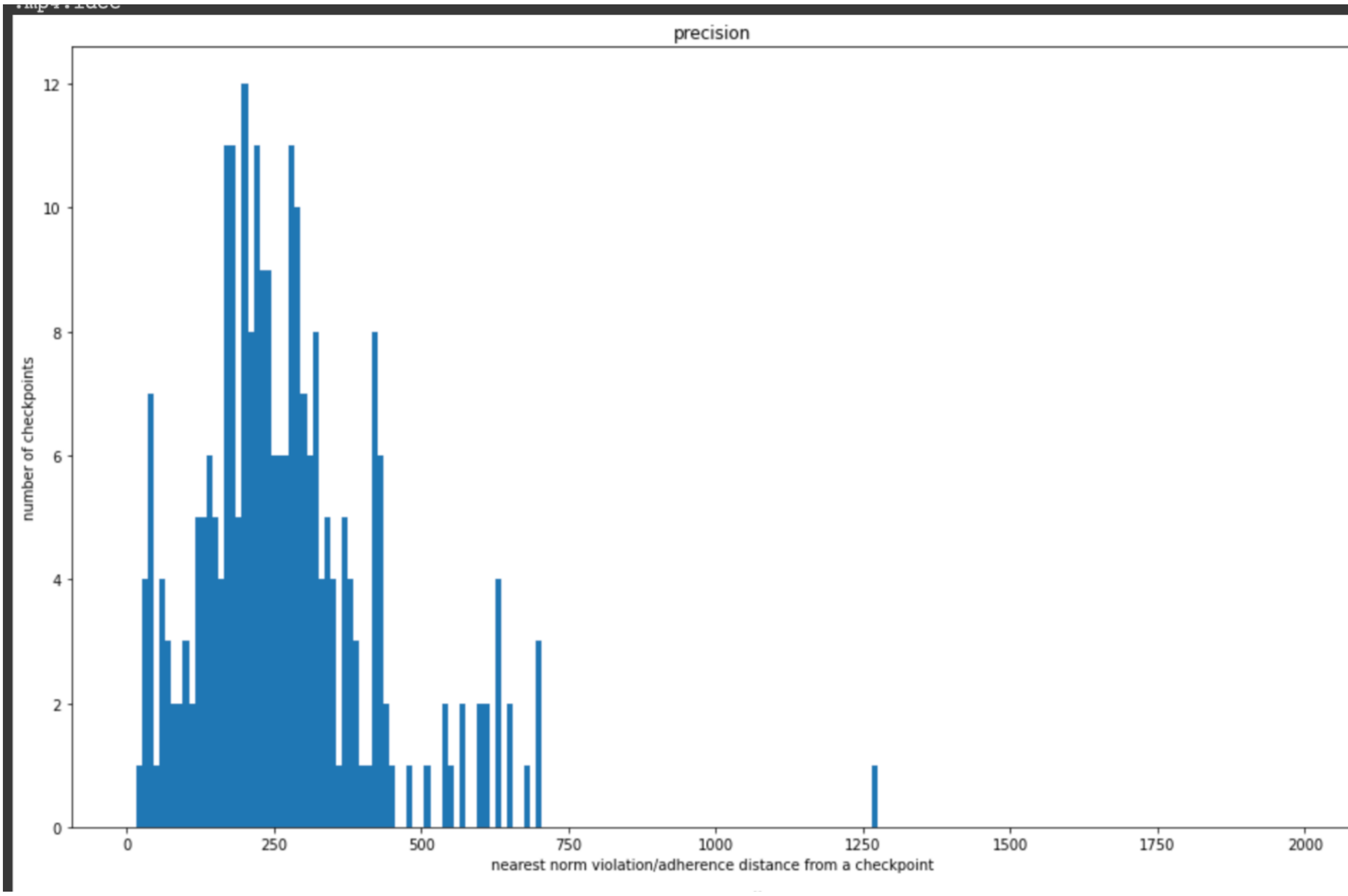
- .psm.xml



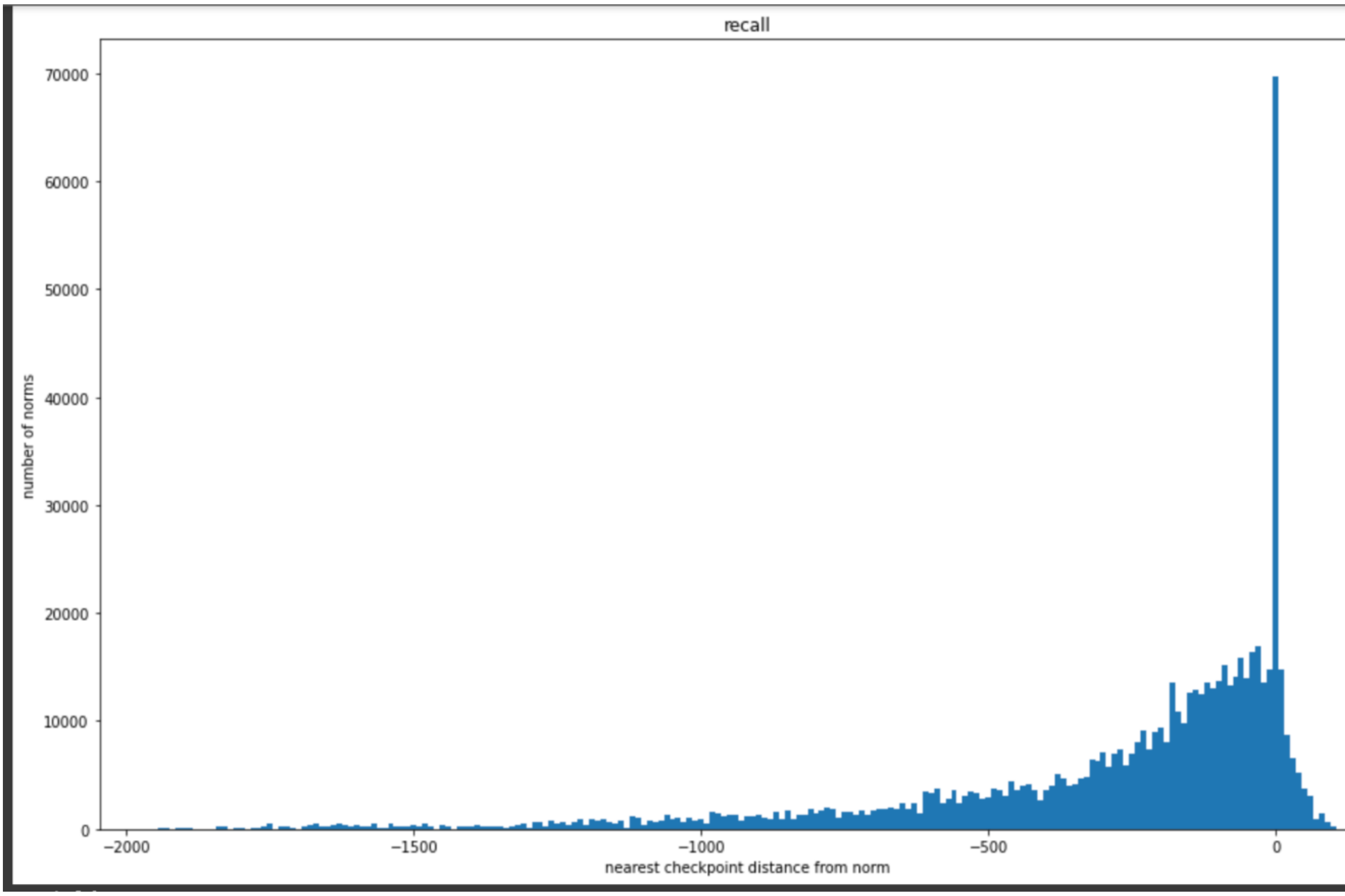


10. Recall and Precision curves for norm violations and adherences separately.

- Adhere

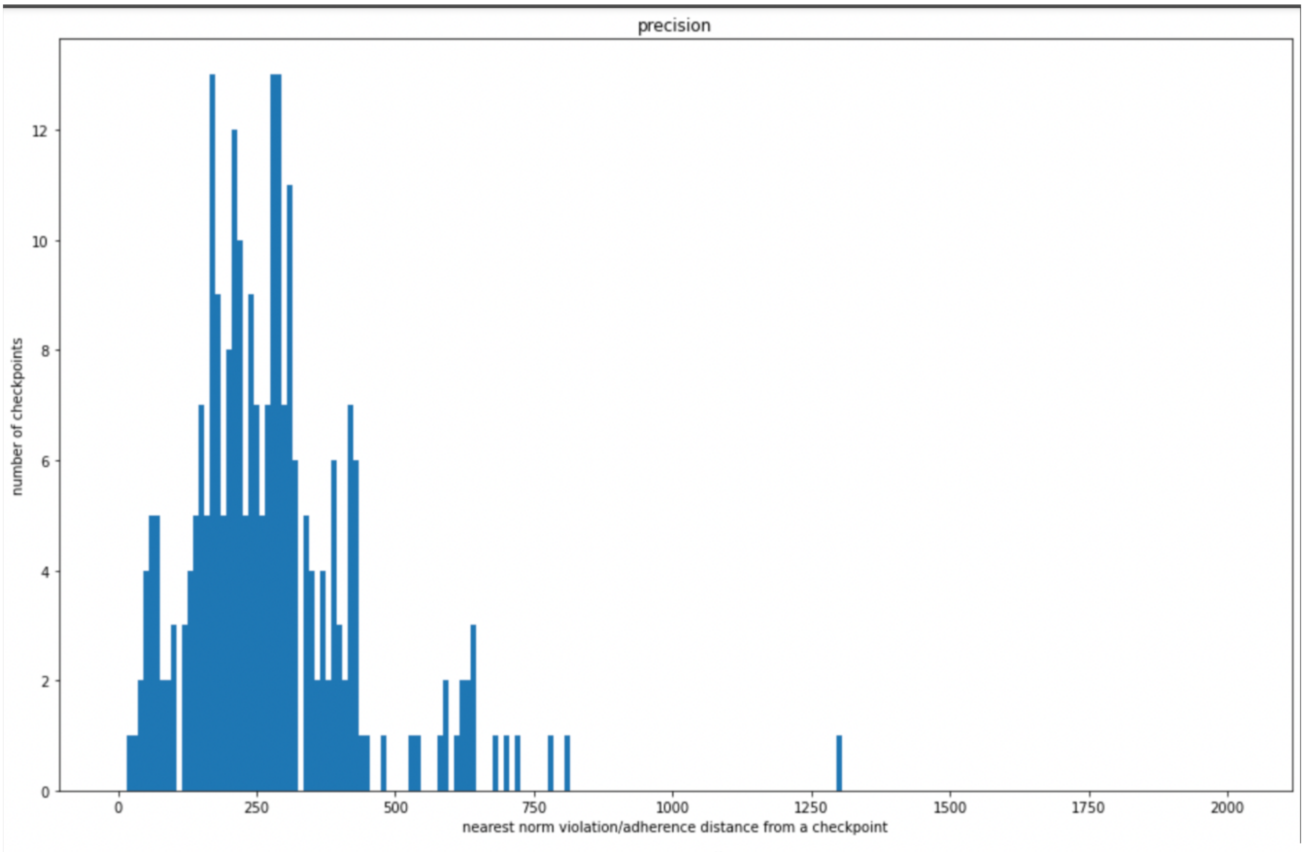


precision
number of checkpoints
nearest norm violation/adherence distance from a checkpoint
0
2
4
6
8
10
12
0
250
500
750
1000
1250
1500
1750
2000

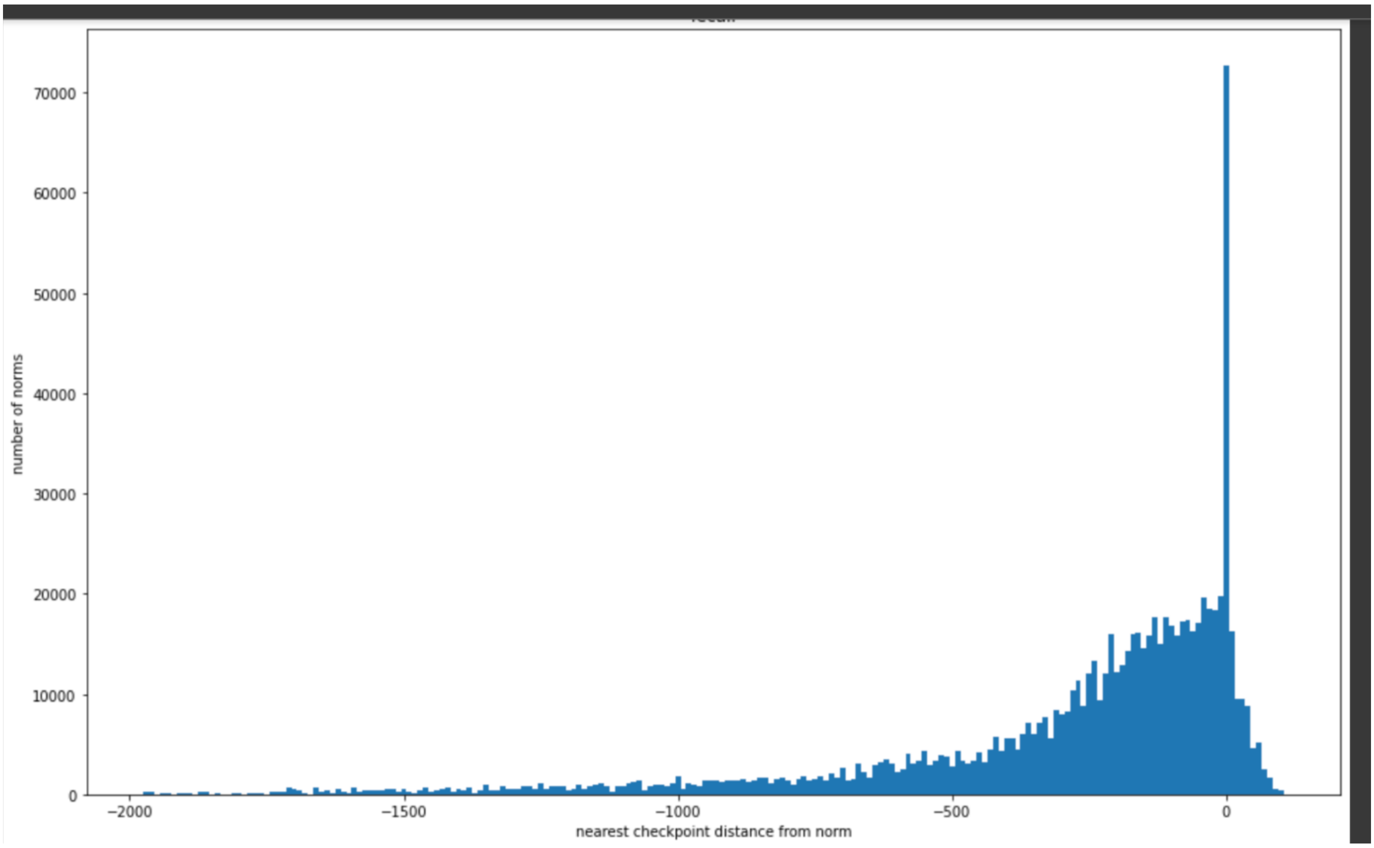


recall
number of norms
nearest checkpoint distance from norm

- Violate

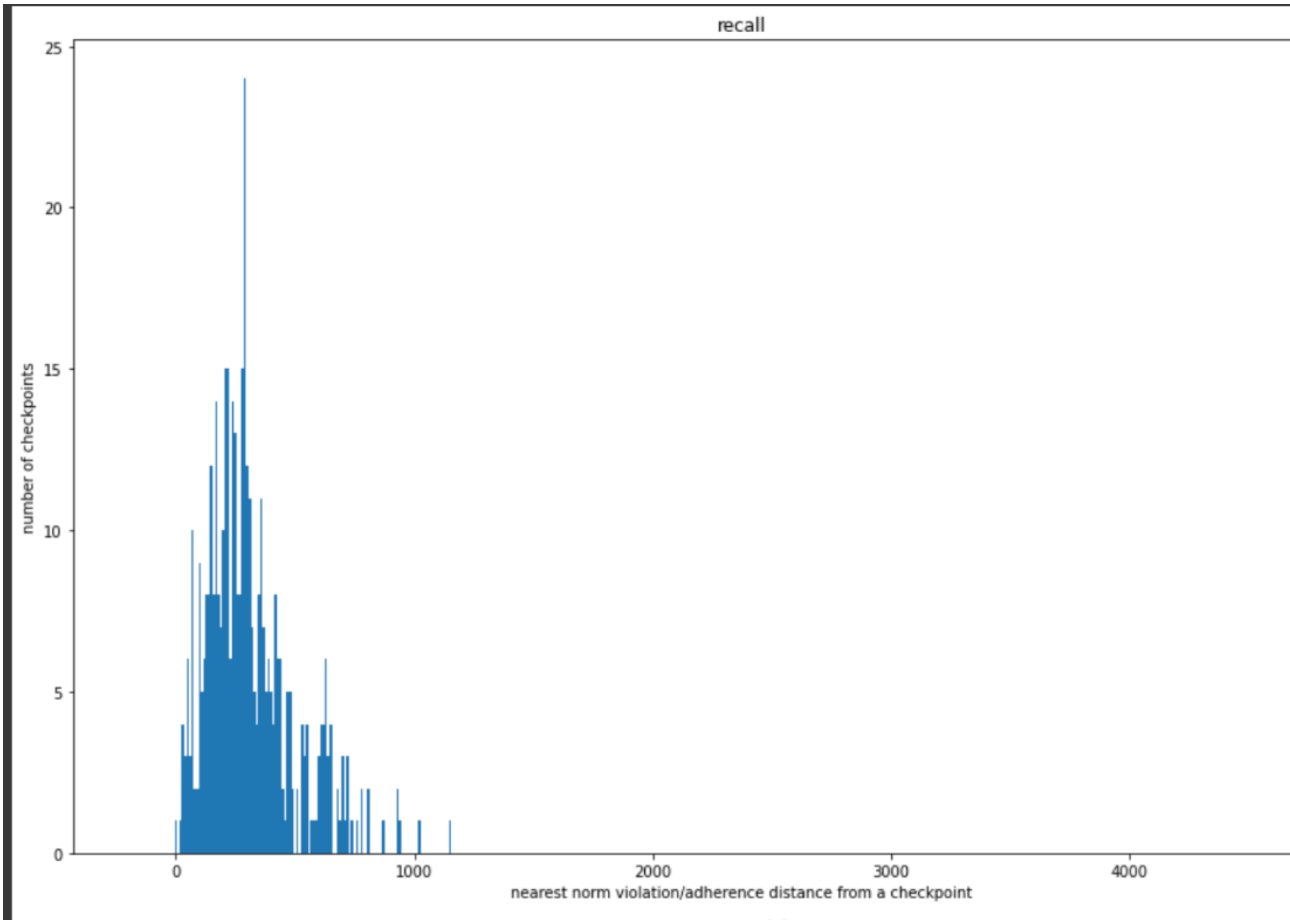


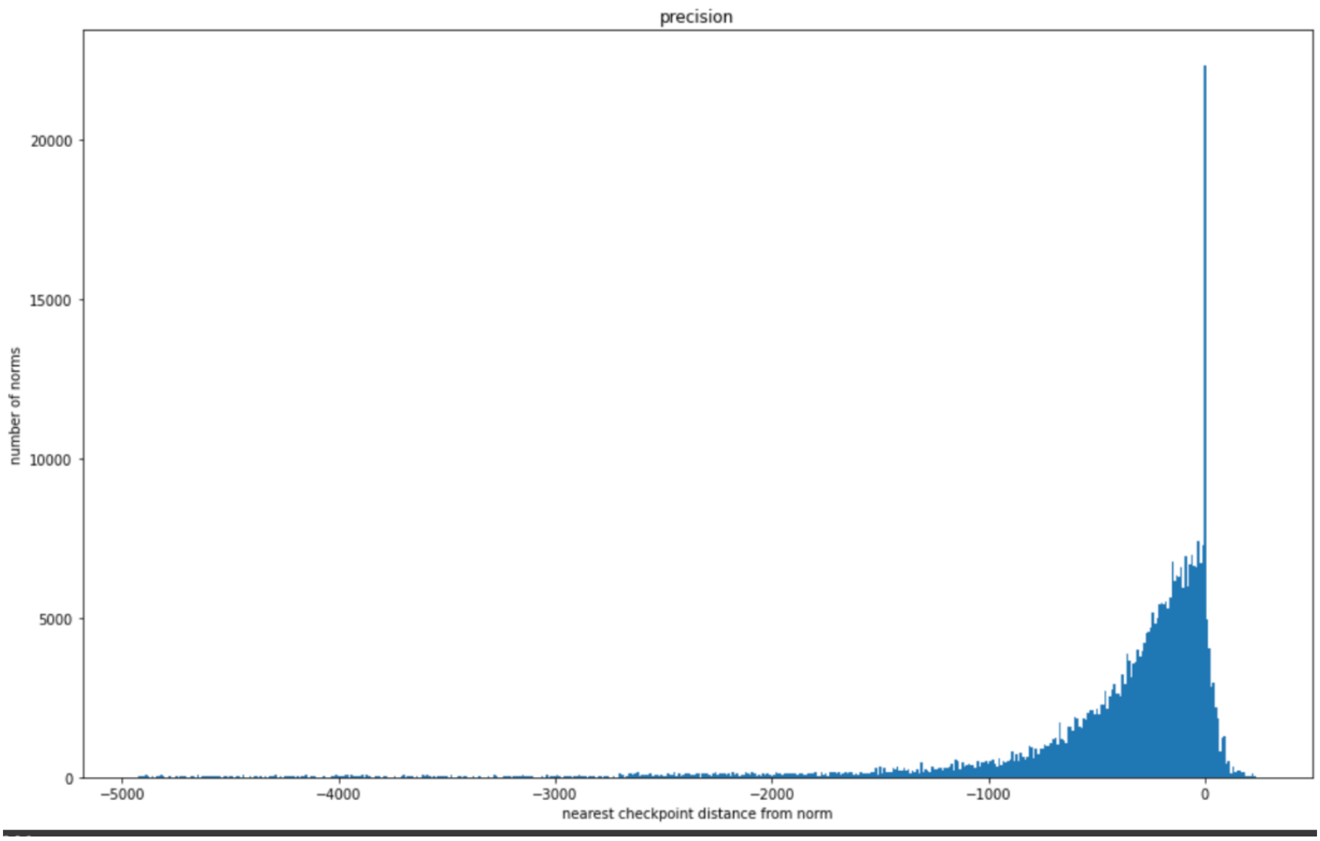
precision
number of checkpoints
nearest norm violation/adherence distance from a checkpoint



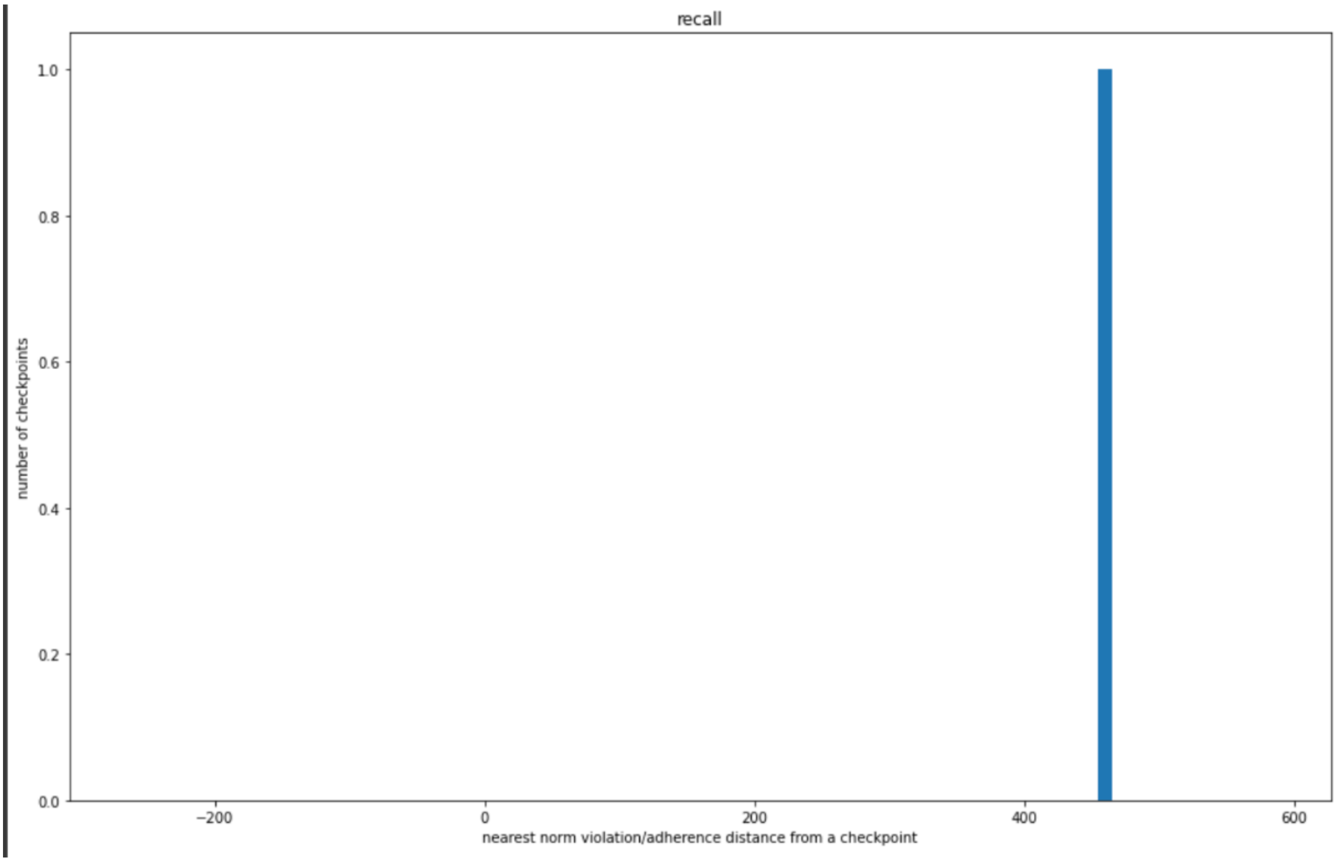
11. Recall and Precision curves for different norm types.

- 105

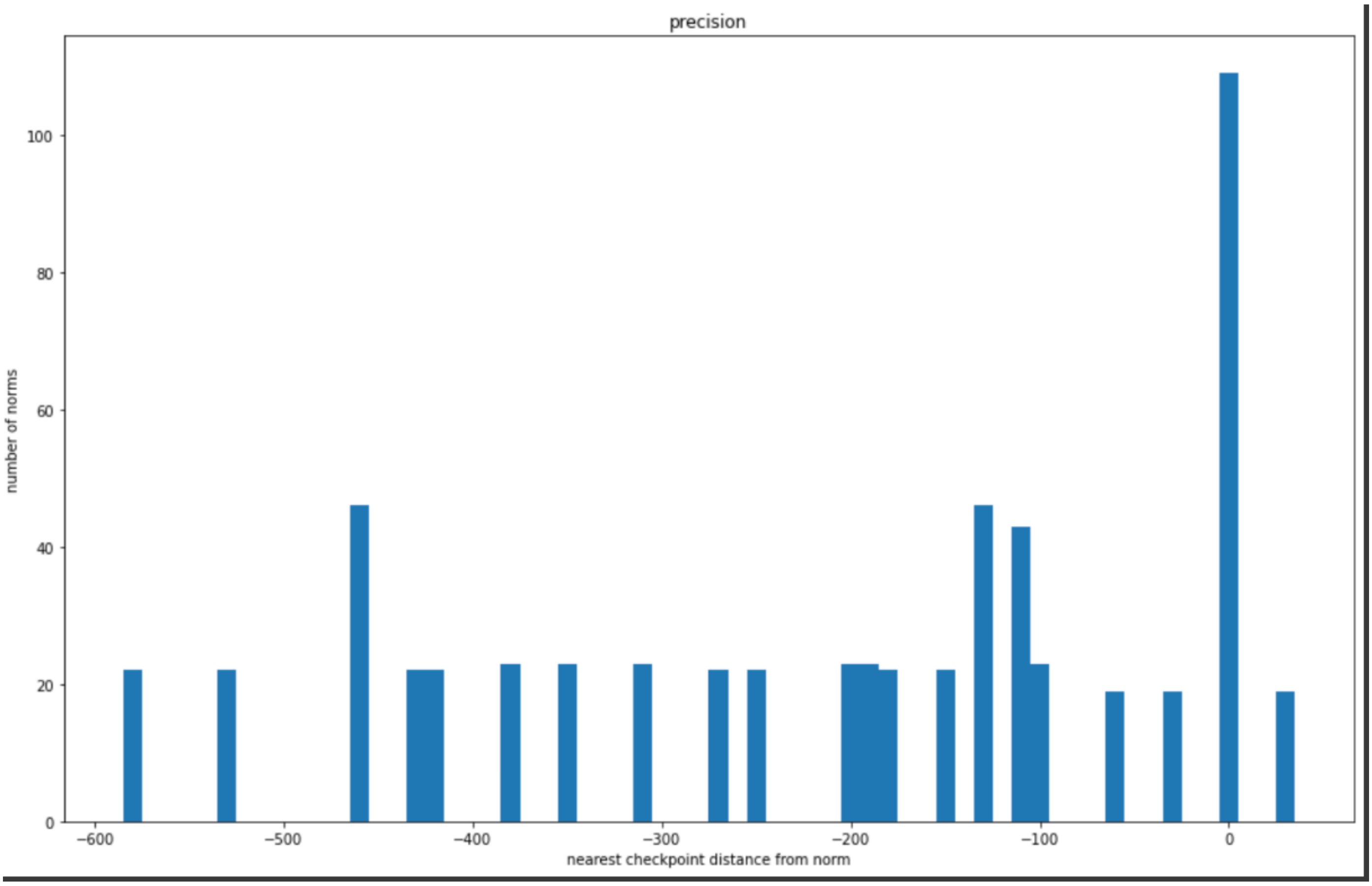

recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint
0
5
10
15
20
25
1000
2000
3000
4000


precision
number of norms
nearest checkpoint distance from norm
0
5000
10000
15000
20000
−5000
−4000
−3000
−2000
−1000

- 201

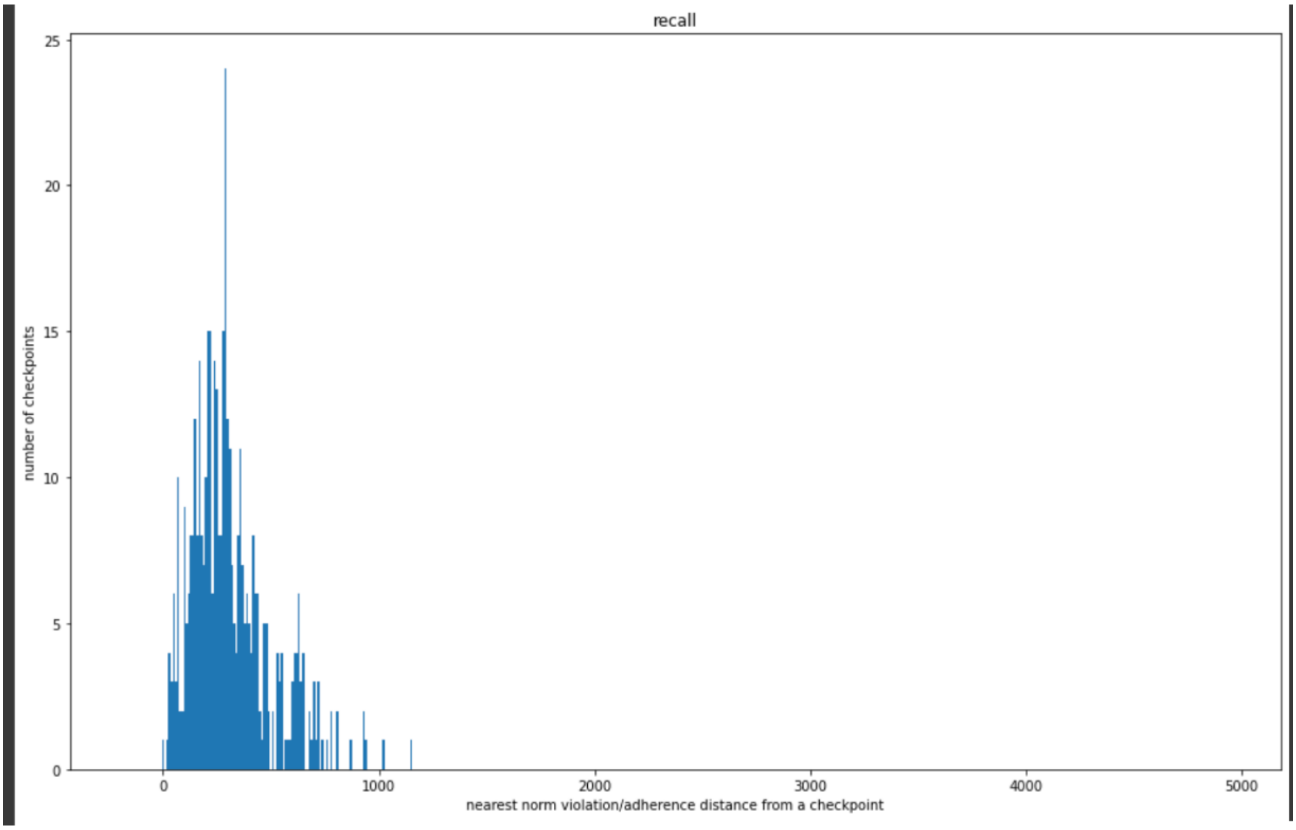


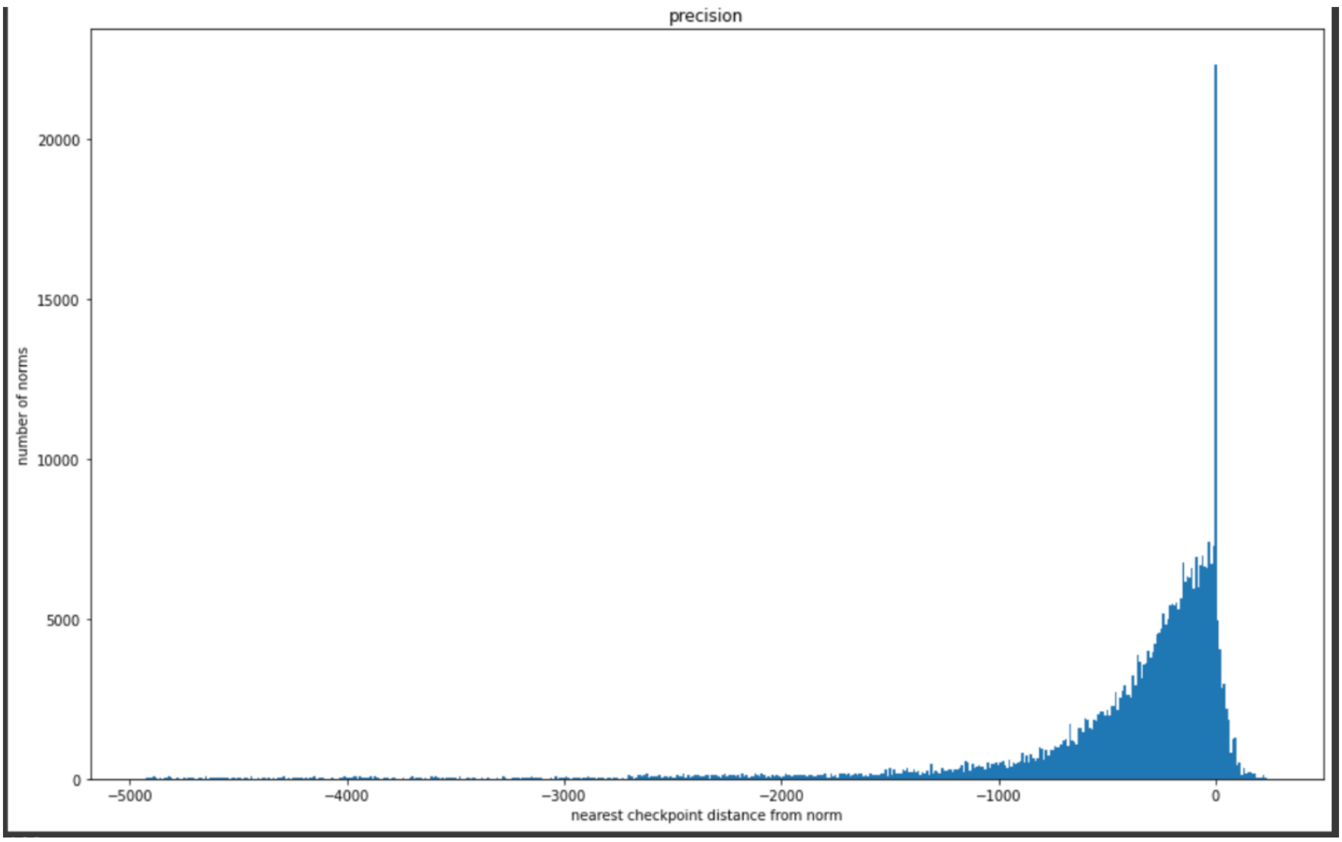
recall
number of checkpoints
1.0
0.8
0.6
0.4
0.2
0.0
−200
0
200
400
600
nearest norm violation/adherence distance from a checkpoint



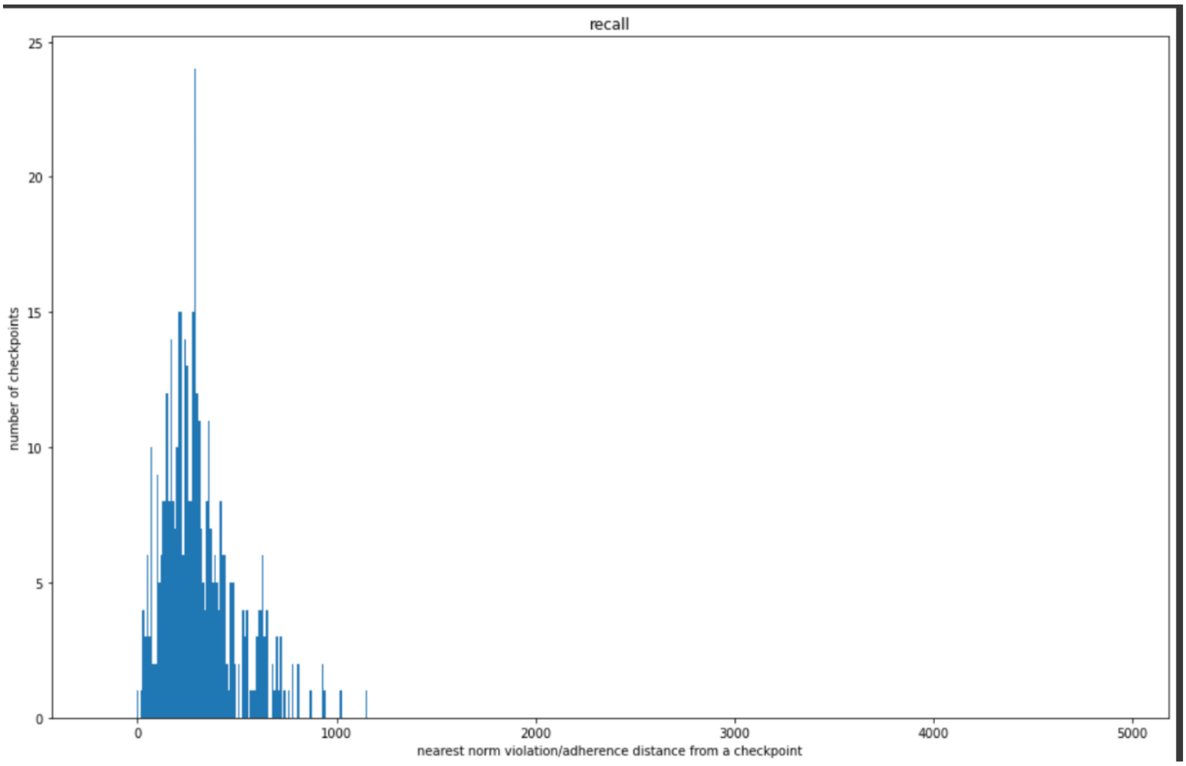
precision
number of norms
100
80
60
40
20
0
−600
−500
−400
−300
−200
−100
0
nearest checkpoint distance from norm

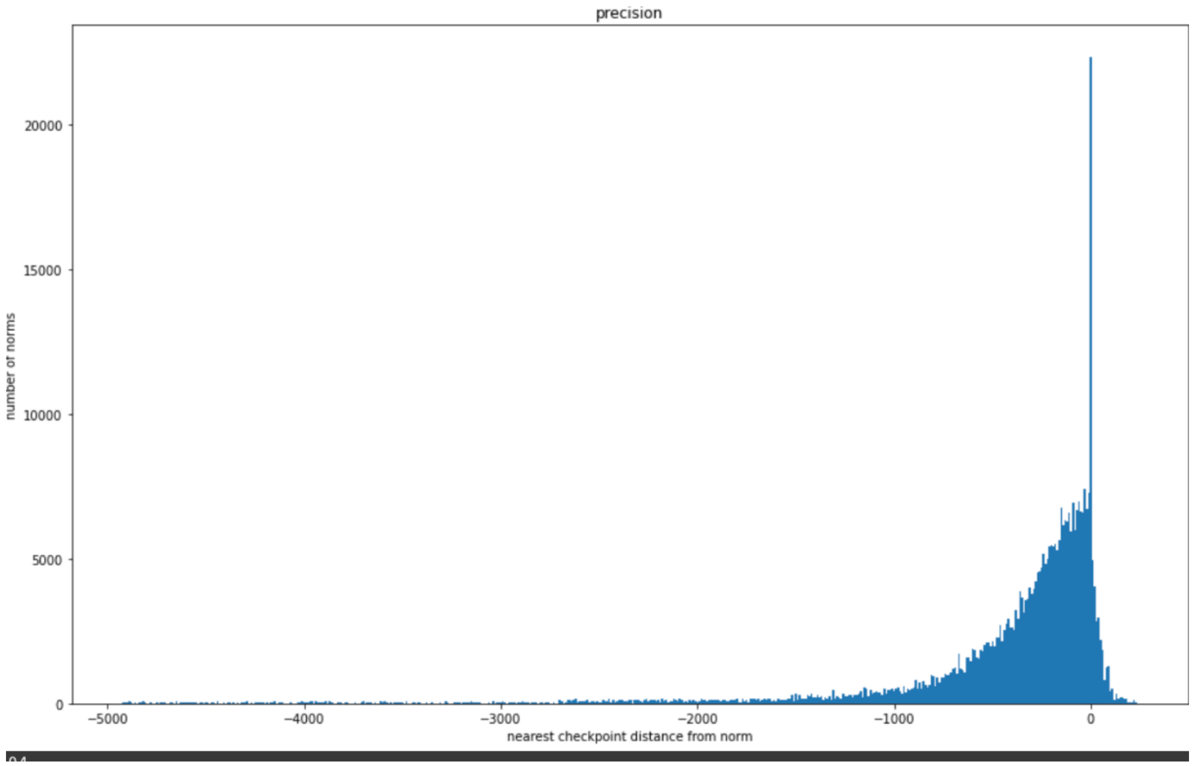
- 103


recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint

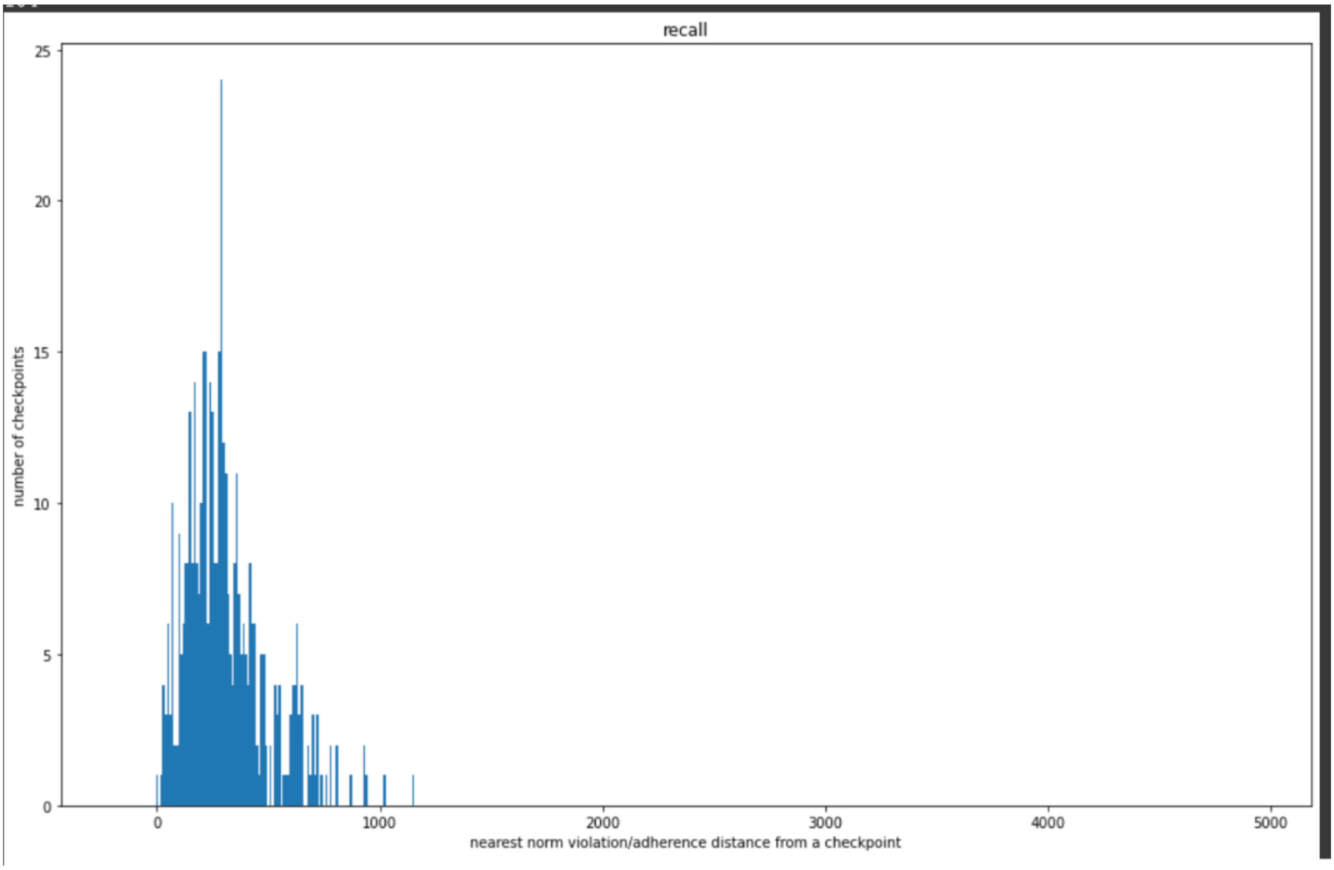

precision
number of norms
nearest checkpoint distance from norm

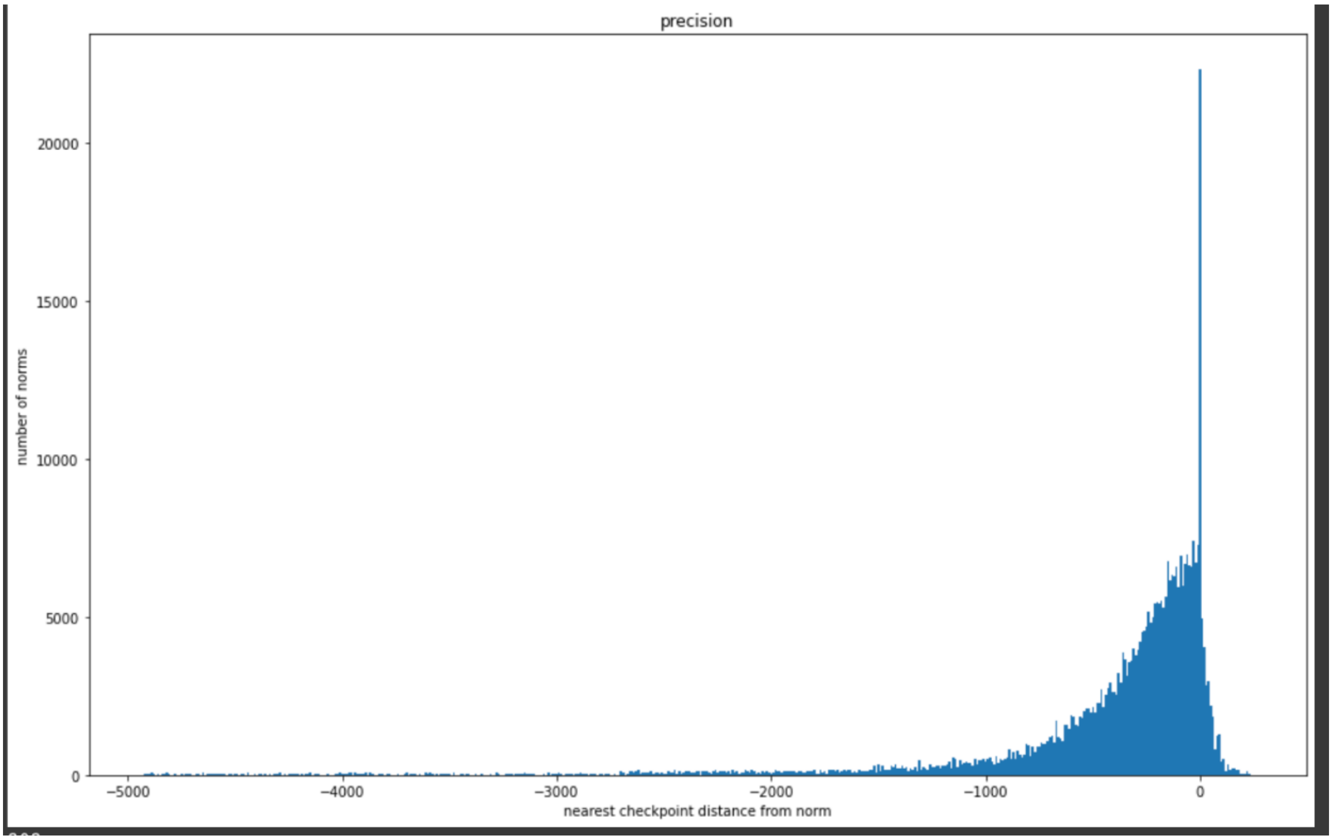
- 102


recall
number of checkpoints
nearest norm violation/adherence distance from a checkpoint


precision
number of norms
nearest checkpoint distance from norm

- 104





- Similarly for others 50 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.

Training our first 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 can use Big pretrained language frameworks 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. The authors of the paper found that while BERT provided an impressive performance boost across multiple tasks it was undertrained. We use the multilingual version of RoBERTa - XLM Roberta.

We have currently built a basic XLM RoBERTa model skeleton which has configurable model and training parameters, down samples the data, stores the best weights and configuration and uses the Adam optimizer and Generalised Cross Entropy loss.
The input format 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)
 }
```

Github link : https://github.com/dm3792/ccu_Dhaarna/blob/main/modelccu.py

We will be training this on the LDC and UIUC data next.

Future work

Training and testing the XLM RoBERTa model on LDC and UIUC data and storing the best configuration.
We will be trying these models in the future to find the best possible model and parameters for changepoint detection using norms:

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.

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