Presentation Outline

Introduction and Motivation

LAW IS ORDER, AND GOOD LAW IS GOOD ORDER.

ARISTOTLE

Related works

- Malik et al. (2021): ILDC for CJPE Indian legal documents corpus for court judgment prediction and explanation.
- Wang et al. (2019): Hierarchical matching network for crime classification.

Data (Quick Overview)

Shared Task Data

Text(legal judgment document)	label	split
CIVIL APPELLATE JURISDICTION Civil Appeal No. 1725 of 1972. Appeal by Special Leave from the Judgment and order dated 22-6-72 of the Bombay High Court in Special Application No. 1441 of 1968. R. Zaiwala, K. J. John and J. S. Sinha for the Appellant. S. Desai, P. B. Agarwala and B. R. Agarwala for Respondent No. 1. The Judgment of the Court was delivered by GOSWAMI, J.-This appeal by special . . .	1	train

Ours Task Data

Text	S.THAKUR, J.Leave granted.These appeals ar...
Constitution art.number	309 , 142 , 141 , 368 , 226
CodeofCriminal art.number	nothing
IndianPenalCode art.number	nothing

Table 1: processed dataset example

main text	constitution text	CrPC text	IPC text
main text	article 309 text	–	–
main text	article 142 text	–	–
main text	article 141 text	–	–
main text	article 368 text	–	–
main text	article 226 text	–	–

Table 2: Feature enriched data

Data

TEXT CLEANING

```
 F NARIMAN J Leave granted In 2008 the Punjab ...
 F NARIMAN J Leave granted In 2008 the Punjab ...
 S THAKUR J Leave granted These appeals are di...
 S THAKUR J Leave granted These appeals are di...
 S THAKUR J Leave granted These appeals are di...
                        ...
civil appellate jurisdiction civil appeal numb...
criminal appellate jurisdiction special leave\...
civil appellate jurisdiction civil appeal numb...
civil appellate jurisdiction civil appeal numb...
```

```
leave grant punjab state issue numberice invit
leave grant punjab state issue numberice invi
leave grant appeal direct order date pass hig
leave grant appeal direct order date pass hig
leave grant appeal direct order date pass hig
                        ...
civil appellate jurisdiction civil appeal num
jurisdiction petition crl number judgment ord
civil appellate jurisdiction civil appeal numb...
civil appellate jurisdiction civil appeal numb...
jurisdiction criminal appeal number appeal spe...
```

LDA ANALYSIS

	case text
Topic 1	0.054048
Topic 2	0.023300
Topic 3	0.078853
Topic 4	0.843799
	$\sum_{n=1}^{4} n = 1$

Table 4: Experiment 2. LDA, 4 topics

n-gram	Topic 1	Topic 2	Topic 3	Topic 4
bigram	death penalty	high court	fundamental rights	high court
	death sentence	police officer	make law	companyid number
trigram	death penalty number	chief criminal procedure	right companyferre part	civil appellate jurisdictions
	death penalty may	chief criminal procedure	power amend constitution	appellate jurisdiction civil

FEATURE ENGINEERING

For finding the relationship between sections of law and case text, we transformed them into vectors with size 500 using the Doc2Vec technique. After finding their vector representation we estimated the cosine angle between the case text vector and the section of the law vector that was referenced in this case. Then these cosines, we used as features for our models.

- cosine(main vector, constitution vector)
- cosine(main vector, CrPC vector)
- cosine(main vector, IPC vector)

PROPOSED IDEAS (CLASSIFICATION)

main text	constitution text	CrPC text	IPC text
main text	article 309 text	–	–
main text	article 142 text	–	–
main text	article 141 text	–	–
main text	article 368 text	–	–
main text	article 226 text	–	–

Table 2: Feature enriched data

LDA Analysis + Cosine Similarity with Shallow Learning

Majority Voting Explanation

We manipulated the dataset's positive and negative instances by artificially increasing their quantity.

Text Classification Deep Learning

LDA Analysis with Deep Learning (RoBERTa)

Classical models

MODELS	F1 SCORE
Extra Tree Classifier (ET)	0.66
XGBoost (Ensemble of DTs)	0.65
Logistic Regression	0.42
Stacking[ET, RF, DT] --(meta-model)Logistic Reg	**0.69**
ILDC Classical Model	0.62
ILDC Sequential Mode	0.64
ILDC Hierarchical Model	**0.77**

Deep Neural Networks models

MODELS	F1 SCORE
BiGRU + attention	0.54
BiLSTM + attention	0.52
CNN	0.62
CNN + constitution embedding	0.60
ILDC Classical Model	0.62
ILDC Sequential Model (BiGRU, BiLSTM)	0.64
ILDC Hierarchical Model (Transformer + Sequential Model)	0.77

Explanation proposed idea

PROPOSED IDEAS (CLASSIFICATION)

main text	constitution text	CrPC text	IPC text
main text	article 309 text	–	–
main text	article 142 text	–	–
main text	article 141 text	–	–
main text	article 368 text	–	–
main text	article 226 text	–	–

Table 2: Feature enriched data

LDA Analysis + Cosine Similarity with Shallow Learning

Presentations are communication tools that can be used as lectures.

Majority Voting Explanation

We manipulated the dataset's positive and negative instances by artificially increasing their quantity.

Text Classification Deep Learning

Presentations are communication tools that can be used as lectures.

LDA Analysis with Deep Learning

Presentations are communication tools that can be used as lectures.

main text	constitution text	CrPC text	IPC text
main text	article 309 text	–	–
main text	article 142 text	–	–
main text	article 141 text	–	–
main text	article 368 text	–	–
main text	article 226 text	–	–

Table 2: Feature enriched data

Majority Voting Explanation

We manipulated the dataset's positive and negative instances by artificially increasing their quantity.

Law1: Approve (70%)

Law2: Approve (54%)

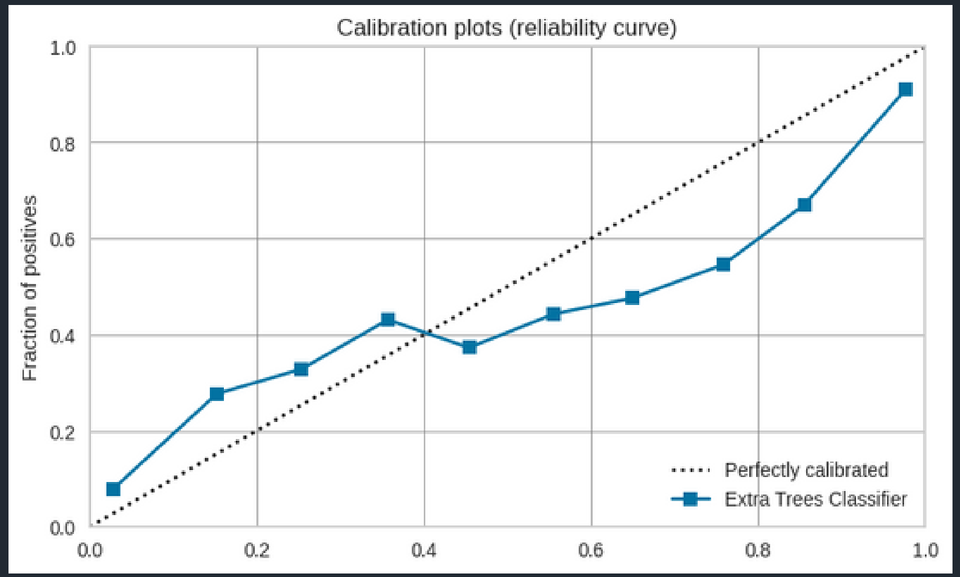
Law3: Reject (30%)

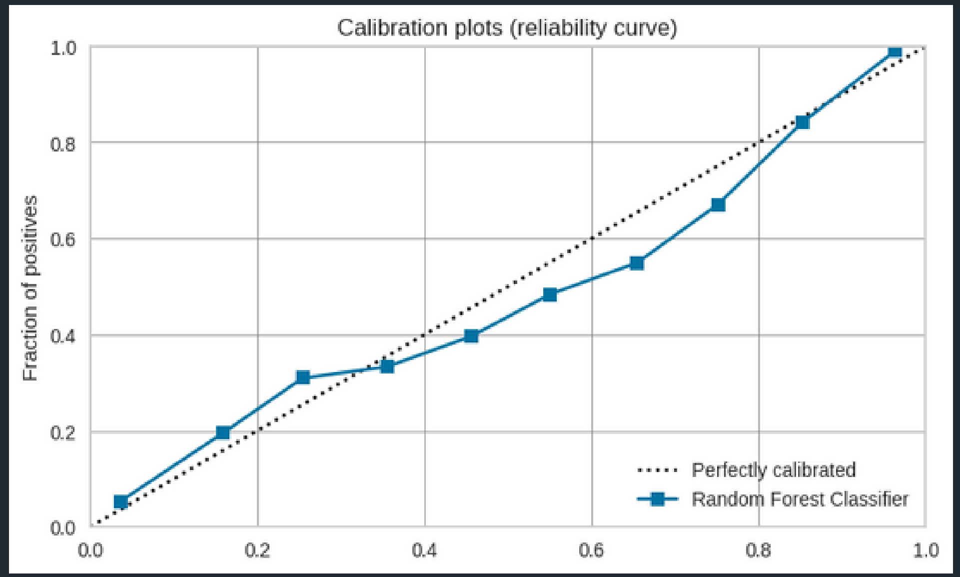
Majority Voting

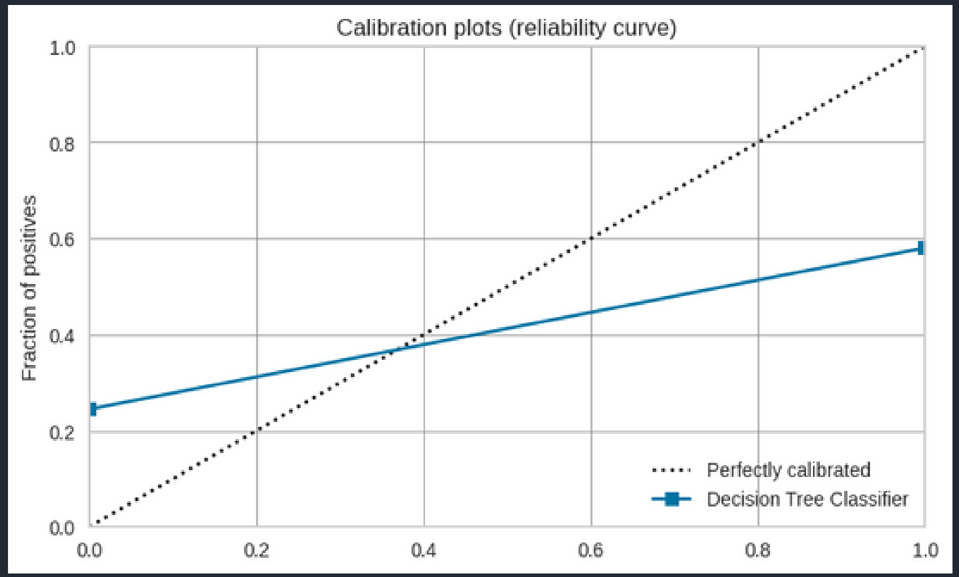
Approve

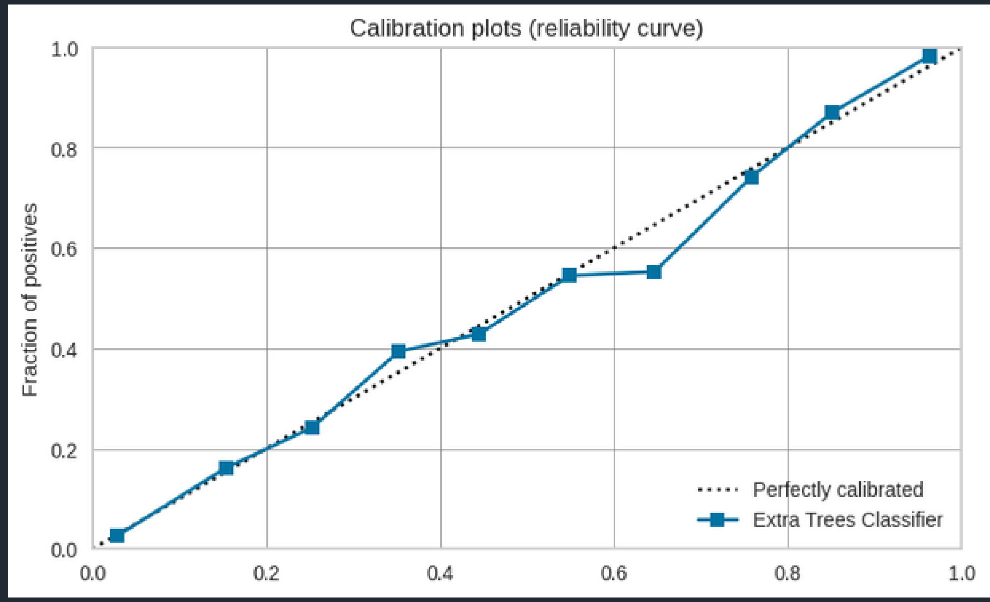
Approve by Law 1 and Law 2

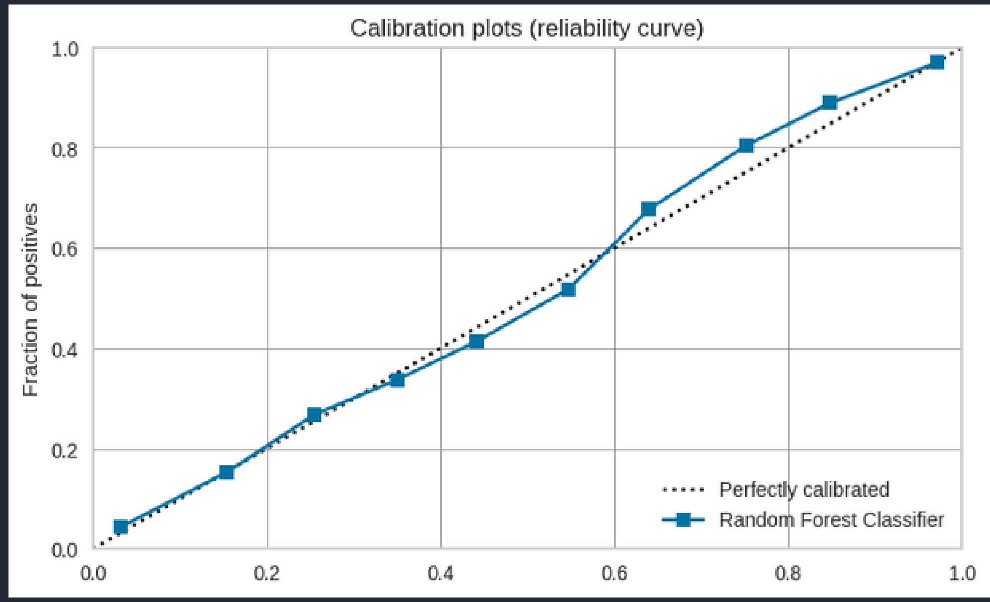
According to [SUMMARY LAW 1] and [SUMMARY LAW 2] this case should be [APPROVED]

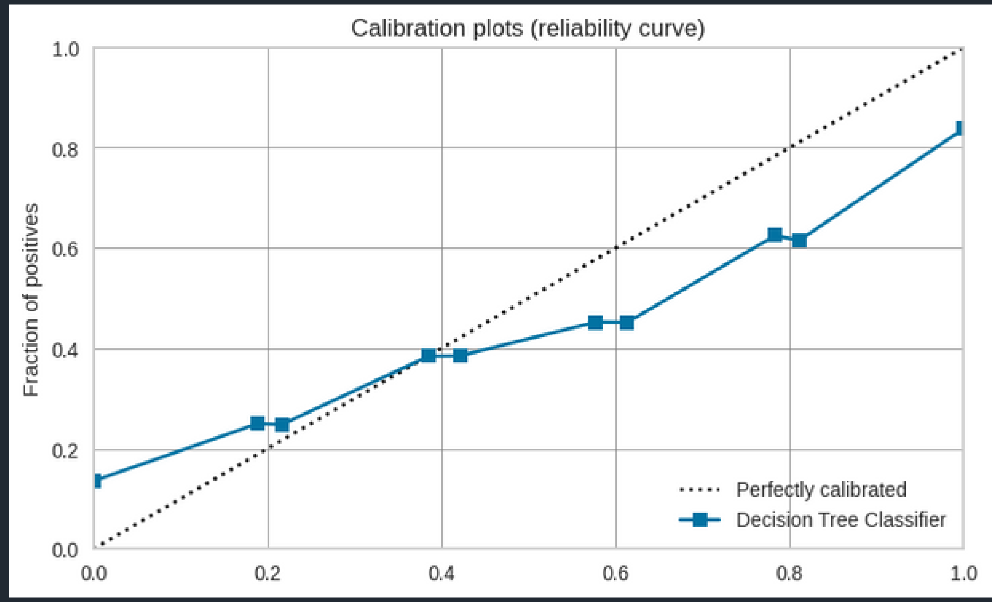




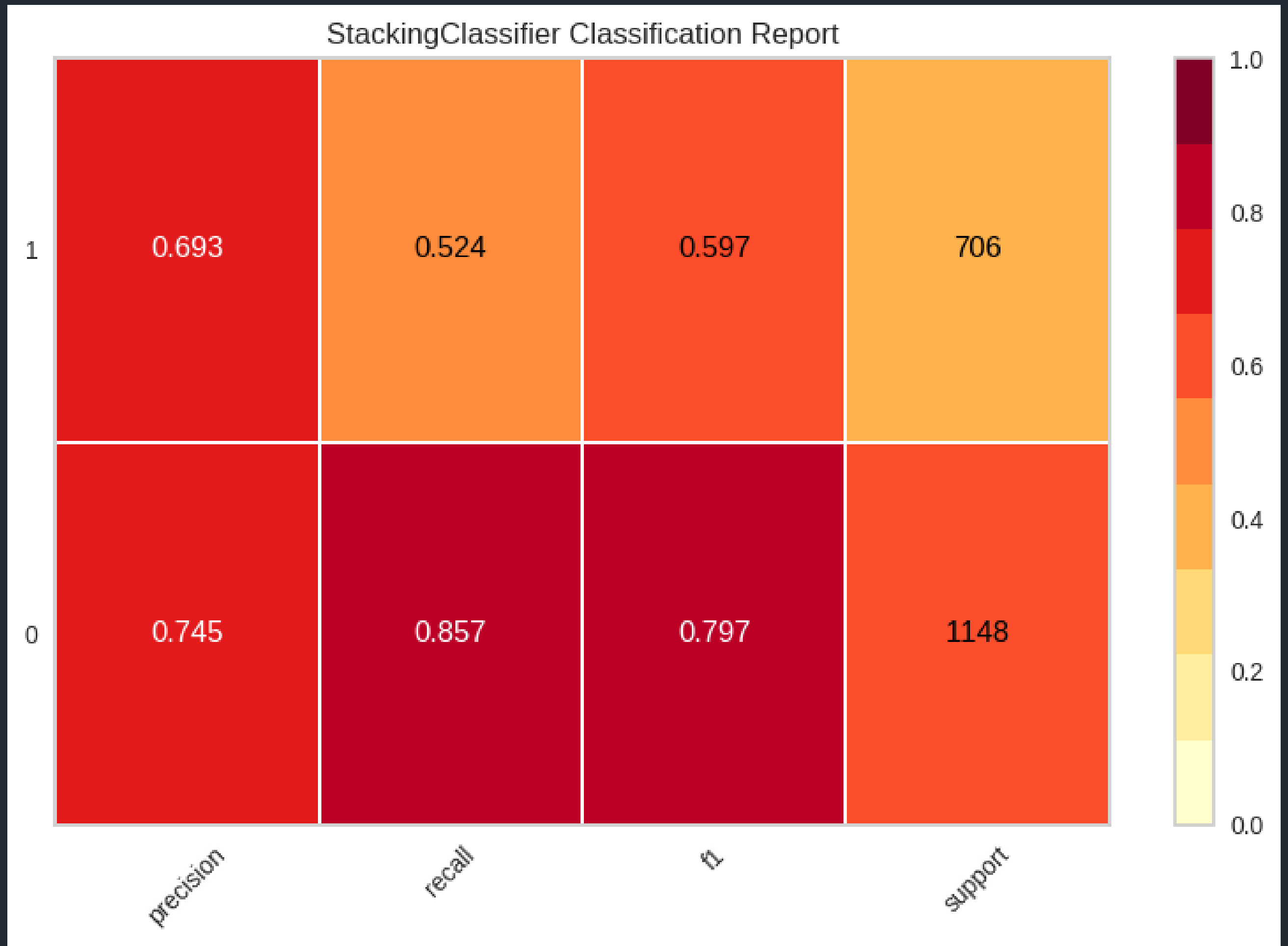




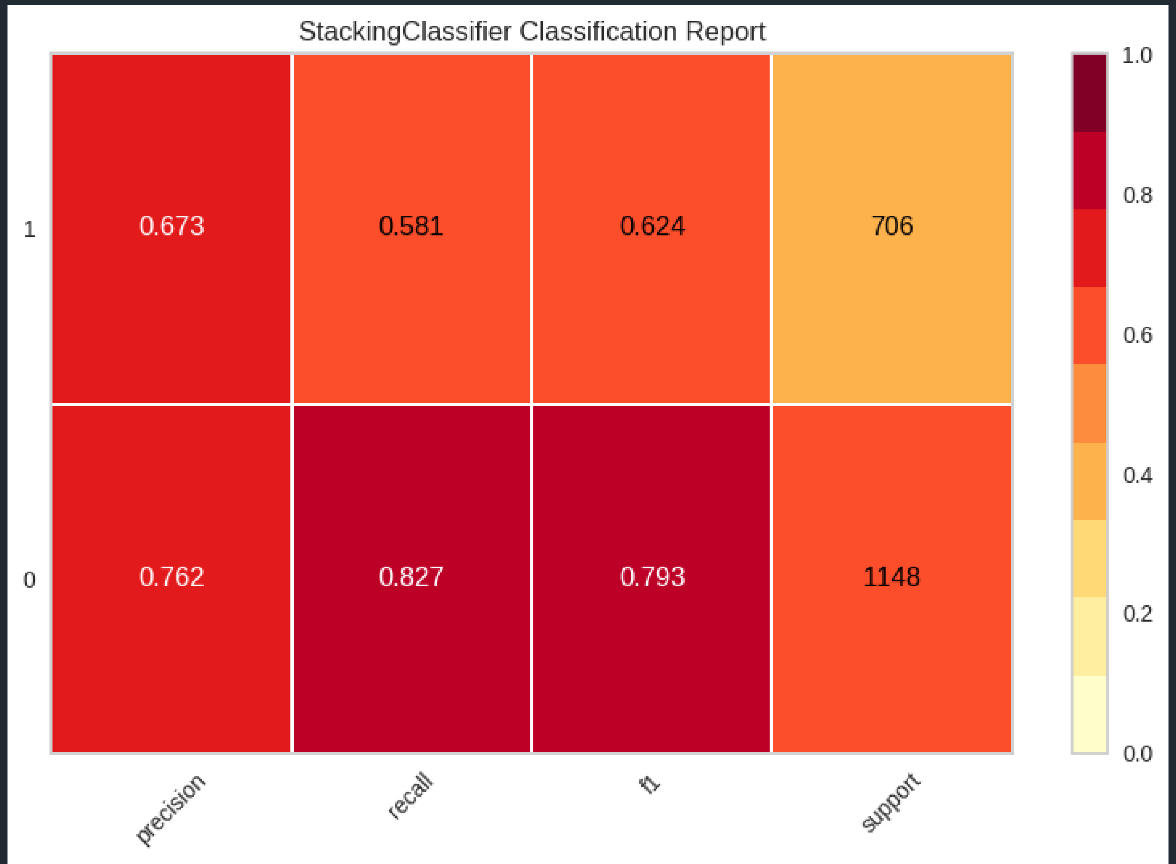




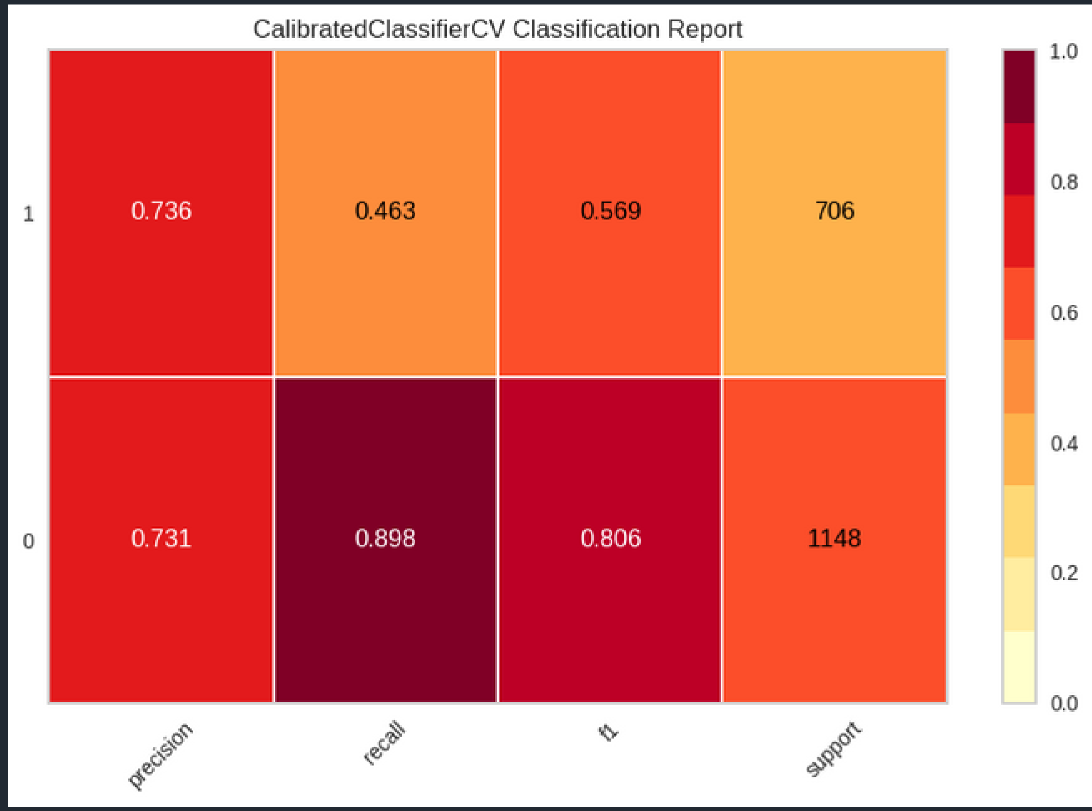
Calibration



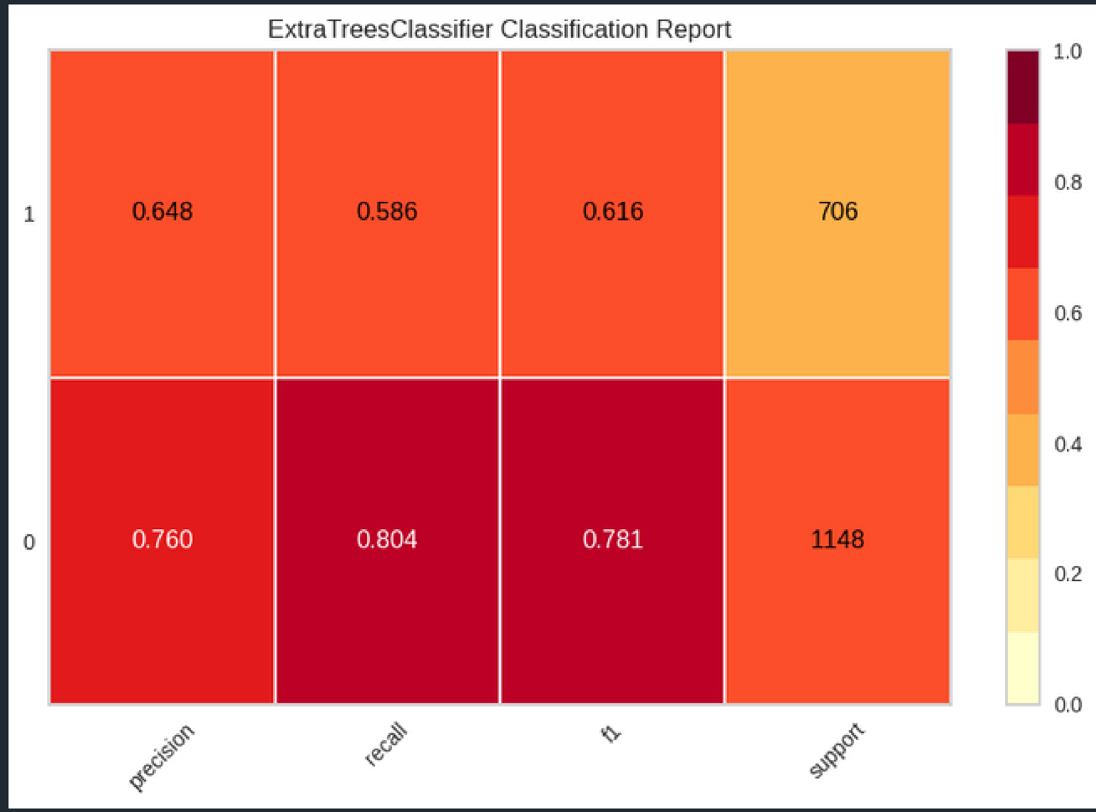
Calibrated



Not Calibrated



Calibrated



Not Calibrated

Future Works & Limitations

```python
from transformers import pipeline
import tqdm

summarizer = pipeline("summarization", model="facebook/bart-large-cnn")
for i, text in tqdm.tqdm(enumerate(data['paragraph'].values.tolist())):
    result = summarizer(text, max_length=30, min_length=10, do_sample=False)[0]['summary_text']
    with open(f'summary/{i}_30_summary.txt', 'w') as f:
        f.write(result)
result
```

✓ 245m 43.4s

```python
from transformers import pipeline
import tqdm

summarizer = pipeline("summarization", model="facebook/bart-large-cnn")
for i, text in tqdm.tqdm(enumerate(data['paragraph'].values.tolist())):
    result = summarizer(text, max_length=100, min_length=50, do_sample=False)
    with open(f'summary100/{i}_100_summary.txt', 'w') as f:
        f.write(result)
result
```

✓ 535m 11.9s

```python
from transformers import pipeline
import tqdm

summarizer = pipeline("summarization", model="sshleifer/distilbart-xsum-12-1")
for i, text in tqdm.tqdm(enumerate(data['paragraph'].values.tolist())):
    result = summarizer(text, max_length=30, min_length=10, do_sample=False)[0]['summary_text']
    with open(f'summary_distilbart/{i}_30_summary.txt', 'w') as f:
        f.write(result)
result
```

✓ 52m 47.6s

```
3821it [52:33,  1.21it/s]
```

```python
from transformers import pipeline
import tqdm

summarizer = pipeline("summarization", model="sshleifer/distilbart-xsum-12-1")
for i, text in tqdm.tqdm(enumerate(data['paragraph'].values.tolist())):
    result = summarizer(text, max_length=100, min_length=50, do_sample=False)[0]['summary_text']
    with open(f'summary_distilbart/{i}_100_summary.txt', 'w') as f:
        f.write(result)
result
```

✓ 86m 49.5s

Questions?