Hasso Plattner Institut
Digital Engineering · Universität Potsdam

ASE 2024 - Assignment 2

Christian, David

Design IT.
Create Knowledge.

Model Training

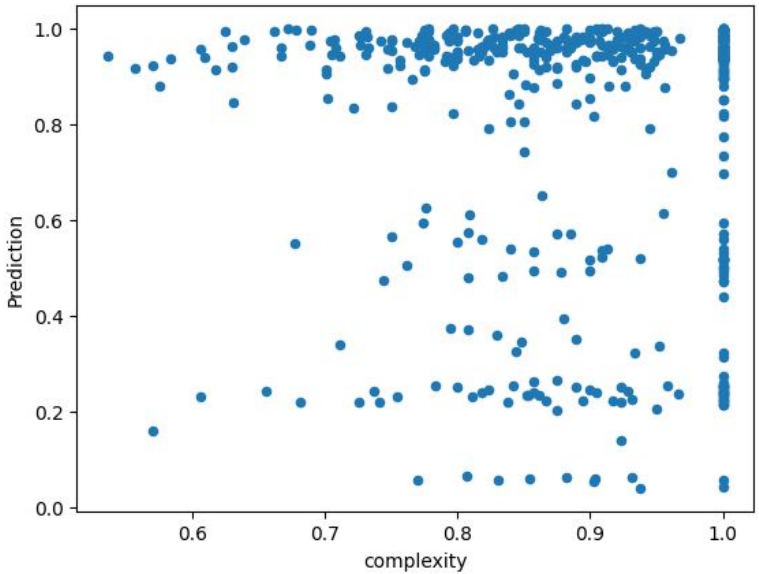
- Random Forest Classifier
  - from Tensorflow Decision Forest (tfdf) python module
- Random splitting of dataset
- Automatic finetuning of hyperparameters using Random Search Tuner

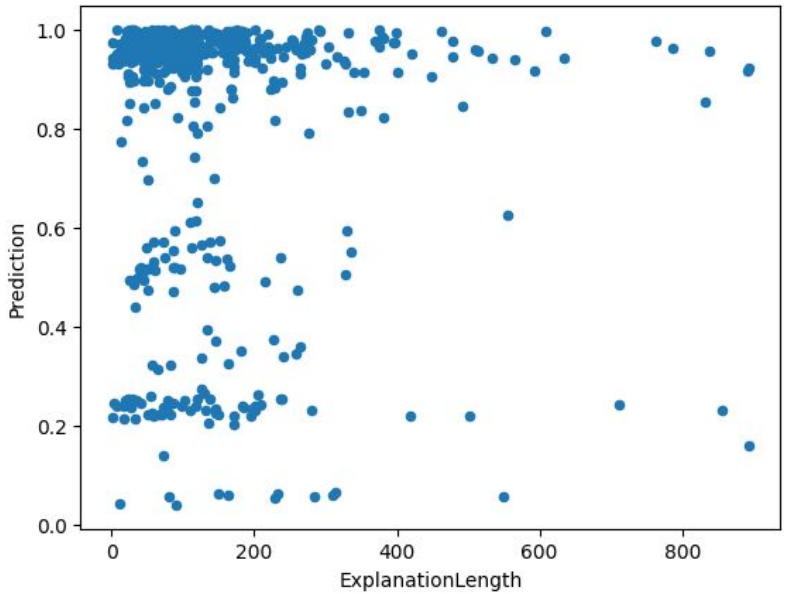
```
Number of nodes by tree:
Count: 300 Average: 57.7867 StdDev: 8.22645
Min: 39 Max: 89 Ignored: 0
---------------------------------------------
[ 39, 41)  2   0.67%   0.67%
[ 41, 44)  6   2.00%   2.67% #
[ 44, 46)  3   1.00%   3.67% #
[ 46, 49) 11   3.67%   7.33% ##
[ 49, 51) 27   9.00%  16.33% #####
[ 51, 54) 54  18.00%  34.33% ##########
[ 54, 56) 40  13.33%  47.67% #######
[ 56, 59) 25   8.33%  56.00% #####
[ 59, 61) 24   8.00%  64.00% ####
[ 61, 64) 50  16.67%  80.67% #########
[ 64, 67) 14   4.67%  85.33% ###
[ 67, 69) 11   3.67%  89.00% ##
[ 69, 72) 17   5.67%  94.67% ###
[ 72, 74)  4   1.33%  96.00% #
[ 74, 77)  4   1.33%  97.33% #
[ 77, 79)  2   0.67%  98.00%
[ 79, 82)  3   1.00%  99.00% #
[ 82, 84)  0   0.00%  99.00%
[ 84, 87)  0   0.00%  99.00%
[ 87, 89]  3   1.00% 100.00% #
```

```
Depth by leafs:
Count: 8818 Average: 6.96042 StdDev: 2.72862
Min: 1 Max: 15 Ignored: 0
---------------------------------------------
[  1,  2)   74   0.84%   0.84% #
[  2,  3)  321   3.64%   4.48% ###
[  3,  4)  512   5.81%  10.29% ####
[  4,  5)  804   9.12%  19.40% #######
[  5,  6) 1055  11.96%  31.37% #########
[  6,  7) 1204  13.65%  45.02% ##########
[  7,  8) 1236  14.02%  59.04% ##########
[  8,  9) 1110  12.59%  71.63% #########
[  9, 10)  917  10.40%  82.03% #######
[ 10, 11)  647   7.34%  89.36% #####
[ 11, 12)  444   5.04%  94.40% ####
[ 12, 13)  270   3.06%  97.46% ##
[ 13, 14)  135   1.53%  98.99% #
[ 14, 15)   59   0.67%  99.66%
[ 15, 15]   30   0.34% 100.00%
```

Model Training

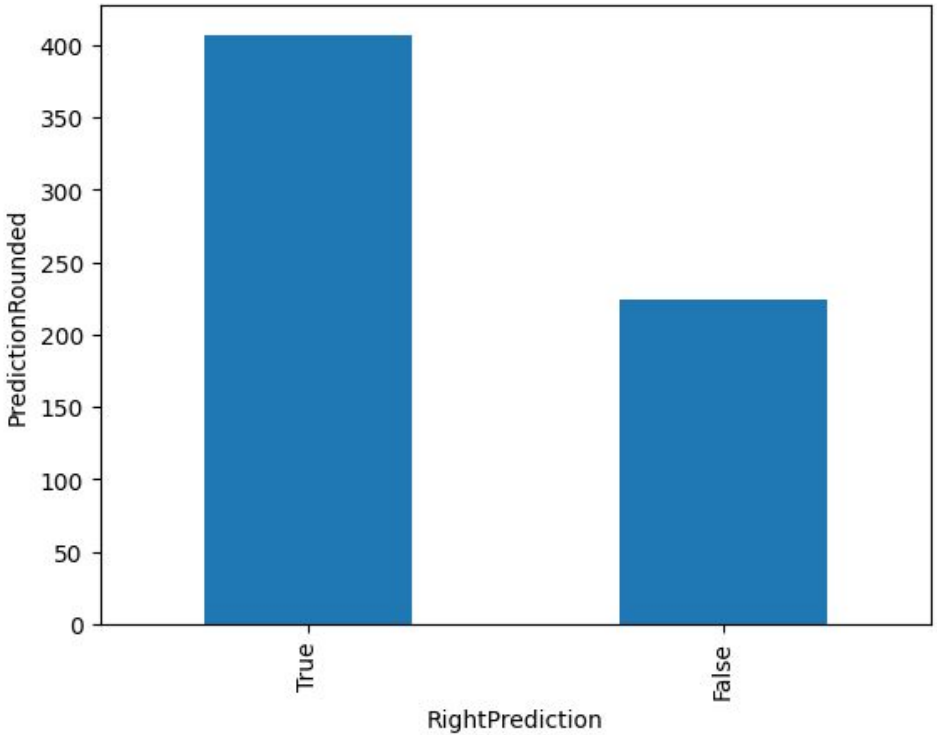
- Distribution of correct labels by complexity and explanation length





Categorize Answers

- Categorization of holdout set prone to errors



Consolidated Explanations

- Summary obtained using Chat-GPT4
  - Summary of all explanations for each method classified as correct
- Short prompt providing only brief instruction and csv containing reference explanations
- Length and complexity of generated explanation varies between methods but is consistent between prompts
- Added context no significant impact on generated explanations

Comparison of Generated and Original Explanations

- Generated and original explanations not very similar
- Dissimilarity between references and summary to be expected
- Bleu and Rouge scores also show low similarity

Bleu Score

```
The hypothesis contains 0 counts of 3-gram overlaps.
Therefore the BLEU score evaluates to 0, independently of
how many N-gram overlaps of lower order it contains.
Consider using lower n-gram order or use SmoothingFunction()
  warnings.warn(_msg)
```

```
The hypothesis contains 0 counts of 4-gram overlaps.
Therefore the BLEU score evaluates to 0, independently of
how many N-gram overlaps of lower order it contains.
Consider using lower n-gram order or use SmoothingFunction()
  warnings.warn(_msg)
```

No Smoothing	With Smoothing
6.867526856655566e-155	0.01772075502789348

Rouge Score

	Rouge-1	Rouge-2	Rouge-4	Rouge-L
Precision	0.10329670329670329	0.008918617614269788	0.0	0.06263736263736264
Recall	0.27647058823529413	0.024464831804281346	0.0	0.1676470588235294
F-Measure	0.1504	0.013071895424836602	0.0	0.09119999999999999

Reflection

- guaranteeing the quality of the input is very subjective and requires manual evaluation
- classifier is very customized to the bugs and programmers from the dataset
  - different bugs might require different classifier
- manual testing of the classifier is very time-intensive
- quality of consolidated explanation is also subjective and hard to evaluate automatically
- difficult to determine if faults are based on the classifier, the LLM or the integration