Software Engineering

WS 2021/22, Assignment 04

Prof. Dr. Sven Apel
Annabelle Bergum
Sebastian Böhm
Christian Hechtl

Handout: 18.01.2022
Handin: 01.02.2022 23:59 CET

Organizational Section:

- The assignment must be accomplished by yourself. You are not allowed to collaborate with anyone. Plagiarism leads to failing the assignment.
- The deadline for the submission is fixed. A late submission leads to a desk reject of the assignment.
- The submission must consist of a *ZIP* archive containing the YAML file named *solution.yaml* and a PDF including your solutions. We only accept solutions with both files.
  - The YAML file needs to be in the specified format and we highly recommend using the template provided on the CMS
  - The PDF must include your solutions, your name, and matriculation number. We only accept solutions created in LaTeX with the template provided on the CMS
- Questions regarding the assignment can be asked in the forum or during tutorial sessions. Please don't share any parts that are specific to your solution, as we will have to count that as attempted plagiarism.

Task 1 [5 Points]

The tool SECompress is a command-line tool that can compress data. Data can be encrypted, signed, segmented, and/or time-stamped. All these features are defined in the Feature Diagram of Figure 1. Since this project is top-secret 😎, we cannot provide you with the source code but instead share the performance measurements with you. Several 20GB files were compressed with different configurations of the tool and the average times in seconds were summarized Table 1.

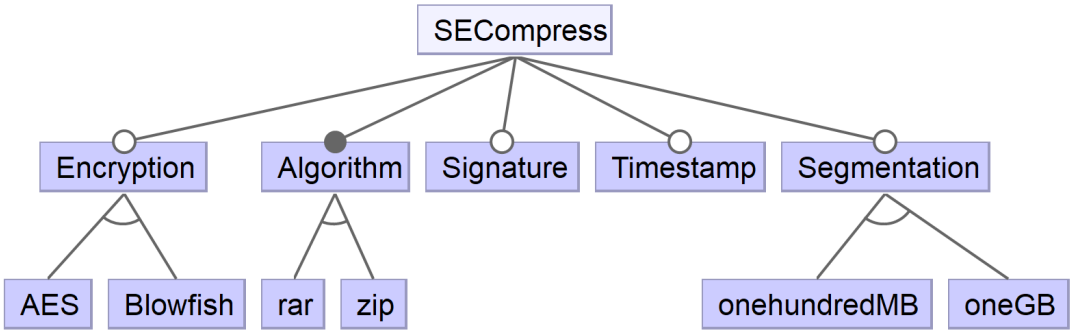

Figure 1: The Feature Diagram of SECompress

Your task is to sample a set of configurations using distance-based sampling introduced in the lecture. The sample shall include as many configurations as possible for each distance, but no more than 4 per distance $d \in \{0, .., 6\}$, in the sample set. The minimum configurations, i.e. {SECompress, Algorithm, rar} and {SECompress, Algorithm, zip}, have a distance of 0. The sampled configurations must be given with their Id, see Table 1, and grouped by their distance value. The sample set has to be inserted in the YAML file.

For example, the distances `42` and `1337` contain the configurations with the Id's `123`, `456` respectively `321`, and `654`. So the submission format looks like Listing 1

```yaml
1. Task:
    # distance 42: configuration Id's 123, 456
    42:
        - 123
        - 456
    # distance 1337: configurations Id's 321, 654
    1337:
        - 321
        - 654
```

Listing 1: The YAML format for the first task

In addition, you have to provide an explanation of how you computed the sample set in the PDF (maximum 200 words).

Task 2 [15 Points]

Your next Task is to construct a *CART* based on your sample set, as presented in the lecture. The CART also has to be stored in the YAML file using the format described in Listing 2. The CART is stored as a nested dictionary with the following fields for each node:

- datapoints: the number of data points that were used
- error_of_split: error of the split as calculated in the lecture
- mean: the mean value of the data points
- name: the name for the root node is `X`. When referring to the left child append `L` and for the right child `R`.
- split_by_feature: the name of the feature as written in the Table 1 (without "-" of course)
- successor_left: contains the left subtree. If there is no left subtree, this item is omitted.
- successor_right: contains the right subtree. If there is no left subtree, this item is omitted.

Additionally, you have to explain your solution in the PDF. First, explain how you have performed the first split. This should also include a reason why the split in that particular case has been chosen, as well as all necessary calculations with the final result each (rounded to 2 decimal places) (maximum 2 pages). Then, briefly outline how the rest of the tree is constructed (maximum 100 words). However, the YAML file must include the complete definition or your CART (rounded to 2 decimal places).

```yaml
2. Task:
  datapoints: 42
  error_of_split: 1337
  mean: 42
  name: X
  split_by_feature: <feature name>
  successor_left:
    datapoints: 21
    error_of_split: 1337
    mean: 42
    name: XL
    split_by_feature: <feature name>
    ...
  successor_right:
    datapoints: 21
    error_of_split: 1337
    mean: 42
    name: XR
    split_by_feature: <feature name>
    ...
```

Listing 2: The YAML format for the second task

Task 3 [5 Points]

Answer the following questions based on your solution of Task 2.

- Given is the following partial configuration:
  $c_{partial} = \{Encryption,\ Algorithm,\ Segmentation\}$
  Use your CART from Task 2 to derive a complete configuration such that the predicted performance is optimal, i.e., there is no other configuration that can be derived from $c_{partial}$ for which your CART predicts a smaller value.
  Please provide the complete configuration via its Id (as explained in Task 1) in the YAML file, as well as an explanation of no more than 100 words on how the configuration was derived by using the model.
- What is the error rate of the model from Task 2?
  Use 10 configurations not included in your sample from Task 1 to calculate the mean error of your CART from Task 2 when predicting the performance of these 10 configurations. The answer should also be given in two parts: first, the value of the error rate and the 10 Id's of the configurations used in the YAML file and, additionally, the calculation method in the PDF with explanation (maximum 3 sentences explanation). (rounded to 2 decimal places in the PDF and YAML)
- Which feature has the highest influence according to your CART from Task 2?
  The answer should contain the name of the feature in the YAML file as well as a 3 sentence explanation in the PDF.

You need to submit your solution in the YAML file, following the example in Listing 3 as well as written explanations in the PDF.

```yaml
3. Task:
    configuration: 654
    error_rate: 8685616.46
    error_sample:
    - 0
    - 1
    - ...
    - 9
    feature_name: <feature name>
```

Listing 3: The YAML format for the third task

Table 1: The given performance measurements

Id	SECompress	Encryption	AES	Blowfish	Algorithm	rar	zip	Signature	Timestamp	Segmentation	100 MB	1 GB	Performance
0	1	0	0	0	1	1	0	0	0	0	0	0	750
1	1	0	0	0	1	1	0	0	0	1	1	0	773
2	1	0	0	0	1	1	0	0	0	1	0	1	770
3	1	0	0	0	1	1	0	0	1	0	0	0	750
4	1	0	0	0	1	1	0	0	1	1	1	0	773
5	1	0	0	0	1	1	0	0	1	1	0	1	770
6	1	0	0	0	1	1	0	1	0	0	0	0	760
7	1	0	0	0	1	1	0	1	0	1	1	0	808
8	1	0	0	0	1	1	0	1	0	1	0	1	795
9	1	0	0	0	1	1	0	1	1	0	0	0	760
10	1	0	0	0	1	1	0	1	1	1	1	0	808
11	1	0	0	0	1	1	0	1	1	1	0	1	795
12	1	0	0	0	1	0	1	0	0	0	0	0	570
13	1	0	0	0	1	0	1	0	0	1	1	0	615
14	1	0	0	0	1	0	1	0	0	1	0	1	610
15	1	0	0	0	1	0	1	0	1	0	0	0	570
16	1	0	0	0	1	0	1	0	1	1	1	0	615
17	1	0	0	0	1	0	1	0	1	1	0	1	610
18	1	0	0	0	1	0	1	1	0	0	0	0	580
19	1	0	0	0	1	0	1	1	0	1	1	0	650
20	1	0	0	0	1	0	1	1	0	1	0	1	635
21	1	0	0	0	1	0	1	1	1	0	0	0	580
22	1	0	0	0	1	0	1	1	1	1	1	0	650
23	1	0	0	0	1	0	1	1	1	1	0	1	635
24	1	1	1	0	1	1	0	0	0	0	0	0	1480
25	1	1	1	0	1	1	0	0	0	1	1	0	1503
26	1	1	1	0	1	1	0	0	0	1	0	1	1500
27	1	1	1	0	1	1	0	0	1	0	0	0	1480
28	1	1	1	0	1	1	0	0	1	1	1	0	1503
29	1	1	1	0	1	1	0	0	1	1	0	1	1500
30	1	1	1	0	1	1	0	1	0	0	0	0	1490
31	1	1	1	0	1	1	0	1	0	1	1	0	1538
32	1	1	1	0	1	1	0	1	0	1	0	1	1525
33	1	1	1	0	1	1	0	1	1	0	0	0	1490
34	1	1	1	0	1	1	0	1	1	1	1	0	1538
35	1	1	1	0	1	1	0	1	1	1	0	1	1525
36	1	1	1	0	1	0	1	0	0	0	0	0	1270
37	1	1	1	0	1	0	1	0	0	1	1	0	1315
38	1	1	1	0	1	0	1	0	0	1	0	1	1310
39	1	1	1	0	1	0	1	0	1	0	0	0	1270
40	1	1	1	0	1	0	1	0	1	1	1	0	1315
41	1	1	1	0	1	0	1	0	1	1	0	1	1310
42	1	1	1	0	1	0	1	1	0	0	0	0	1280
43	1	1	1	0	1	0	1	1	0	1	1	0	1350
44	1	1	1	0	1	0	1	1	0	1	0	1	1335
45	1	1	1	0	1	0	1	1	1	0	0	0	1280
46	1	1	1	0	1	0	1	1	1	1	1	0	1350
47	1	1	1	0	1	0	1	1	1	1	0	1	1335
48	1	1	0	1	1	1	0	0	0	0	0	0	950
49	1	1	0	1	1	1	0	0	0	1	1	0	973
50	1	1	0	1	1	1	0	0	0	1	0	1	970
51	1	1	0	1	1	1	0	0	1	0	0	0	950
52	1	1	0	1	1	1	0	0	1	1	1	0	973
53	1	1	0	1	1	1	0	0	1	1	0	1	970
54	1	1	0	1	1	1	0	1	0	0	0	0	960
55	1	1	0	1	1	1	0	1	0	1	1	0	1008
56	1	1	0	1	1	1	0	1	0	1	0	1	995
57	1	1	0	1	1	1	0	1	1	0	0	0	960
58	1	1	0	1	1	1	0	1	1	1	1	0	1008

59	1	1	0	1	1	1	0	1	1	1	0	1	995
60	1	1	0	1	1	0	1	0	0	0	0	0	770
61	1	1	0	1	1	0	1	0	0	1	1	0	815
62	1	1	0	1	1	0	1	0	0	1	0	1	810
63	1	1	0	1	1	0	1	0	1	0	0	0	770
64	1	1	0	1	1	0	1	0	1	1	1	0	815
65	1	1	0	1	1	0	1	0	1	1	0	1	810
66	1	1	0	1	1	0	1	1	0	0	0	0	780
67	1	1	0	1	1	0	1	1	0	1	1	0	850
68	1	1	0	1	1	0	1	1	0	1	0	1	835
69	1	1	0	1	1	0	1	1	1	0	0	0	780
70	1	1	0	1	1	0	1	1	1	1	1	0	850
71	1	1	0	1	1	0	1	1	1	1	0	1	835