The Normalized Compression Distance as a File Fragment Classifier

By

Stefan Axelsson

Presented At

The Digital Forensic Research Conference

DFRWS 2010 USA Portland, OR (Aug 2nd - 4th)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

http:/dfrws.org

The Normalized Compression Distance as a File Fragment Classifier

Stefan Axelsson

Blekinge Institute of Technology, Sweden

stefan.axelsson@bth.se

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

The problem

- An analyst is often faced with an assortment of file fragments from slack space on e.g. usb-sticks, hard-disks etc.
- Often enough of the header hasn't survived to help identify the type of the file fragment
  - The type is an important piece of information when trying to reconstruct the actual files (or parts of them at least)
- Thus automatic file fragment type classification is a worthwhile endeavour

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

The approach

- Normalised compression distance:
  - Built on the idea that modern compression algorithms are close to the (incomputable) Kolmogorov complexity measure of distance
  - The idea is that by compressing two vectors (A and B) and then the concatenation (AB) then the shorter the compressed concatenation is compared to A (or B) then the "closer" they are, i.e. the less redundancy there is between them
  - Formally:

$$NCD(x,y) = \frac{C(x,y) - min(C(x), C(y))}{max(C(x), C(y))}$$

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Classification

- Straight old k-nearest-neighbour (for different values of 'k')
  - i.e. with k=10 if three of the closest feature vectors were of type zip and seven were of type exe, then the feature vector under classification is assigned class exe, even if the closest example might have been a zip feature vector
- We do the full n-valued classification here, i.e. we don't try and classify pairs of files against each other, but classify all file fragments into one of the (twenty-eight) different file types

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

The corpus

- In order to promote comparison between approaches we've used the public corpus by Simpson et.al.
- In particular the 1 million files of different file types
- It contains 28 different types of files (based on file name endings)
  - pdf, html, jpg, text, doc, xls, ppt, xml, gif, ps, csv, gz, eps, png, swf, pps, sql, java, pptx, docx, ttf, js, pub, bmp, xbm, xlsx, jar, and zip

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Experimental data

- As a rule fourteen 512byte fragments were selected at random from each file (for files that were long enough)
  - 512 was chosen since that's the minimum frag size in common use and also the most conservative choice
- In total close to 32000 fragments were chosen
  - Since run time is quadratic and in lieu of doing ten-fold cross validation we ran the experiment ten times on ca 3000 blocks each time

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

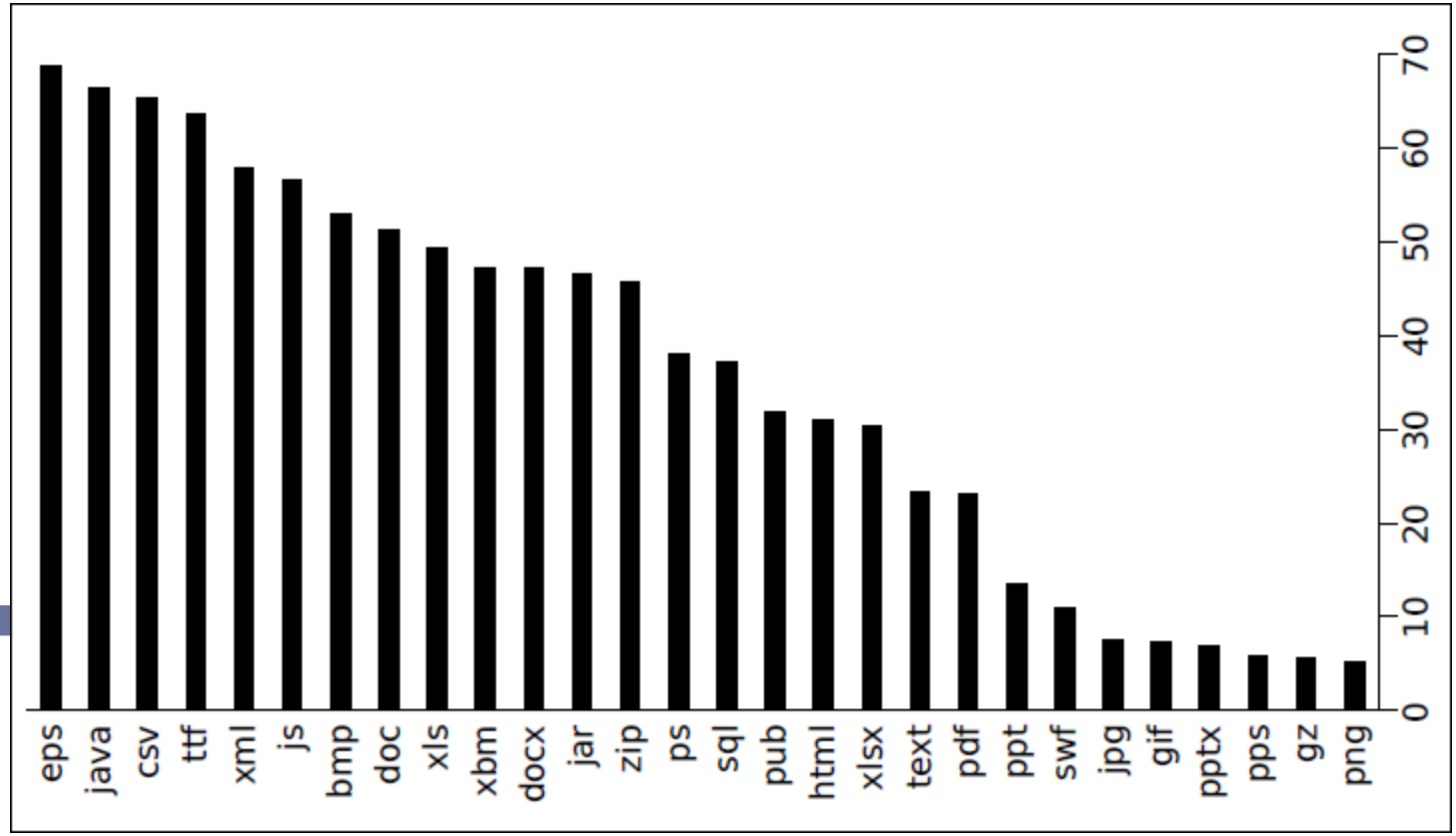
Overall results

- Overall accuracy (random would be ≈ 3.5%, i.e. 1/28

k	Average hit rate (%)
1	36.43
2	34.95
3	34.96
4	34.31
5	34.17
6	33.65
7	33.50
8	33.06
9	33.00
10	32.86

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Results per file type



Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Class	1	σ	2	σ	3	σ	4	σ	5	σ	6	σ	7	σ	8	σ	9	σ	10	σ
eps	69	(8.1)	64	(13)	70	(10)	67	(13)	70	(10)	69	(12)	71	(9.1)	68	(12)	70	(11)	70	(12)
java	68	(12)	62	(12)	66	(12)	67	(12)	68	(12)	67	(13)	66	(13)	67	(13)	67	(13)	67	(13)
csv	67	(12)	61	(14)	63	(14)	64	(12)	65	(13)	65	(13)	66	(13)	67	(12)	66	(13)	67	(12)
ttf	69	(12)	73	(12)	69	(11)	66	(11)	62	(10)	62	(9.0)	59	(10)	60	(8.3)	58	(9.0)	58	(7.5)
xml	58	(15)	56	(16)	58	(15)	58	(15)	59	(15)	58	(14)	58	(14)	58	(15)	58	(15)	57	(15)
js	62	(12)	59	(13)	58	(15)	56	(16)	58	(17)	57	(18)	57	(18)	55	(18)	54	(19)	52	(22)
bmp	51	(12)	39	(12)	48	(9.2)	53	(11)	54	(8.9)	56	(9.0)	57	(10)	56	(11)	57	(10)	58	(10)
doc	48	(10)	56	(13)	54	(11)	53	(12)	50	(10)	52	(11)	49	(11)	50	(13)	48	(14)	52	(15)
xls	51	(13)	46	(10)	48	(10)	47	(12)	50	(13)	49	(13)	51	(14)	50	(13)	51	(14)	50	(13)
xbm	61	(14)	51	(17)	51	(17)	50	(18)	50	(18)	44	(18)	45	(19)	40	(19)	40	(20)	39	(19)
docx	47	(14)	44	(13)	35	(15)	45	(18)	49	(20)	49	(21)	50	(22)	50	(22)	51	(23)	51	(22)
jar	59	(8.9)	55	(9.2)	47	(11)	49	(13)	48	(13)	44	(15)	44	(14)	40	(12)	39	(11)	39	(12)
zip	61	(17)	44	(24)	51	(21)	45	(22)	44	(21)	43	(21)	43	(22)	42	(23)	41	(22)	42	(23)
ps	40	(14)	43	(14)	39	(14)	39	(15)	37	(15)	38	(14)	37	(14)	37	(14)	36	(14)	36	(15)
sql	40	(12)	38	(11)	37	(12)	37	(12)	37	(12)	37	(14)	36	(14)	36	(14)	37	(15)	36	(15)
pub	52	(26)	43	(22)	36	(22)	33	(22)	31	(21)	29	(19)	27	(20)	24	(19)	22	(18)	21	(19)
html	25	(10)	36	(10)	35	(10)	34	(8)	31	(9)	31	(9)	30	(10)	30	(9)	29	(10)	28	(10)
xlsx	43	(13)	20	(11)	30	(14)	31	(17)	32	(16)	31	(18)	29	(17)	29	(19)	30	(20)	29	(20)
text	19	(10)	27	(12)	25	(13)	23	(12)	23	(12)	23	(12)	22	(12)	23	(12)	23	(12)	24	(13)
pdf	23	(9)	26	(10)	25	(10)	24	(10)	24	(11)	23	(10)	22	(11)	21	(11)	21	(10)	21	(10)
ppt	14	(7.7)	14	(9.5)	11	(7.8)	13	(9.2)	13	(8.1)	12	(9.4)	14	(7.8)	15	(8.5)	14	(8.7)	15	(8.6)
swf	11	(16)	11	(15)	11	(14)	10	(15)	10	(16)	12	(15)	12	(14)	12	(13)	10	(12)	10	(11)
jpg	7.2	(6.5)	12	(10)	11	(9.4)	9.2	(7.7)	7.8	(6.8)	6.3	(6.4)	5.4	(6.0)	5.7	(5.7)	5.4	(5.4)	5.2	(4.0)
gif	7.4	(5.7)	14	(10)	12	(7.7)	8.4	(5.2)	6.2	(4.5)	5.3	(4.6)	5.5	(4.4)	5.4	(3.6)	5.0	(3.7)	4.9	(3.8)
pptx	8.9	(8.9)	12	(10)	7.9	(4.3)	5.8	(5.7)	6.0	(7.3)	6.2	(7.5)	5.9	(5.9)	6.1	(6.1)	5.8	(5.3)	4.7	(5.0)
pps	6.8	(4.0)	4.1	(2.9)	5.4	(4.4)	4.7	(3.8)	5.1	(5.0)	5.1	(4.8)	5.7	(5.3)	6.3	(5.2)	7.0	(5.2)	7.8	(5.1)
gz	4.4	(3.8)	8.7	(6.3)	8.2	(6.0)	5.2	(3.8)	4.0	(5.6)	4.8	(6.2)	5.0	(6.0)	5.7	(6.1)	5.8	(4.9)	4.9	(3.8)
png	5.5	(4.0)	11	(8.1)	8.0	(7.3)	4.3	(4.7)	4.2	(3.4)	3.6	(3.1)	3.6	(3.2)	4.2	(4.0)	3.9	(3.6)	3.8	(4.7)

Min accuracy per frag type (%)

Class	min
java	45.2
ttf	38.3
eps	37.9
csv	33.3
doc	25.3
xml	25.2
xls	23.4
jar	21.3
bmp	16.7
html	14.2
js	12
sql	11.1
xbm	11
pdf	10.7
xlsx	5.7
zip	5.4
docx	5.3
pub	4
text	2.5
ps	2.4
gif	0.7
gz	0.7
jpg	0.7
png	0.7
ppt	0.7
pptx	0.7
swf	0.7
pps	0.6

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Class k-value	pps $k = 10$	gz $k = 2$	png $k = 2$	swf $k = 7$	jpg $k = 2$	pptx $k = 2$	gif $k = 2$	ppt $k = 10$
bmp	27	3	3	5	5	8	10	35
csv	1							
doc	74	2	2	50	5	33	32	117
docx	495	470	470	464	381	400	440	435
eps	8			2	2	9	3	4
gif	44	128	128	63	136	122		23
gz	42			26	97	87	85	40
html	7			12				4
jar	2							2
java								4
jpg	46	90	90	50		107	60	30
js		2	2	4				
pdf	20	25	25	9	44	22	26	6
png	36	122	122	66	134	125	111	31
pps		31	31	15	24	24	20	66
ppt	93	10	10	6	15	18	10	
pptx	44	138	138	75	117		118	20
ps	1							
pub	1			2		1	1	
sql				1			1	
swf	14	36	36		40	36	22	9
text	3					1	3	2
ttf		7	7					3
xbm		1	1					
xls	77	23	23	25	21	24	34	61
xlsx	339	117	117	309	119	147	130	223
xml	1			1	1		1	5
zip		3	3	5				5
Sum	1375	1208	1067	1190	1141	1164	1107	1125

The prototype detector

- 20 fragments that were the best at predicting their brethren were chosen for each file type (counting excluded fragments from the same file)
- In total 560 fragments
- This includes examples of file types that do not do that well as the classifier would be blind to these types otherwise
- Downloaded ten files for each type at random and did a similar experiment as before (but only five runs)

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng

Class	1	σ	2	σ	3	σ	4	σ	5	σ	6	σ	7	σ	8	σ	9	σ	10	σ
pub	67	(6)	59	(11)	50	(8)	57	(9)	48	(7)	48	(8)	47	(7)	51	(8)	46	(7)	49	(8)
eps	61	(6)	61	(9)	61	(8)	63	(9)	63	(9)	63	(10)	61	(8)	61	(9)	59	(7)	58	(9)
csv	59	(10)	55	(8)	59	(11)	59	(10)	61	(11)	62	(10)	62	(9)	63	(10)	63	(9)	63	(9)
java	53	(12)	52	(14)	50	(12)	52	(13)	52	(12)	53	(13)	53	(13)	53	(13)	53	(13)	52	(14)
xml	50	(8)	51	(11)	51	(9)	50	(9)	49	(8)	49	(7)	49	(6)	50	(7)	50	(7)	50	(5)
ttf	48	(16)	49	(16)	49	(16)	49	(16)	49	(16)	49	(16)	49	(16)	49	(16)	49	(16)	49	(16)
bmp	46	(10)	38	(10)	45	(11)	47	(10)	47	(12)	49	(12)	50	(12)	50	(11)	51	(12)	53	(12)
xlsx	43	(5)	22	(7)	33	(7)	28	(9)	24	(9)	20	(9)	15	(10)	13	(10)	12	(9)	12	(9)
sql	36	(20)	35	(20)	32	(20)	35	(21)	33	(20)	35	(21)	36	(20)	35	(20)	35	(18)	36	(18)
zip	35	(2)	26	(3)	35	(2)	39	(4)	34	(4)	36	(5)	36	(5)	39	(6)	45	(5)	47	(6)
js	35	(7)	35	(7)	35	(7)	36	(6)	37	(7)	38	(7)	37	(7)	38	(6)	37	(7)	38	(7)
xls	34	(7)	36	(4)	35	(4)	35	(4)	36	(5)	36	(4)	38	(5)	40	(5)	53	(8)	51	(8)
docx	31	(5)	39	(5)	26	(6)	42	(6)	52	(8)	50	(5)	43	(3)	43	(4)	45	(3)	49	(3)
text	29	(5)	27	(7)	26	(7)	23	(6)	23	(7)	22	(7)	22	(8)	23	(8)	22	(9)	22	(8)
html	28	(7)	39	(9)	36	(8)	33	(8)	31	(9)	31	(8)	30	(9)	30	(8)	29	(8)	29	(9)
jar	23	(13)	23	(13)	24	(12)	24	(12)	24	(12)	24	(12)	30	(15)	30	(15)	30	(15)	30	(15)
doc	21	(3)	27	(3)	30	(3)	29	(2)	30	(2)	32	(2)	34	(2)	34	(2)	35	(2)	36	(2)
pdf	19	(6)	22	(6)	24	(7)	25	(6)	23	(7)	21	(7)	22	(6)	23	(6)	23	(6)	23	(5)
xbm	18	(10)	19	(8)	17	(11)	18	(11)	18	(11)	18	(11)	18	(11)	18	(11)	18	(11)	18	(11)
png	18	(3)	24	(3)	14	(3)	4	(0)	6	(1)	10	(1)	10	(2)	8	(2)	9	(1)	9	(2)
swf	14	(8)	10	(6)	9	(6)	10	(7)	10	(7)	9	(5)	10	(5)	10	(6)	10	(6)	10	(6)
ppt	10	(2)	12	(5)	10	(3)	11	(3)	10	(4)	11	(4)	9	(3)	10	(4)	9	(3)	10	(5)
gz	7	(2)	15	(4)	18	(4)	13	(0)	13	(3)	19	(2)	24	(3)	24	(4)	24	(1)	22	(3)
gif	7	(3)	12	(3)	13	(4)	11	(5)	9	(3)	11	(3)	13	(1)	15	(2)	13	(1)	13	(1)
pps	6	(2)	3	(1)	4	(2)	3	(2)	3	(2)	2	(2)	4	(2)	2	(2)	4	(2)	3	(2)
jpg	6	(2)	10	(2)	14	(3)	13	(2)	9	(1)	6	(1)	7	(2)	9	(2)	10	(3)	9	(2)
ps	5	(3)	7	(5)	8	(5)	6	(4)	5	(4)	4	(3)	2	(2)	8	(0)	7	(0)	5	(0)
pptx	5	(4)	5	(4)	5	(4)	4	(3)	4	(3)	5	(4)	5	(3)	5	(3)	5	(4)	6	(5)

Conclusion

- Even though the method is very straightforward and simple, simplistic even and easily implemented we still manage to correctly classify quite a few file types with a worthwhile degree of accuracy
  - Given a fairly conservative setting, n-valued detection problem etc.
- However, even though the method is difficult to compare to other research in the area, others report better figures; so there's room for improvement
- Our data is available on request

Blekinge Institute of Technology
SE-371 79 Karlskrona
+46 455 38 50 00
www.bth.se/eng