UNIVERSITÀ DEGLI STUDI DI TRENTO

Department of Engineering and Information Science

Master in Computer Science

Research Project in Multimedia Data Security

Classification of Sharing Applications

January 28, 2020

Supervisors:
Prof. Giulia Boato
PHD. Quoc Tin Phan

Student:
Kritjan Gjika

Academic Year 2018/2019

Contents

1 Single Scenario Classification, KFold Validation

Starting with fitting randomly the classifiers, there are some statistics of the data used for the first test:

	count train	count test
messenger	249	100
telegram	244	106
whatsapp	243	107
original	243	107

1.1 Logistic regression results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	100	0	0	0
telegram	0	106	0	0
whatsapp	0	0	103	4
original	0	0	0	107

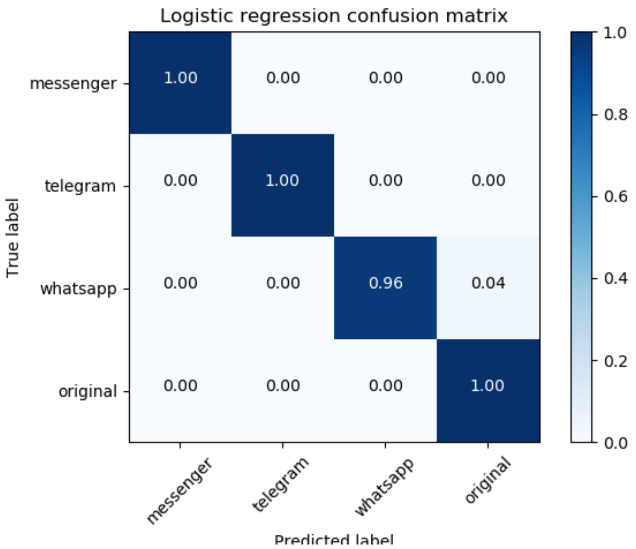


Figure 1.1: logistic regression

Result of the KFold validation with 10 bins:

0.9796	0.9898	1.0000	1.0000	1.0000	0.9898	0.9898	1.0000	0.9898	1.0000

The mean is : 0.993878

1.2 Linear Support Vector Machine results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	100	0	0	0
telegram	0	106	0	0
whatsapp	0	0	103	4
original	0	0	0	107

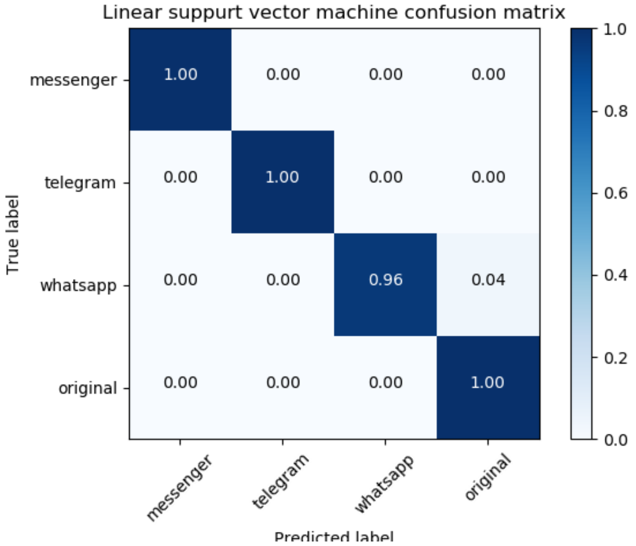


Figure 1.2: linear SVM

Result of the KFold validation with 10 bins:

0.9898	0.9898	1.0000	1.0000	1.0000	0.9796	0.9898	1.0000	0.9898	1.0000

The mean is : 0.993878

1.3 Random forest results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	100	0	0	0
telegram	0	106	0	0
whatsapp	0	0	103	4
original	0	0	0	107

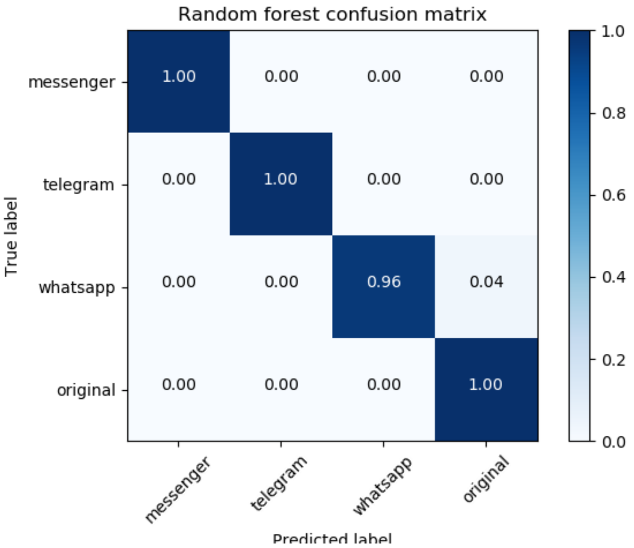


Figure 1.3: random forest

Result of the KFold validation with 10 bins:

1.0000	0.9898	1.0000	1.0000	1.0000	0.9796	0.9898	1.0000	0.9898	0.9897

The mean is : 0.993867

2 Single Scenario Classification, Circularly Validation

Here was used the same dataset as before but the training used a 0.3 of the dataset, and it is shifted circulary to cover all the dataset.Here is the table of all steps calculated

step	logistic	linear SVM	random fo.
0	0.989179818268771	0.9861800141743444	0.9901213241201993
1	0.9848642886352729	0.989179818268771	0.9901086744322249
2	0.9859958494970797	0.989179818268771	0.992039295392954
3	0.9840239043824701	0.989179818268771	0.9879411617983916
4	0.9838346881813623	0.989179818268771	0.9878321961314647
5	0.9850775862706985	0.9888435941925785	0.9929795918367347
6	0.9849757394589151	0.9899004065040651	0.989900983984118
7	0.9869068386254386	0.9899004065040651	0.99103468547913
8	0.9859149679167976	0.9899004065040651	0.990969961616947
9	0.9826572092251535	0.9861966137690223	0.9817251354156551
10	0.9847290722474007	0.9870545842786601	0.9827326856656295
11	0.9837185571115683	0.9857771334299855	0.9846840787373938
12	0.9838354913678619	0.9859658778205833	0.9846840787373938
13	0.9813181579293129	0.9861166500498505	0.9807610095111248
14	0.9813181579293129	0.9853611471537412	0.9787703014260097
15	0.9831900496861925	0.9862857095347368	0.980725773647614
16	0.9841168266469769	0.9862857095347368	0.9808059618649602

17	0.9822572998070824	0.9822572998070824	0.9760144649257553
18	0.9821251322105606	0.9822572998070824	0.97795683313976
19	0.9820101172758178	0.982107843137255	0.9750549818320903
20	0.9820101172758178	0.9826435137223949	0.9769817171132961
21	0.9820101172758178	0.9822440033492588	0.9769817171132961
22	0.9820101172758178	0.9819674282059272	0.9769817171132961
23	0.9789859263543474	0.9826435137223949	0.9734258819806992
24	0.9789859263543474	0.9844528594528594	0.9734258819806992
25	0.9790240688968155	0.9835470085470086	0.9734258819806992
26	0.978963179539905	0.9808615772912023	0.9734258819806992
27	0.981011696187139	0.9881608339538348	0.981094861660079
28	0.9809466587092924	0.9880438882784184	0.981094861660079
29	0.978957428886153	0.9869016393442622	0.981094861660079
30	0.9771308523409363	0.9880438882784184	0.981094861660079
31	0.9839638554216867	0.9889326989562411	0.981094861660079
32	0.9821736011477762	0.9839576074332173	0.974576923076923
33	0.9632234670976825	0.963381121890158	0.9618357875948238
34	0.955915762290795	0.9604524917457968	0.9523383383383384
35	0.9558080031175651	0.9615025224051383	0.9332107165025093
36	0.9537713472485769	0.9616828738173668	0.9342712270274949
37	0.9567246849068246	0.9705229237156167	0.941807112194959
38	0.9624805441127516	0.9689582071471836	0.941807112194959
39	0.9656916766799837	0.9754108565737052	0.9426760297719203
40	0.9645393196105017	0.9744245524296675	0.9426760297719203
41	0.9674626293689195	0.9725627105089125	0.9426760297719203
42	0.9654192933722927	0.970744883788362	0.9435515300577979
43	0.9695591349062311	0.9723367392625123	0.9638905905957089
44	0.9684887580521552	0.9724221573471613	0.9735226067675696
45	0.968972132612202	0.9742295202245372	0.9713033424446343
46	0.9682197824252712	0.9742295202245372	0.9735226067675696
47	0.9693788613812181	0.9731363489522036	0.9724089271961905
48	0.9668187320808225	0.9683336860555347	0.9713033424446343
49	0.9642240738507779	0.9642997792344016	0.9631028529724224
50	0.9629520363275152	0.9642997792344016	0.9641429955913738
51	0.9631771897864273	0.9642997792344016	0.9609153080205712
52	0.9643385011275081	0.9642997792344016	0.9651904231493449
53	0.9738195798137318	0.9726644779063561	0.9702353383569476
54	0.9782388663967612	0.9752631578947368	0.9778754788737738
55	0.9782388663967612	0.9695209703947368	0.9778754788737738
56	0.9782388663967612	0.9713281539030707	0.9800443458980044
57	0.9789586940956656	0.9714048901782014	0.9789560728306903
58	0.9808488835137682	0.9750631313131313	0.980188679245283
59	0.9809913155949741	0.9854702263238849	0.9790973762010348
60	0.9820075757575757	0.9808654423423285	0.9800443458980044
61	0.9820075757575757	0.9820075757575757	0.9800443458980044
62	0.9820075757575757	0.9820075757575757	0.9780137313157126
63	1.0	1.0	0.9989837398373984
64	0.9949551291586097	0.9939669421487604	0.9858662941153005
65	0.9901960784313726	0.9892578125	0.9831053292616855
66	0.9901960784313726	0.9892578125	0.9898912530352956
67	0.9901960784313726	0.9892578125	0.9920351473922903

68	0.9881717869333969	0.9852417482429718	0.986020872302839
69	0.9881717869333969	0.9861800141743444	0.9910744534968137

Average of all steps:

logistic r.	linear SVM	random f.
0.9777519138070937	0.9801399772382293	0.9746721183192149

Confusion matrix estimated on overall tests:

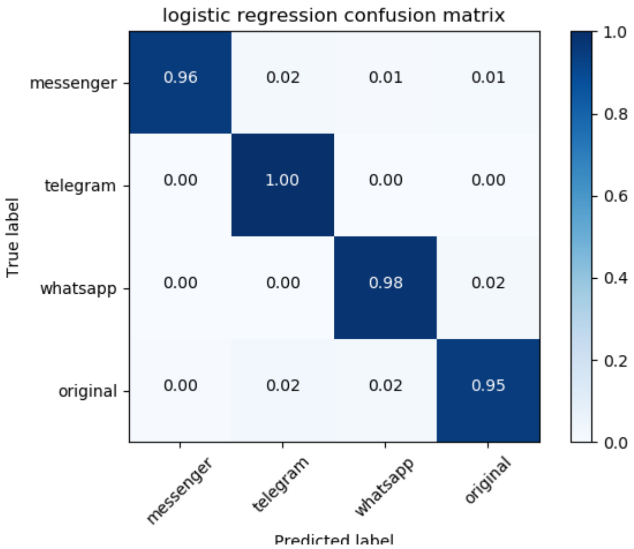


Figure 2.1: logistic regression

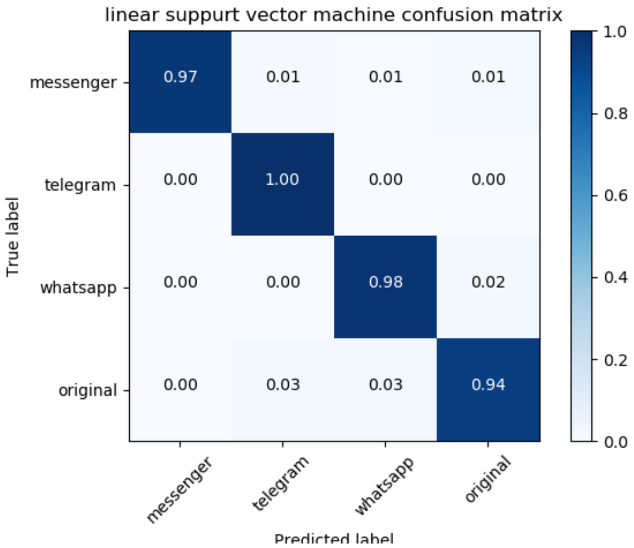


Figure 2.2: linear SVM

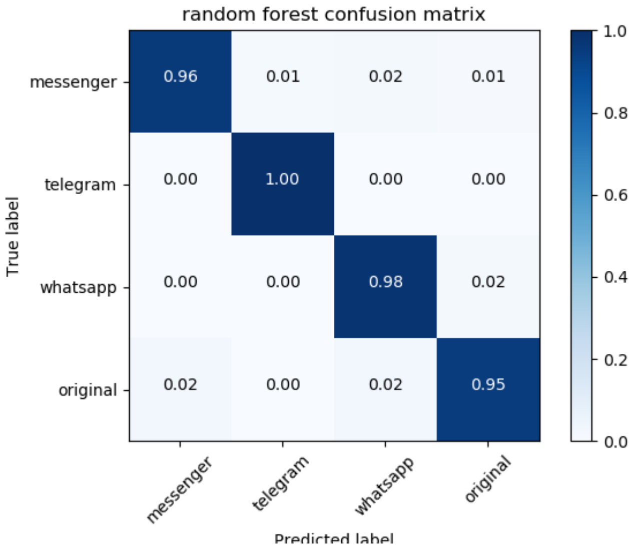


Figure 2.3: random forest

3 Double Scenario Classification of the last shared app, KFold Validation

Starting with fitting randomly the classifiers, there are some statistics of the data used for the first test:

	count train	count test
messenger	302	748
telegram	314	736
whatsapp	340	710
original	94	256

3.1 Logistic regression results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	746	0	2	0
telegram	0	618	118	0
whatsapp	0	216	494	0
original	0	0	8	248

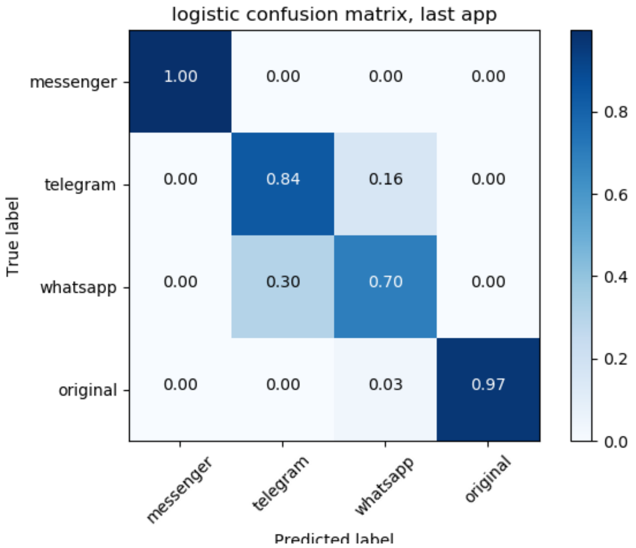


Figure 3.1: logistic regression, last app classified

Result of the KFold validation with 10 bins:

0.8476	0.8000	0.9143	0.8667	0.8286	0.8762	0.8381	0.8190	0.8476	0.8571

The mean is : 0.849524

3.2 Linear Support Vector Machine results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	730	6	12	0
telegram	0	535	201	0
whatsapp	1	197	511	1
original	0	0	6	250

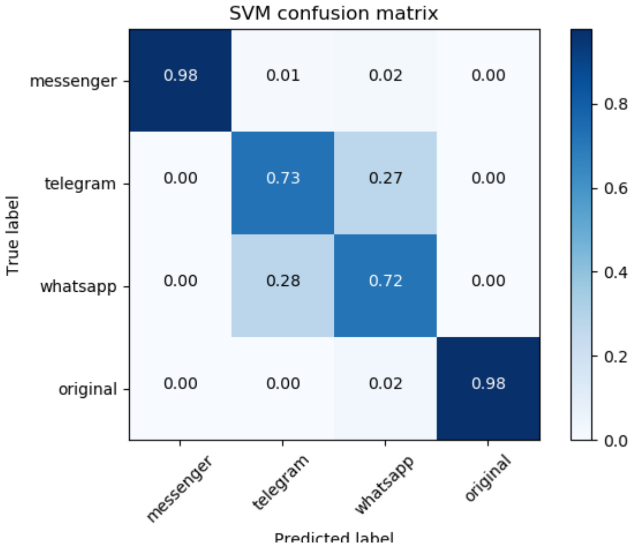


Figure 3.2: linear SVM, last app classified

Result of the KFold validation with 10 bins:

0.8476	0.8095	0.8667	0.7905	0.7810	0.8857	0.8000	0.8000	0.8000	0.8571

The mean is : 0.823810

3.3 Random forest results:

Confusion matrix with number of sample and with normalization:

	messenger	telegram	whatsapp	original
messenger	740	0	8	0
telegram	0	627	109	0
whatsapp	1	242	463	4
original	0	0	2	254

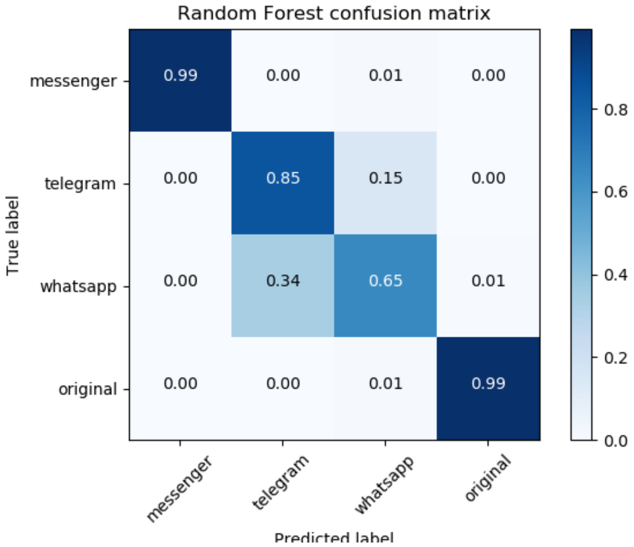


Figure 3.3: random forest, last app classified

Result of the KFold validation with 10 bins:

0.8381	0.8381	0.8857	0.9048	0.8571	0.8952	0.8571	0.8762	0.8381	0.8286

The mean is : 0.861905

4 Double Scenario Classification of the first and last shared app, KFold Validation

Starting with fitting randomly the classifiers, there are some statistics of the data used for the first test:

	count train	count test
mess_mess	96	254
tele_mess	99	251
what_mess	107	243
mess_tele	98	252
tele_tele	111	239
what_tele	105	245
mess_what	116	234
tele_what	103	247
what_what	121	229
original	94	256

4.1 Logistic regression results:

Confusion matrix with number of sample and with normalization:

	m_m	m_t	m_w	t_m	t_t	t_w	w_m	w_t	w_w	original
mess_mess	248	5	1	0	0	0	0	0	0	0
tele_mess	2	233	14	0	0	0	2	0	0	0
what_mess	13	26	202	0	0	2	0	0	0	0
mess_tele	0	0	0	65	116	3	0	68	0	0
tele_tele	0	0	0	66	57	2	0	114	0	0
what_tele	0	0	0	3	4	235	1	2	0	0
mess_what	0	0	0	1	0	1	80	0	152	0
tele_what	0	0	0	65	129	3	0	50	0	0
what_what	0	0	0	0	0	0	104	0	123	2
original	0	0	0	0	0	0	0	0	5	251

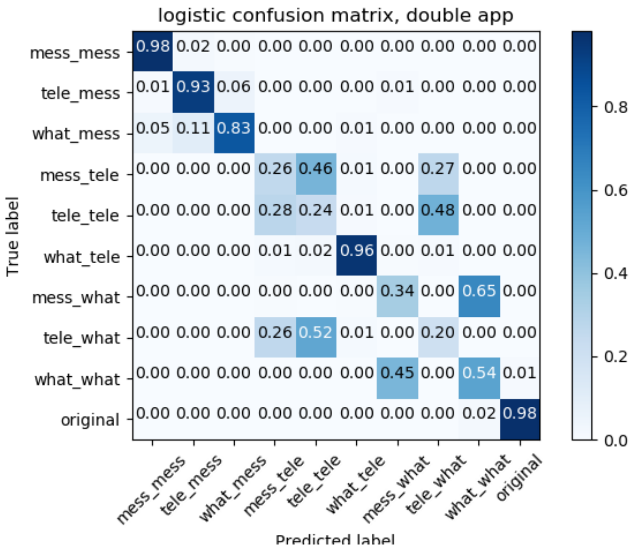


Figure 4.1: logistic regression, last app classified

Result of the KFold validation with 10 bins:

0.6000	0.5619	0.6381	0.6095	0.6190	0.6667	0.6000	0.5810	0.5429	0.6190

The mean is : 0.603810

4.2 Linear Support Vector Machine results:

Confusion matrix with number of sample and with normalization:

	m_m	m_t	m_w	t_m	t_t	t_w	w_m	w_t	w_w	original
mess_mess	246	4	2	0	2	0	0	0	0	0
tele_mess	1	213	17	2	2	0	12	1	3	0
what_mess	14	16	194	3	0	1	7	5	3	0
mess_tele	0	0	0	65	105	5	0	76	1	0
tele_tele	0	1	0	61	52	2	0	123	0	0
what_tele	0	0	0	3	5	232	1	4	0	0
mess_what	0	0	0	2	2	0	78	0	152	0
tele_what	0	1	0	58	137	3	0	48	0	0
what_what	0	0	0	1	1	0	96	1	130	0
original	0	0	0	2	1	0	0	0	5	248

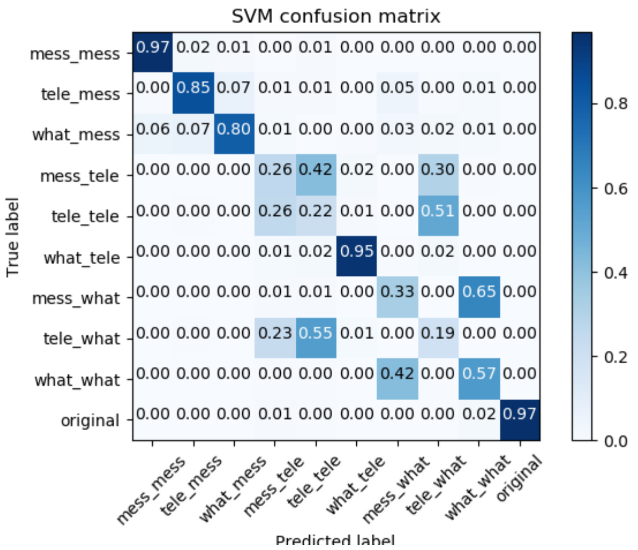


Figure 4.2: linear SVM, last app classified

Result of the KFold validation with 10 bins:

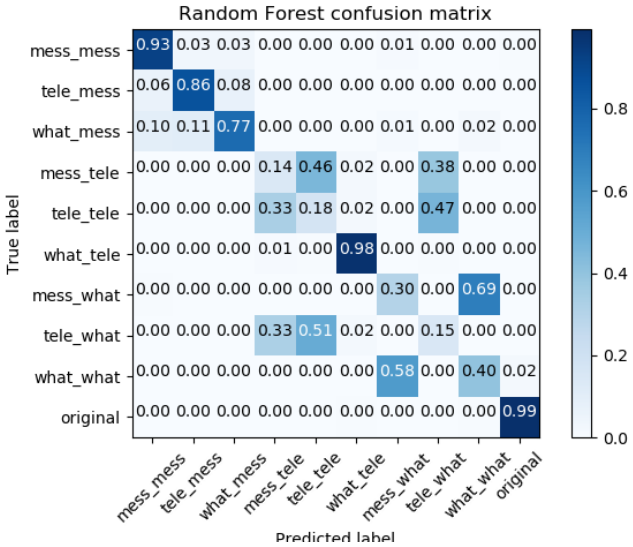
0.6095	0.5619	0.6000	0.5905	0.6000	0.6952	0.6476	0.5905	0.5524	0.6381

The mean is : 0.608571

4.3 Random forest results:

Confusion matrix with number of sample and with normalization:

	m_m	m_t	m_w	t_m	t_t	t_w	w_m	w_t	w_w	original
mess_mess	235	8	8	0	0	0	2	0	0	1
tele_mess	16	216	19	0	0	0	0	0	0	0
what_mess	24	26	186	0	0	0	3	0	4	0
mess_tele	0	0	0	36	115	4	0	97	0	0
tele_tele	0	0	0	80	42	5	0	112	0	0
what_tele	0	0	0	2	1	241	0	1	0	0
mess_what	1	0	0	0	0	0	71	0	162	0
tele_what	0	0	0	81	125	4	0	37	0	0
what_what	0	0	0	0	0	0	133	0	92	4
original	0	0	0	0	0	1	0	0	1	254



	mess_mess	tele_mess	what_mess	mess_tele	tele_tele	what_tele	mess_what	tele_what	what_what	original
mess_mess	0.93	0.03	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.00
tele_mess	0.06	0.86	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00
what_mess	0.10	0.11	0.77	0.00	0.00	0.00	0.01	0.00	0.02	0.00
mess_tele	0.00	0.00	0.00	0.14	0.46	0.02	0.00	0.38	0.00	0.00
tele_tele	0.00	0.00	0.00	0.33	0.18	0.02	0.00	0.47	0.00	0.00
what_tele	0.00	0.00	0.00	0.01	0.00	0.98	0.00	0.00	0.00	0.00
mess_what	0.00	0.00	0.00	0.00	0.00	0.00	0.30	0.00	0.69	0.00
tele_what	0.00	0.00	0.00	0.33	0.51	0.02	0.00	0.15	0.00	0.00
what_what	0.00	0.00	0.00	0.00	0.00	0.00	0.58	0.00	0.40	0.02
original	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99

Figure 4.3: random forest, last app classified

Result of the KFold validation with 10 bins:

0.5333	0.5048	0.6190	0.5429	0.5619	0.6000	0.5714	0.6286	0.4952	0.5619

The mean is : 0.561905

5 Single and double scenario, KFold Validation

Starting with fitting randomly the classifiers, there are some statistics of the data used for the first test:

	count train	count test
mess	97	253
tele	335	715
what	219	481
mess_mess	99	251
tele_mess	113	237
what_mess	93	257
mess_tele	89	261
what_tele	103	247
mess_what	106	244
original	111	239

5.1 Logistic regression results:

Confusion matrix with number of sample and with normalization:

	m	t	w	m_m	m_t	m_w	t_m	t_w	w_m	original
mess	184	0	0	46	4	19	0	0	0	0
tele	0	710	0	0	0	0	0	5	0	0
what	0	0	419	0	0	0	0	0	58	4
mess_mess	21	0	0	220	6	4	0	0	0	0
tele_mess	0	0	0	0	215	22	0	0	0	0
what_mess	0	0	0	0	88	169	0	0	0	0
mess_tele	0	256	0	0	0	0	0	5	0	0
what_tele	0	9	0	0	0	0	2	236	0	0
mess_what	0	0	191	0	0	0	0	1	52	0
original	0	0	0	0	0	0	0	0	0	239

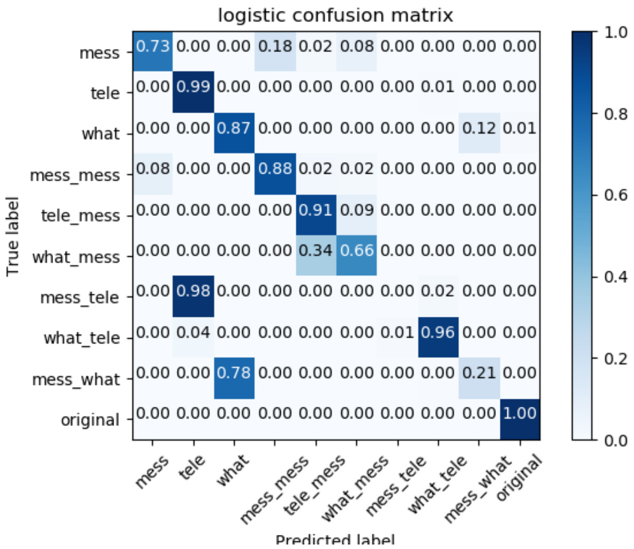


Figure 5.1: logistic regression, last app classified

Result of the KFold validation with 10 bins:

0.8029	0.7810	0.7810	0.7737	0.7372	0.7353	0.7059	0.8162	0.7721	0.7794

The mean is : 0.768474

5.2 Linear Support Vector Machine results:

Confusion matrix with number of sample and with normalization:

	m	t	w	m_m	m_t	m_w	t_m	t_w	w_m	original
mess	192	0	0	36	5	20	0	0	0	0
tele	0	715	0	0	0	0	0	0	0	0
what	0	0	409	0	0	0	0	0	68	4
mess_mess	39	0	0	204	6	2	0	0	0	0
tele_mess	0	0	0	0	221	16	0	0	0	0
what_mess	0	0	0	0	66	191	0	0	0	0
mess_tele	0	261	0	0	0	0	0	0	0	0
what_tele	0	5	0	0	0	0	0	242	0	0
mess_what	0	0	174	0	0	0	0	1	69	0
original	0	0	0	0	0	0	0	0	0	239

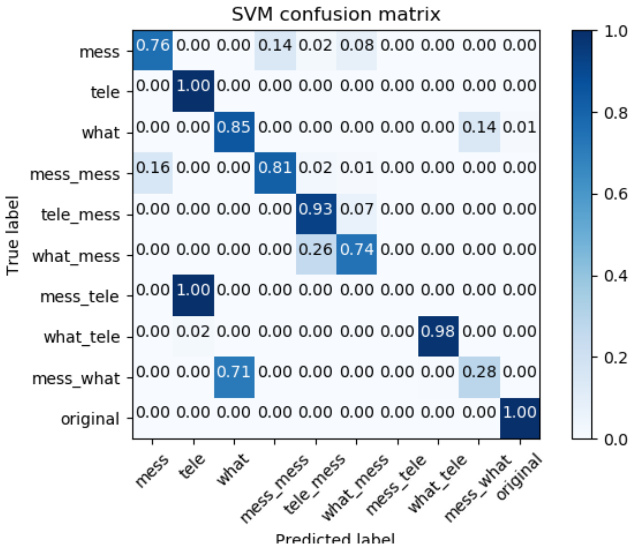


Figure 5.2: linear SVM, last app classified

Result of the KFold validation with 10 bins:

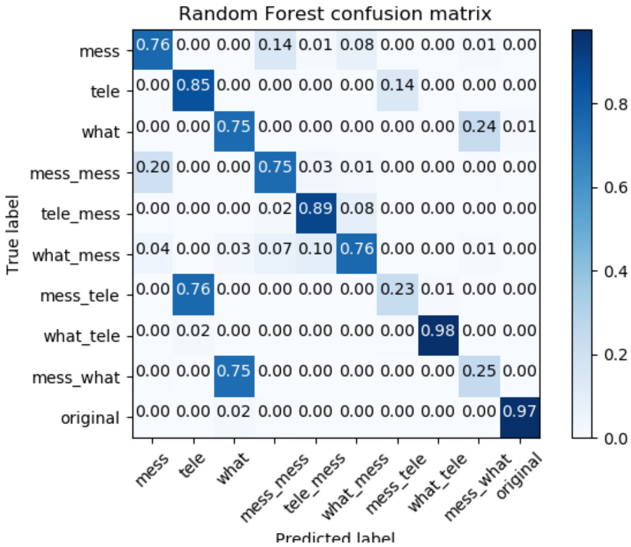
0.8102	0.7956	0.7737	0.8175	0.7664	0.8015	0.7426	0.8015	0.8162	0.7941

The mean is : 0.791939

5.3 Random forest results:

Confusion matrix with number of sample and with normalization:

	m	t	w	m_m	m_t	m_w	t_m	t_w	w_m	original
mess	192	0	0	36	3	19	0	0	3	0
tele	0	609	0	0	0	0	103	3	0	0
what	0	0	362	0	0	0	0	0	115	4
mess_mess	50	0	0	189	8	3	0	0	1	0
tele_mess	1	0	0	5	211	20	0	0	0	0
what_mess	10	0	7	17	25	196	0	0	2	0
mess_tele	0	199	0	0	0	0	60	2	0	0
what_tele	0	5	0	0	0	0	1	241	0	0
mess_what	1	0	182	0	0	0	0	0	61	0
original	0	0	5	1	0	0	0	0	0	233



	mess	tele	what	mess_mess	tele_mess	what_mess	mess_tele	what_tele	mess_what	original
mess	0.76	0.00	0.00	0.14	0.01	0.08	0.00	0.00	0.01	0.00
tele	0.00	0.85	0.00	0.00	0.00	0.00	0.14	0.00	0.00	0.00
what	0.00	0.00	0.75	0.00	0.00	0.00	0.00	0.00	0.24	0.01
mess_mess	0.20	0.00	0.00	0.75	0.03	0.01	0.00	0.00	0.00	0.00
tele_mess	0.00	0.00	0.00	0.02	0.89	0.08	0.00	0.00	0.00	0.00
what_mess	0.04	0.00	0.03	0.07	0.10	0.76	0.00	0.00	0.01	0.00
mess_tele	0.00	0.76	0.00	0.00	0.00	0.00	0.23	0.01	0.00	0.00
what_tele	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00
mess_what	0.00	0.00	0.75	0.00	0.00	0.00	0.00	0.00	0.25	0.00
original	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.97

Figure 5.3: random forest, last app classified

Result of the KFold validation with 10 bins:

0.7591	0.7664	0.7007	0.7445	0.6496	0.8309	0.7206	0.7647	0.6618	0.7574

The mean is : 0.735573

Bibliography