Ciphered text classification

Пичужкина Ольга,
НИУ ВШЭ, 3 курс

Objective

Multiclass classification on ciphered texts.

Source: https://www.kaggle.com/c/20-newsgroups-ciphertext-challenge

Data

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Newsgroups used in the dataset

20 Newsgroups dataset – a dataset often used as a sample set for multiclass classification and NLP. 39052 news texts partitioned across 20 newsgroups.

For Kaggle competition each text was encrypted with up to 4 simple ciphers.

Each encrypted text has a difficulty level from 1 to 4.

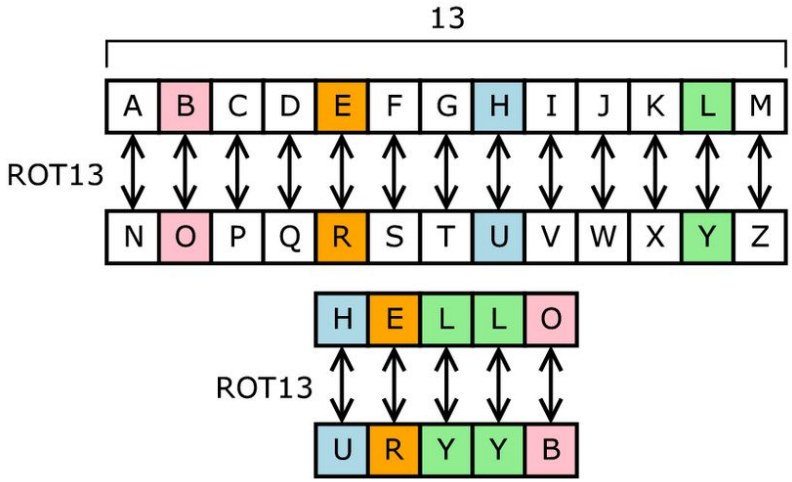
A difficulty level *n* means the text was encrypted consecutively with ciphers *1,...,n*.

Split into 0.75 train and 0.25 test set.

Data

	Id	difficulty	ciphertext	target
0	ID_88b9bbd73	4	obIIK?zzhX*L{83B3Z,FuL*Pusm$83L\t@r$$*38,8s...	10
1	ID_f489bd59f	1	c1IFaAO120O'8ovfoy1W#atvGs1[1s1[1/1]O-a8o1-...	13
2	ID_f90fee9c7	2	1*e4N8f0ccOuihkHek$k*V*hoeV$Hj8VhH8...	19
3	ID_8303ced65	1	O8v^10O#to1'#^'^tv1^]s111t01Otaq>-ata_1...	17
4	ID_72abc2cb7	2	eV}H}khfe4b8'S.Vc}{A.#VikV.fV?{$f7$Hjb8...	0

Data: ciphers 1 and 2



Looking for info about this dataset on Kaggle,
I found out a user named Flal
already cracked ciphers 1 and 2!

- Symbol frequencies distribution in texts encrypted with cipher 1 indicates cipher 1 is simple **substitution cipher**.
- Cipher 1 can be manually cracked by mapping symbol frequencies.
- The same works for cipher 2.
- Cipher 3 didn’t appear to be substitution cipher.
- Therefore, cipher 4 is also not decipherable manually.

Manual deciphering

	Id	difficulty	ciphertext	target
0	ID_88b9bbd73	4	oblIK?zzhX*L{83B3Z,FuL*Pusm$83L\t@r$$*38,8s...	10
1	ID_f489bd59f	1	u (Lida Chaplynsky) writes:\n > \n > A family ...	13
2	ID_f90fee9c7	2	From: ece_0028@bigdog.engr.arizona.edu (David ...	19
3	ID_8303ced65	1	also hearty proponents of\n the anti-Semitic...	17
4	ID_72abc2cb7	2	o.asd.sgi.com> <1pa0stINNpqa@gap.caltech.edu> ...	0

The data after deciphering texts encrypted with ciphers 1 and 2.

Vectorization

Texts were vectorized by characters inside word boundaries.

N-grams (*n* from 1 to 6) were extracted.

Count Vectorizer	Tf-Idf Vectorizer
Converts each text in a dataset into a matrix of token counts.	Converts each text in a dataset into tf-idf matrix.
An issue: our dataset is a mix of encrypted and plain texts. Since encrypted and plain texts have different symbols frequency distribution, character count in all dataset probably isn't a very significant measure.	Since tf-idf measures how important a symbol is to this specific document, it's a better measure.
Indeed: a classifier trained on texts vectorized with CountVectorizer was training VERY slowly.	

Dimensionality reduction

Dimensionality reduction (TruncatedSVD) on text vectors of was shown to make our model perform **worse**, so it wasn’t used.

Accuracy on a test set (with TruncatedSVD): 0.1103144525248386 8

Macro F1 score on a test set (with TruncatedSVD): 0.02124415833546395

Classification

Linear Support Vector Classifier

(a classifier that performs best at the original plaintext 20 Newsgroups dataset, as shown by many analyses, for example, https://acardocacho.github.io/capstone/)

Accuracy on a test set: 0.535286284953395 5

Macro F1 score on a test set: 0.520113094999397 7

Cross-validation

Stratified KFold approach with 5 folds.

Cross-validation didn’t appear to improve scores. :(

```
Split: 0, train accuracy: 0.9882584005806755, train F1: 0.9876443094584729,
 test accuracy: 0.5071574642126789, test F1: 0.487302032548013 74.
Split: 1, train accuracy: 0.988346779357152, train F1: 0.9876398981650564,
 test accuracy: 0.5146707608324804, test F1: 0.4974644677288125.
Split: 2, train accuracy: 0.9886048397422218, train F1: 0.9880555547836576,
 test accuracy: 0.5131444178900648, test F1: 0.4913497607741813.
Split: 3, train accuracy: 0.988224251215974, train F1: 0.9874781383627107,
 test accuracy: 0.5026491198085797, test F1: 0.481021150775665 54.
Split: 4, train accuracy: 0.9878407781901958, train F1: 0.9872097871859193,
 test accuracy: 0.5116239316239316, test F1: 0.4898475545031541.
```

Accuracy on a test set: 0.5216634231281368

Macro F1 score on a test set: 0.5039209266390035

Different pipelines

Solution: train different pipelines on different (decrypted, encrypted with level 3, encrypted with level 4) texts. To predict a label for a text with difficulty *n*, use the corresponding pipeline.

To see, which performs better by itself, 4 pipelines were trained – on decrypted texts, texts, encrypted with level 3, texts, encrypted with level 4 and both texts encrypted with level 3 and level 4.

Each pipeline contains TfidfVectorizer and LinearSVC(). Decrypted texts were vectorized by word n-grams, encrypted – by character n-grams.

To evaluate performance, data for each pipeline was split into 0.75 train and 0.25 test set.

Different pipelines: performance by itself

Pipeline trained only on decrypted texts performed better than our first model, trained on all texts, while all pipelines trained on encrypted texts performed worse.

But how good will combined predictions of different pipelines be?

	Deciphered	**Difficulty 3**	**Difficulty 4**	**Difficulty 3+4**
Accuracy	0.6714926590538336	0.49324324324324326	0.22101885278780586	0.35390946502057613
Macro F1	0.6602408664878151	0.46180669278374487	0.14626217930595922	0.316334352059986

Different pipelines: final model

For the final model, only two pipelines were used.

- For decrypted texts: TfidfVectorizer (word n-grams) + LinearSVC()
- For encrypted texts: TfidfVectorizer (character n-grams) + LinearSVC()

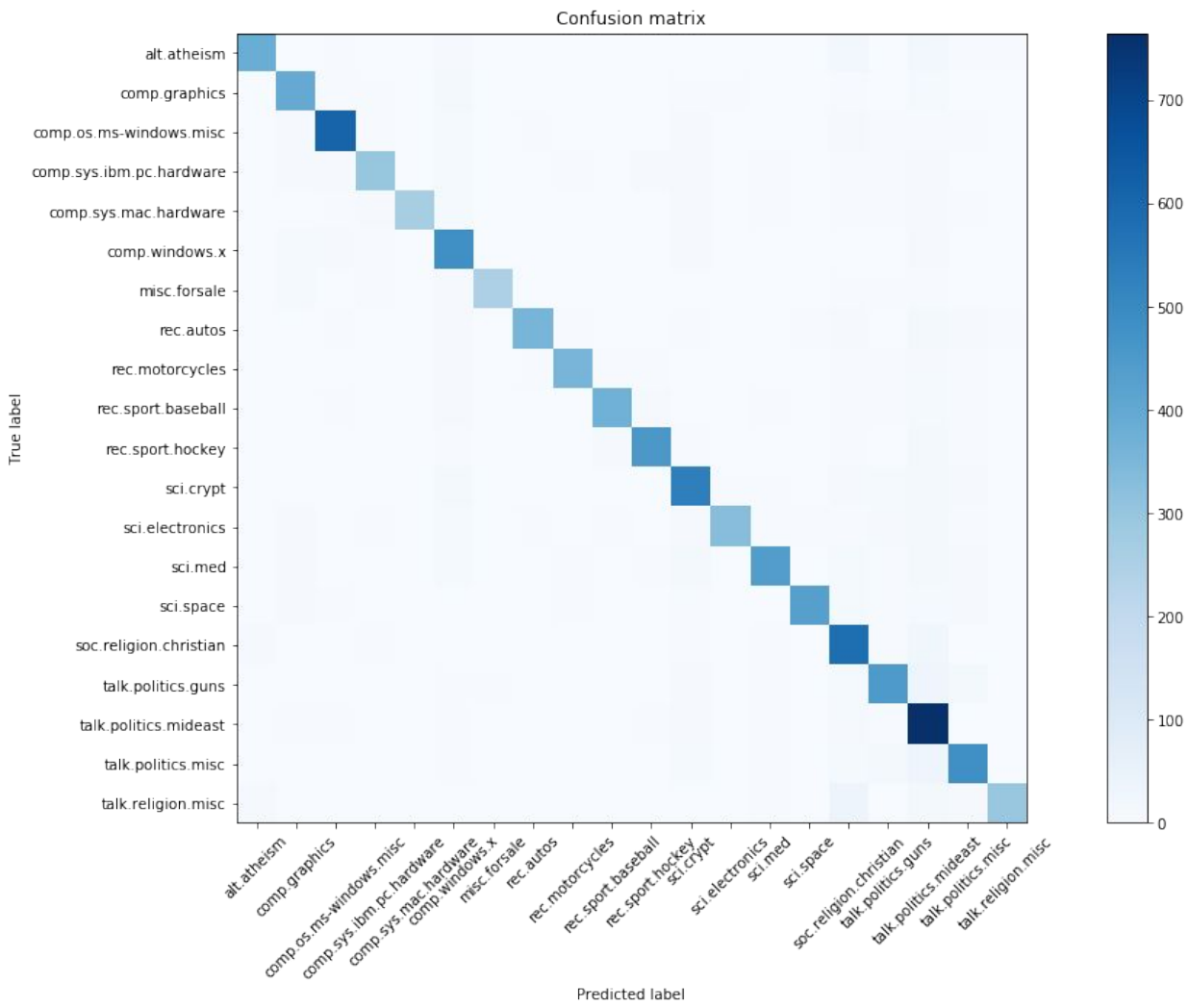
Despite the fact that a classifier trained only on decrypted texts performed poorly by itself, two classifiers combined gave significantly better predictions.

Accuracy on a test set: 0.8729898596742804

Macro F1 score on a test set: 0.8761930273095888

Classification report and confusion matrix

	precision	recall	f1-score	support
alt.atheism	0.93	0.86	0.89	449
comp.graphics	0.85	0.87	0.86	452
comp.os.ms-windows.misc	0.92	0.92	0.92	667
comp.sys.ibm.pc.hardware	0.89	0.83	0.86	364
comp.sys.mac.hardware	0.97	0.81	0.88	323
comp.windows.x	0.79	0.91	0.85	530
misc.forsale	0.94	0.85	0.89	293
rec.autos	0.94	0.85	0.89	421
rec.motorcycles	0.93	0.88	0.90	407
rec.sport.baseball	0.93	0.87	0.90	426
rec.sport.hockey	0.92	0.91	0.91	500
sci.crypt	0.84	0.89	0.86	599
sci.electronics	0.94	0.81	0.88	400
sci.med	0.90	0.84	0.87	520
sci.space	0.95	0.88	0.91	488
soc.religion.christian	0.79	0.91	0.84	638
talk.politics.guns	0.86	0.85	0.85	533
talk.politics.mideast	0.74	0.94	0.83	811
talk.politics.misc	0.86	0.85	0.85	568
talk.religion.misc	0.95	0.79	0.87	374
accuracy			0.87	9763
macro avg	0.89	0.87	0.88	9763
weighted avg	0.88	0.87	0.87	9763


Confusion matrix
alt.atheism
comp.graphics
comp.os.ms-windows.misc
comp.sys.ibm.pc.hardware
comp.sys.mac.hardware
comp.windows.x
misc.forsale
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
sci.crypt
sci.electronics
sci.med
sci.space
soc.religion.christian
talk.politics.guns
talk.politics.mideast
talk.politics.misc
talk.religion.misc
True label
Predicted label
700
600
500
400
300
200
100
0

Predictions on a Kaggle competition test set

Final evaluation of the model – competition test set (91122 encrypted texts).

	Id	difficulty	ciphertext
0	ID_65f17e60f	1	a 1t0 1OAAata^ O81 /e1'O_f1q1at;v1'# ttos1QQv8...
1	ID_ba83b2917	2	^V*$z8d*e4z8 8SF38e*8l8SFO8Hh}m}[8^7$ 8A84ej {...
2	ID_ff4e56b9c	3	$*c8?l{{?$8G iw{8{7$8Z9J8Edj?O$*8_n}{87U $8{7...
3	ID_81b64cbc5	2	nL6! 8lL 6M84VkVoh $ 88 7h}8h}88, 8z888888888...
4	ID_851f738e5	2	}8z8}{Vv$H8h 86*4$ hVWz8z8\%p8\Sp8lf9V*{7v[8\...

Since test set is huge, decrypting it all beforehand, like we did with train set, will be really inefficient. Instead, when needed function *decrypt()* was simply called on each text before making a prediction.

Macro F1 score on a Kaggle competition test set – 0.49459

Submission and Description	Public Score
submission.csv 8 minutes ago by Olga Pichuzhkina LinearSVC + TfIdf + two different pipelines for texts with difficulty 1 or 2 and texts with difficulty 3 or 4.	**0.49459**

Thank you for your attention!

The code is located here:

https://github.com/vyhuholl/ciphertext_multiclass_classification