CSE586.1 SPECIAL TOPICS (INFORMATION RETRIEVAL)

Assigment 2 Report

Text Classification using Naive Bayes

Feyza Özkefe, Işık University, İstanbul – Turkey

Multinomial Naive Bayes Text Classification is a text classifier with implementation of Multinomial N aive Bayes. Reuters 21578 data set is used for training.

Multinomial NB text classifier model assigns an input test document into classes, which are the five most common topics in the Reuters-21578 corpus:

- earn
- acq
- money-fx
- grain
- crude

Training and test sets consist of the news articles that belong to one of the above five topics. The news stories that belong to more than one of these topics are eliminated. The news stories that belong to only one of the above five topics, even if they belong to more than one topic is included. Stop words is removed.

After creating the training and test sets, the parameters of Multinomial Naive Bayes model using the training set with using all words in the lexicon are learned. For comparison, the program was first run with the library, then it was run without the library.

Frequency-based feature selection was used instead of mutual information for the most repeated word, and the 50 most repeated words were printed for each topic. Retraining the data set according to the most words failed.

Reports:

1. **Report the number of documents in each class in the training and test sets**

Number of documents in train dataset = 5783

Number of documents in test dataset = 2302

2. **Report the k most discriminating words (where k = 50) for each class based on Mutual Information.**

Most discriminating 50 words in 'acq' topic: [('stake', 432.0), ('have', 432.0), ('merger', 450.0), ('acquisition', 452.0), ('common', 504.0), ('also', 515.0), ('co', 558.0), ('had', 560.0), ('about', 598.0), ('he', 605.0), ('as', 653.0), ('or', 666.0), ('which', 696.0), ('group', 700.0), ('not', 761.0), ('was', 766.0), ('share', 789.0), ('stock', 797.0), ('would', 821.0), ('from', 835.0), ('at', 848.0), ('offer', 880.0), ('an', 909.0), ('corp', 1022.0), ('will', 1063.0), ('with', 1088.0), ('by', 1120.0), ('be', 1123.0), ('on', 1125.0), ('that', 1161.0), ('has', 1220.0), ('pct', 1276.0), ('inc', 1330.0), ('shares', 1363.0), ('is', 1376.0), ('company', 1393.0), ('reuter', 1455.0), ('3', 1495.0), ('mln', 1530.0), ('dlrs', 1876.0), ('its', 2012.0),

('for', 2214.0), ('it', 3202.0), ('in', 3712.0), ('and', 4284.0), ('a', 4457.0), ('said', 4896.0), ('to', 5471.0), ('of', 6571.0), ('the', 10152.0)]

Most discriminating 50 words in 'crude' topic: [('energy', 183.0), ('have', 189.0), ('about', 191.0), ('gas', 196.0), ('are', 197.0), ('which', 202.0), ('this', 203.0), ('not', 229.0), ('year', 239.0), ('but', 250.0), ('has', 250.0), ('production', 251.0), ('were', 251.0), ('last', 255.0), ('us', 264.0), ('opec', 277.0), ('barrels', 288.0), ('an', 289.0), ('3', 314.0), ('would', 314.0), ('reuter', 316.0), ('with', 322.0), ('bpd', 325.0), ('be', 335.0), ('as', 342.0), ('crude', 348.0), ('prices', 351.0), ('will', 375.0), ('pct', 388.0), ('he', 417.0), ('dlrs', 421.0), ('was', 431.0), ('at', 433.0), ('by', 441.0), ('is', 450.0), ('its', 458.0), ('it', 475.0), ('from', 498.0), ('on', 502.0), ('that', 565.0), ('for', 660.0), ('mln', 671.0), ('oil', 1246.0), ('a', 1399.0), ('and', 1445.0), ('said', 1495.0), ('in', 1668.0), ('of', 1868.0), ('to', 2136.0), ('the', 4084.0)]

Most discriminating 50 words in 'earn' topic: [('31', 564.0), ('stock', 577.0), ('at', 577.0), ('per', 592.0), ('prior', 593.0), ('dividend', 607.0), ('1985', 613.0), ('be', 642.0), ('shrs', 657.0), ('earnings', 657.0), ('avg', 662.0), ('is', 670.0), ('by', 674.0), ('record', 767.0), ('april', 773.0), ('quarter', 822.0), ('or', 835.0), ('will', 847.0), ('sales', 891.0), ('oper', 892.0), ('note', 925.0), ('pct', 973.0), ('company', 1094.0), ('on', 1104.0), ('billion', 1153.0), ('share', 1312.0), ('1986', 1363.0), ('its', 1379.0), ('it', 1517.0), ('revs', 1568.0), ('from', 1590.0), ('for', 1816.0), ('profit', 2091.0), ('year', 2110.0), ('shr', 2561.0), ('reuter', 2695.0), ('3', 2728.0), ('said', 2915.0), ('a', 3229.0), ('and', 3233.0), ('net', 3397.0), ('loss', 3455.0), ('to', 3628.0), ('in', 3840.0), ('dlrs', 4250.0), ('of', 5230.0), ('cts', 5424.0), ('the', 6625.0), ('mln', 7697.0), ('vs', 9143.0)]

Most discriminating 50 words in 'grain' topic: [('trade', 181.0), ('usda', 183.0), ('soviet', 187.0), ('they', 188.0), ('program', 194.0), ('department', 199.0), ('export', 206.0), ('but', 220.0), ('last', 227.0), ('were', 244.0), ('agriculture', 263.0), ('he', 264.0), ('as', 268.0), ('have', 272.0), ('are', 273.0), ('has', 284.0), ('an', 285.0), ('dlrs', 287.0), ('not', 289.0), ('year', 294.0), ('this', 302.0), ('was', 303.0), ('with', 308.0), ('would', 315.0), ('will', 322.0), ('pct', 338.0), ('by', 365.0), ('it', 381.0), ('corn', 386.0), ('reuter', 393.0), ('3', 400.0), ('on', 414.0), ('grain', 420.0), ('from', 440.0), ('at', 443.0), ('be', 470.0), ('that', 480.0), ('is', 493.0), ('us', 565.0), ('wheat', 654.0), ('mln', 733.0), ('tonnes', 823.0), ('for', 960.0), ('a', 1142.0), ('said', 1408.0), ('in', 1516.0), ('and', 1558.0), ('of', 2116.0), ('to', 2328.0), ('the', 4530.0)]

Most discriminating 50 words in 'money-fx' topic: [('have', 268.0), ('foreign', 273.0), ('they', 285.0), ('an', 287.0), ('money', 296.0), ('banks', 310.0), ('had', 311.0), ('central', 320.0), ('has', 323.0), ('yen', 326.0), ('currency', 330.0), ('will', 345.0), ('not', 355.0), ('would', 355.0), ('rates', 363.0), ('its', 366.0), ('rate', 367.0), ('this', 369.0), ('as', 395.0), ('but', 397.0), ('stg', 407.0), ('exchange', 419.0), ('be', 420.0), ('reuter', 427.0), ('billion', 432.0), ('from', 434.0), ('3', 442.0), ('mln', 470.0), ('with', 476.0), ('us', 489.0), ('was', 498.0), ('by', 512.0), ('market', 519.0), ('is', 552.0), ('he', 576.0), ('it', 600.0), ('dollar', 611.0), ('pct', 634.0), ('at', 698.0), ('on', 705.0), ('that', 716.0), ('for', 731.0), ('bank', 847.0), ('and', 1637.0), ('a', 1702.0), ('said', 1760.0), ('in', 2079.0), ('of', 2506.0), ('to', 2727.0), ('the', 6331.0)]

3. **Report the macro-averaged and micro-averaged precision, recall, and F-measure values obtained by your two classifiers on the test set, as well as the performance values obtained for each class separately by using Laplace smoothing with _ = 1.**

	precision	recall	f1-score	support
grain	0.79	0.99	0.88	717
earn	1.00	0.24	0.39	183
acq	0.93	0.98	0.96	1081
crude	1.00	0.57	0.73	147
money-fx	1.00	0.84	0.91	174
accuracy			0.89	2302
macro avg	0.95	0.72	0.77	2302
weighted avg	0.90	0.89	0.87	2302

4. **Include screenshots showing sample runs of your two programs.**

	topics	title	body	LEWIS
0	[cocoa]	BAHIA COCOA REVIEW	Showers continued throughout the week in\nthe ...	"TRAIN"
1	[None]	STANDARD OIL <SRD> TO FORM FINANCIAL UNIT	Standard Oil Co and BP North America\nInc said...	"TRAIN"
2	[None]	TEXAS COMMERCE BANCSHARES <TCB> FILES PLAN	Texas Commerce Bancshares Inc's Texas\nCommerc...	"TRAIN"
3	[None]	TALKING POINT/BANKAMERICA <BAC> EQUITY OFFER	BankAmerica Corp is not under\npressure to act...	"TRAIN"
4	[grain, wheat, corn, barley, oat, sorghum]	NATIONAL AVERAGE PRICES FOR FARMER-OWNED RESERVE	The U.S. Agriculture Department\nreported the ...	"TRAIN"

Fig1. All dataset

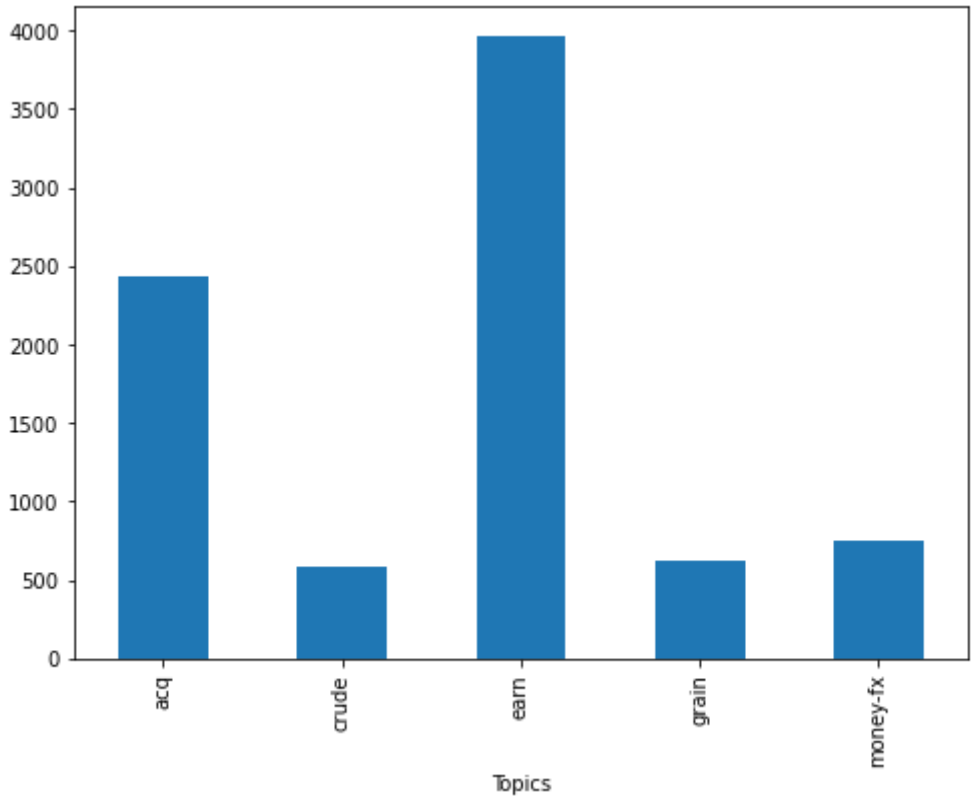

Fig2. Topic numbers in the dataset shown as graph

```
# Quick look at the topic numbers on the train set

print("Train data",df_new[df_new["LEWIS"] == '"TRAIN"'].groupby('Topics').Topics.count())
print("Test data",df_new[df_new["LEWIS"] == '"TEST"'].groupby('Topics').Topics.count())
```

```
Train data Topics
acq         1634
crude        354
earn        2859
grain        430
money-fx     506
Name: Topics, dtype: int64
Test data Topics
acq          717
crude        183
earn        1081
grain        147
money-fx     174
Name: Topics, dtype: int64
```

Fig3. Topic numbers in the train and the test set

```
print("Measuring values of 5 topics:")
if __name__ == '__main__':
    MNB = text_classifier()
    MNB.fit(X_train.values.tolist(), y_train_topic.values.tolist())
    pred = MNB.predict(X_test.values.tolist())
    true = y_test_topic.values.tolist()
    true_codes = y_test.values.tolist()

    accuracy = sum(1 for i in range(len(pred)) if pred[i] == true_codes[i]) / float(len(pred))
    print("Accuracy is: {0:.2f}".format(accuracy))
    print(metrics.classification_report(true_codes, pred, target_names=df_new['Topics'].unique()))
```

```
Measuring values of 5 topics:
Accuracy is: 0.89
              precision    recall  f1-score   support

       grain       0.79      0.99      0.88       717
        earn       1.00      0.24      0.39       183
         acq       0.93      0.98      0.96      1081
       crude       1.00      0.57      0.73       147
    money-fx       1.00      0.84      0.91       174

    accuracy                           0.89      2302
   macro avg       0.95      0.72      0.77      2302
weighted avg       0.90      0.89      0.87      2302
```

Fig4. Running text_classifier class to get results