CMPE493 PROJECT 1

I)

Program is two modules in single file. The program first checks whether the revers index is created using reuters data or not. If index is not created, then it creates the reverse index and then stores it in the file. If there is a file that contains reverse index data of retuters, then the program simply reads the reverse index file. It doesn't recreate reverse index.

Each document is read one by one. Then following operations are applied to each to retrieve "tittle", "body" and "newid".

1. To get the each story in file, document is splitted using "<REUTERS" as split point.
2. Then using "find()" method "NEWID=", "'", "<TITTLE>", "</TITTLE>", "<BODY>", "</BODY>" tags are found. After finding each interval using string operations each data extracted. Doc_id is added to documents list, other data is sent to preprocess() method.
3. Preprocess method returns the processed tokens.
4. It removes HTML entities and tags.
5. It removes punctuation characters.
6. It lowercases the string.
7. If the processed token is alphanumeric token, then it adds the token to result array.
8. Returns the result array.
9. Result array is matched with document_id.

II)

Dictionary is used in creating reverse index and storing documents and tokens.

Reverse index is dictionary list where first keys are composed of "tokens" and values are list of tuples. Tuple form is (doc_id, position).

Documents is a dictionary where keys are document id and values are the tokens of that document.

III)

If the reverse index is not created, then program runs like this:

```
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 --version
Python 3.10.10
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 solution.py
Creating inverted index...
Saving inverted index...
Done!
Enter query (type exit to exit):
```

If the reverse index is created and stored in a file in previous runs, then the program simply loads it. You can examine the process in the screenshot below.

```
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 --version
Python 3.10.10
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 solution.py
Creating inverted index...
Saving inverted index...
Done!
Enter query (type exit to exit):exit
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 solution.py
Loading inverted index...
Enter query (type exit to exit):
```

IV)

```
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 solution.py
Creating inverted index...
Saving inverted index...
Done!
Enter query (type exit to exit):"old crop cocoa"
Handling phrase query: "old crop cocoa"
[1]
Enter query (type exit to exit):old 1 cocoa
Handling proximity query: old 1 cocoa
[1, 19570]
Enter query (type exit to exit):exit
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 %
```

```
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 % python3 solution.py
Loading inverted index...
Enter query (type exit to exit):"reported after"
Handling phrase query: "reported after"
[1984, 8979, 12022, 16354, 20419]
Enter query (type exit to exit):reported 5 paying
Handling proximity query: reported 5 paying
[252, 358, 8979, 13258]
Enter query (type exit to exit):exit
(base) mustafa.cihan@Mustafa-MacBook-Air Project 1 %
```

- How many tokens does the corpus contain before and after case-folding?

  Before case-folding there are 2751260 tokens.
  After case-folding there are 2743698 tokens.

- How many terms (unique tokens) are there before and after case-folding?

  Before case-folding there are 130507 unique tokens.
  After case-folding there are 73998 unique tokens.

- List the top 100 most frequent terms after case-folding.

the: 144587
of: 73700
to: 72943
in: 55115
and: 54504
said: 53186
a: 51282
for: 26987
mln: 26810
it: 22492
dlrs: 21378
on: 19213
reuter: 19127
pct: 18104
is: 16915
that: 15474
its: 15432
from: 15334
by: 15111
vs: 14948

will: 14875
be: 14759
at: 14605
with: 13747
was: 11966
billion: 10697
year: 10656
he: 10640
us: 10416
has: 10204
as: 9688
an: 9559
cts: 9299
would: 9207
company: 8317
not: 8311
net: 7732
which: 7571
inc: 7368
bank: 7344
new: 7169
but: 7141
are: 7067
this: 6871
have: 6774
were: 6495
corp: 6375
up: 5963
last: 5951
market: 5801
had: 5700
stock: 5682
loss: 5637
or: 5491
1986: 5393
shares: 5254
one: 5239
also: 5187
about: 5128
they: 4979
share: 4776
trade: 4694
two: 4525

been: 4517
may: 4322
shr: 4271
oil: 4198
sales: 4121
debt: 3996
april: 3979
more: 3968
co: 3966
first: 3946
banks: 3712
after: 3662
march: 3559
exchange: 3528
government: 3526
than: 3425
prices: 3359
other: 3359
over: 3346
1987: 3285
profit: 3282
dlr: 3273
per: 3268
group: 3207
price: 3198
no: 3163
their: 3090
rate: 3035
international: 2995
interest: 2973
foreign: 2961
some: 2945
told: 2916
agreement: 2903
if: 2894
we: 2869
three: 2856