Report

Question: 1

Dataset : 20newsgroup

Dataset Description: This data set has 20k articles taken from 20 newsgroups.

One thousand articles were taken from each of the following 20 categories.

- 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

Total Size of Data : 43.9 MB

Pre-Processing Steps :

- All the articles were broken down into tokens using RegexpTokenizer.
- All the punctuations were removed.
- Porter steamer was used to perform stemming.
- All the stop words were left as it is.

Creating Inverted List:

- Inverted list was reverse sorted on the basis of static quality score.
- Each term in the inverted list consists of two list High and Low respectively on the basis of tf values.
- Both High and Low lists are reverse sorted by static quality score.
- Inverted List is implemented using Python dictionary.
- The structure of each term value pair is as follows:

  { Key : [doc id , static quality score , tf] }

Processing High and Low list for a query:

- Each query is broken down into stem of query terms.
- User is asked for the value of K (Total number of results he wants).
- Set union is performed on the High List of each query term.
- If the set size is greater than K then for this document set the net score is calculated as follows:
  - Net Score = Static quality score + cosine -score
- This document set is sorted on the basis of the net score and the resulting order of document is displayed with their respective Net score.
- If the set size is less than K then the low lists of the query terms are also considered and processed in the same way in which High lists were processed.

Handling Foreign query terms:

Try-Except is used to handle such exceptions.

Analysis for the size of High list ('r'):

Heuristic: Have used the average value of the total size of the postings for each term.

```
print("total length ",length)
print("total terms ",terms)
print("Average lenght",length/terms)
norm_len=length/(max)
print("normalised length",norm_len)
```

```
total length  3850470
total terms  186078
Average lenght 20.69277399800084
normalised length 235.7911818738518
```

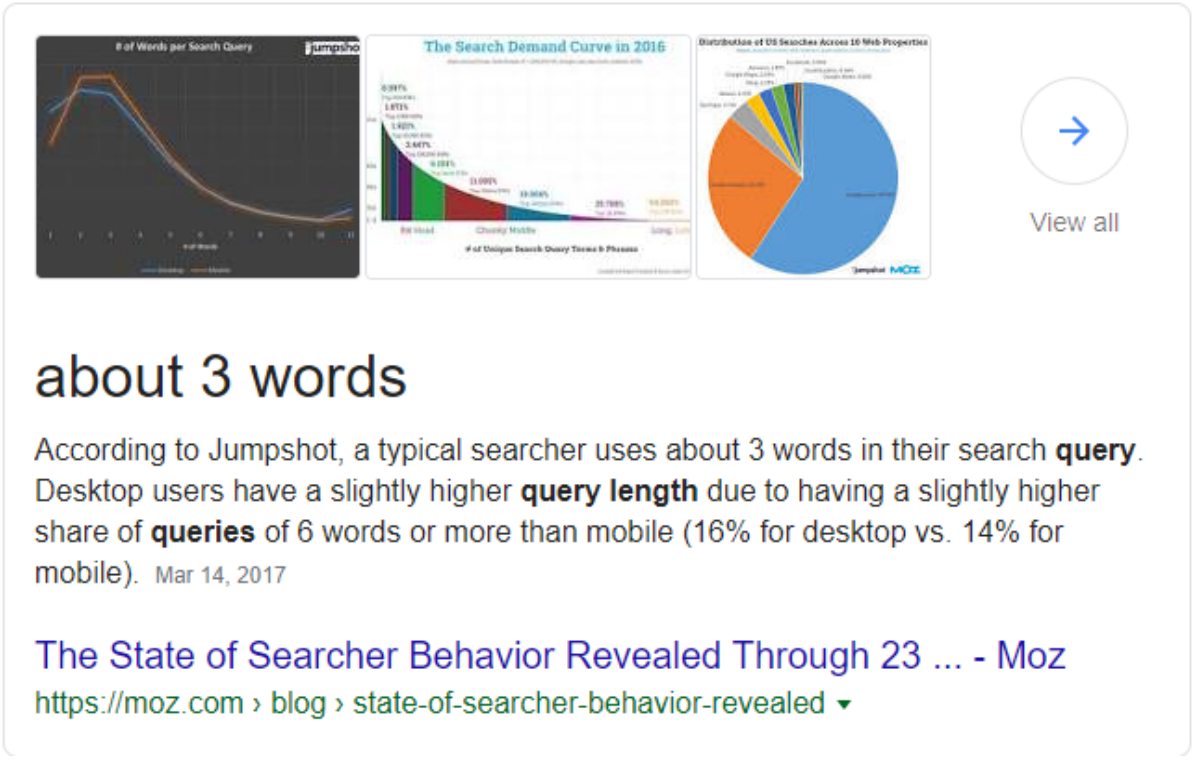
The value of size from the above heuristic is r = 20

- In order to support the correctness of the above heuristic following analysis is performed:
- Normalized average posting list is calculated using the below formula:
    - Normalized Query Length (L)= Average Query Length /Maximum Query Length

- All the terms of dictionary are divided into three categories on the basis of Normalized Query Length as follows:

Rare Terms (R)	(Posting List Length <= (0.25 * L))
Normal Terms (N)	((0.25 * L)) < Posting List Length <= (0.75* L))
Frequent Terms (F)	(Posting List Length > (0.75 * L))

- Have used the standard Average query length for the queries on google search engine as the standard query length value for our analysis.



Now if the query length is three then we can divided the queries into following combinations of Rare terms, Normal Terms and Frequent Terms as follows:

Number of Rare Terms	Number of Frequent Terms	Number of Normal Terms
3	0	0
0	3	0
0	0	3
0	1	2
0	2	1
1	0	2
1	2	0
2	0	1
2	1	0
1	1	1

Now Number of hits to the Low list is counted for each of the above category for the value of ‘r’ varing from [15, 25].

```
print(L_H)
```

```
[4, 5, 2, 3, 3, 3, 4, 3]
```

Question 2 :

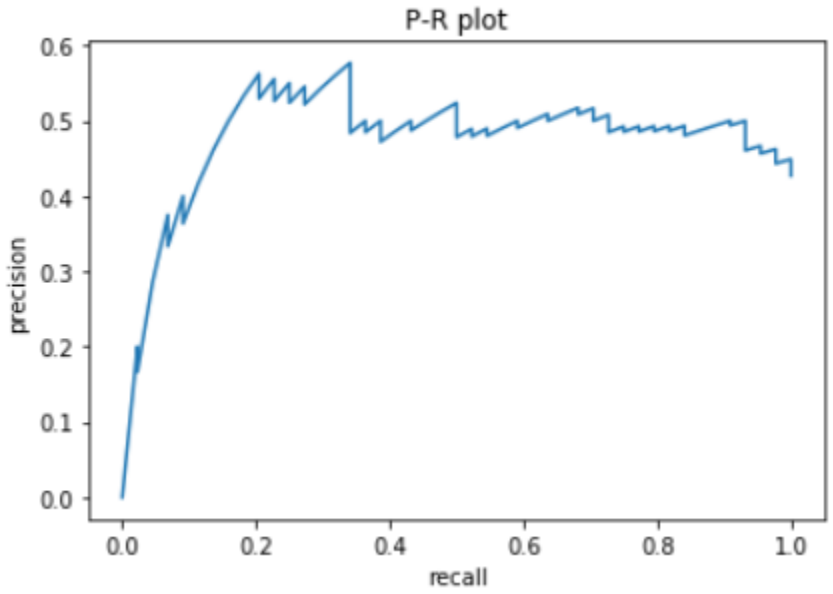
Dataset Used : https://drive.google.com/file/d/1aG_sOmDqN2cIx0ChUdxfGjdSVZAt7LGA/view

Dataset Size : 270,124 Kb

(A) MAXIMUM DCG REPORTED : 19.4072

(B) Total Number of permutations reported :
19893497375938370599826047614905329896936840170566570588205180312704857
99269519348241268656543105024000000000000000000000000

Number of URLs	nDCG Value Reported
At 50	0.37071213897397365
Whole Data set (at 103)	0.6357153091990775



Analysis of the precision Recall Curve:

Maximum precision reported is `0.57692 at recall=0.4`

Question 3:

Ques:- Relationship between ROC and PR curve

Both the curves are used to evaluate the performance of any ML algorithm. Few of the relations between these curves are as follows:

(i) There exists one to one correspondance between various points in ROC and P-R curve for a given data set. Provided that the value of recall is non zero.

Brief Explaination

for P-R curve we can uniquely determine TP, FP and FN values. Using this value TN value can also be derived. Similarly all these four values TP, TN, FP, FN can be uniquely determined for any point on ROC curve as well.

This implies that we can draw confusion matrix at any point on P-R or ROC space. This confusion matrix will be same for points in both the spaces.

Please note that if Recall = 0

Date / /

Then TN value won't be unique in PR curve so we won't be able to draw relationship between both the curves.

(ii) ~~If~~ A curve in ROC space dominates iff it dominates in PR space.

↳ The proof is written in the answer of question 2.

CLASSTIME | Page No.
Date / /

Question 2

Part 1: If curve dominates in ROC then it dominates in PR.

Proof: "Proof By contradiction"

- Suppose curve I dominates in ROC than curve II. In order to prove by contrad" we assume that curve II dominates in the corresponding PR curve than curve I.

- ∃ pt A on curve II s.t. pt B on curve I with same recall has lower precision, ie for Recall (r)
  $Pr(A) > Pr(B)$

- ∴ Recall (r) = Recall (A) = Recall (B)
  && recall ≡ TP rate
  ∴ TPR(A) = TPR(B)

Now since curve I dominates than II in ROC then $FPR(A) \geq FPR(B)$

$$FPR(A) = \frac{FP(A)}{\text{Total neg.}}$$

$$FPR(B) = \frac{FP(B)}{\text{Total neg.}}$$

$\Rightarrow FP_A \geq FP_B$ ——— (1)

Now

$$\text{Precision}(A) = \frac{TP}{FP_A + TP}$$

$$\text{Precision}(B) = \frac{TP}{FP_B + TP}$$

$\Rightarrow$ from (1)
we have $\text{Precision}(A) \leq \text{Precision}(B)$

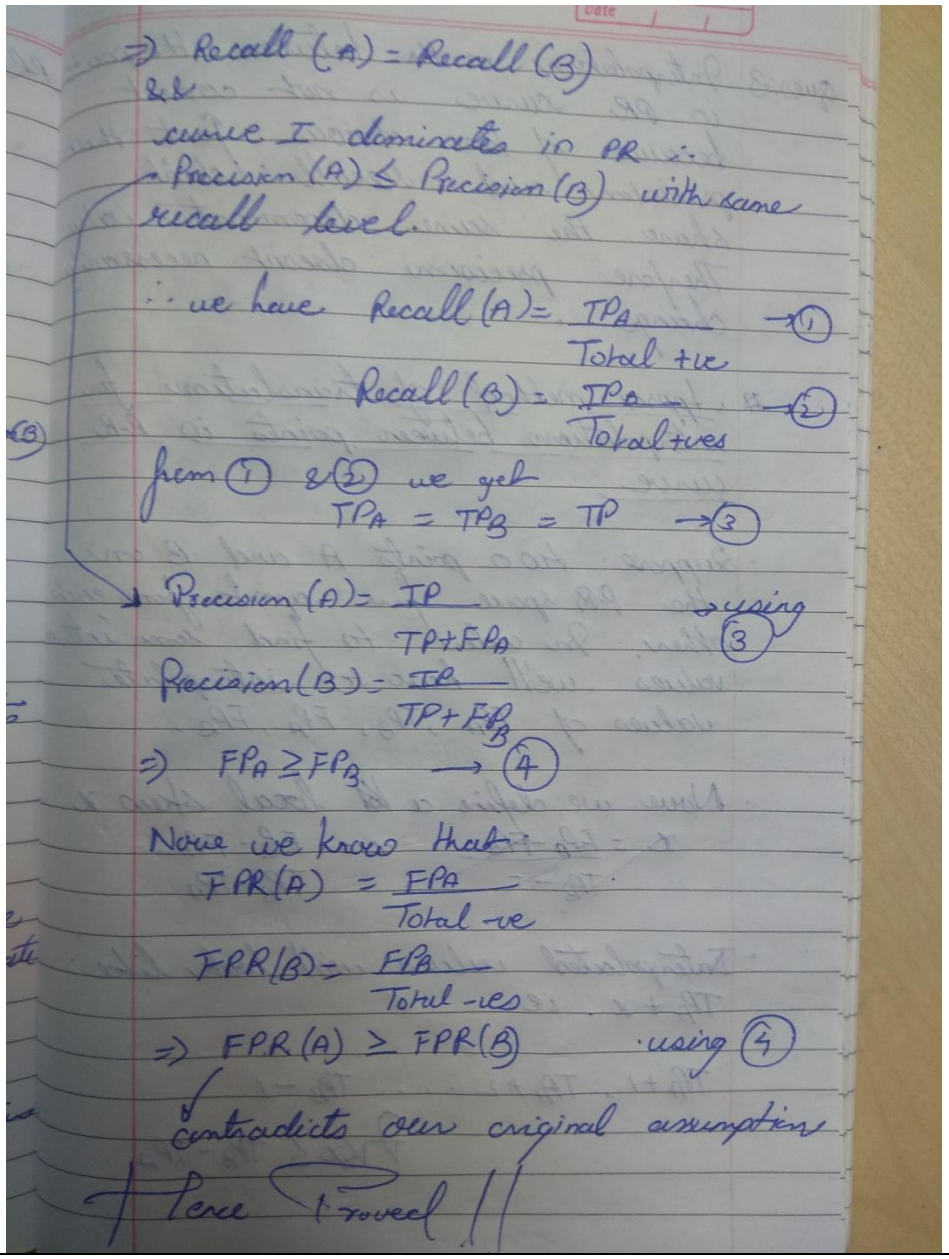
which contradicts our hypothesis.
Hence Proved

* Part II: If curve dominates in PR space then it dominates in Roc space.

"Proof by contradiction"

Assume curve I dominates in PR space than curve II but doesn't dominate in the corresponding ROC curve.

$\therefore$ $\exists$ pt A on curve II such that pt B on curve I with same y-axis value (TPR)
and $FPR(A) < FPR(B)$.



⇒ Recall (A) = Recall (B)

&&

curve I dominates in PR i.e.

Precision (A) ≤ Precision (B) with same recall level.

∴ we have Recall (A) = $\frac{TP_A}{\text{Total +ve}}$ → (1)

Recall (B) = $\frac{TP_B}{\text{Total +ves}}$ → (2)

from (1) & (2) we get

$TP_A = TP_B = TP$ → (3)

Precision (A) = $\frac{TP}{TP+FP_A}$ → using (3)

Precision (B) = $\frac{TP}{TP+FP_B}$

⇒ $FP_A \geq FP_B$ → (4)

Now we know that:

$FPR(A) = \frac{FP_A}{\text{Total -ve}}$

$FPR(B) = \frac{FP_B}{\text{Total -ves}}$

⇒ FPR (A) ≥ FPR (B) · using (4)

contradicts our original assumption

Hence Proved !!

Date / /

ques:3 Interpolation ~~on the~~ between the points in PR curve is not correct because of the obvious fact that precision and recall don't share the same denominator value. Therfore precision doesn't necessarily change.

Approx method of translation for interpolation between points in P-R curve

- Suppose two points A and B on the PR space far apart from each other. In order to find some intermediate values, we'll have to interpolate values of TP_A, TP_B, FP_A, FP_B.

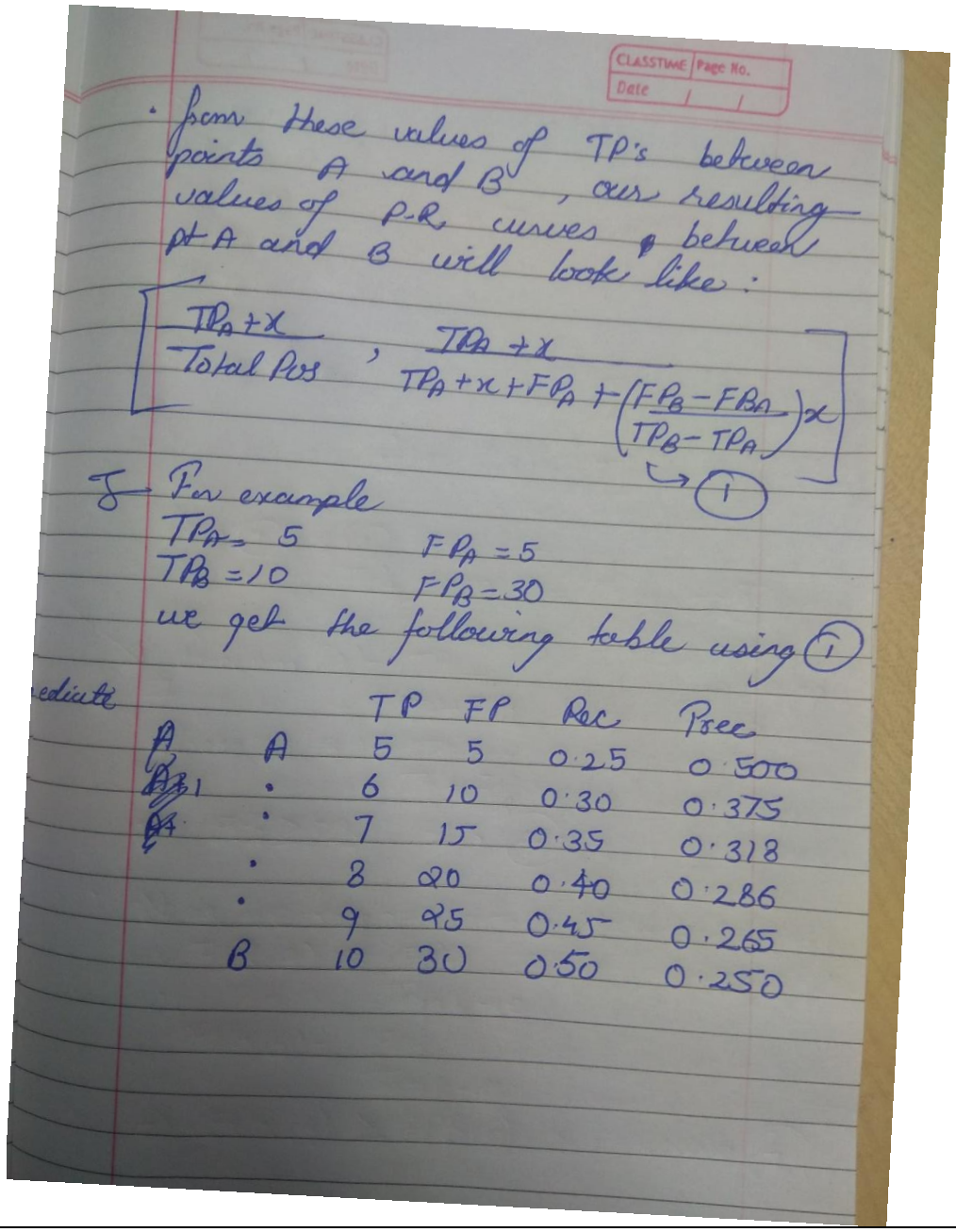
- Now we define a ~~kf~~ local skew x

$$x = \frac{FP_B - FP_A}{TP_B - TP_A} \qquad x = \frac{FP_B - FP_A}{TP_B - TP_A}$$

- Interpolated values will look like: $TP_A + x$. ie.

$TP_A + 1$, $TP_A + 2$ . . . $TP_B - 1$

$\forall 1 \leq x \leq TP_B - TP_A$



- from these values of TP's between points A and B, our resulting values of P-R curves between pt A and B will look like:

$$\left[\frac{TP_A + x}{\text{Total Pos}}, \frac{TP_A + x}{TP_A + x + FP_A + \left(\frac{FP_B - FP_A}{TP_B - TP_A}\right)x} \right] \quad \hookrightarrow (1)$$

- For example

$TP_A = 5$ $\quad FP_A = 5$

$TP_B = 10$ $\quad FP_B = 30$

we get the following table using (1)

		TP	FP	Rec	Prec
A	A	5	5	0.25	0.500
A1	.	6	10	0.30	0.375
A4	.	7	15	0.35	0.318
	.	8	20	0.40	0.286
	.	9	25	0.45	0.265
	B	10	30	0.50	0.250