

データサイエンス特論 授業課題 第三回分（情報検索評価）

2つ以上の検索エンジン（例 Google, Yahoo!, Yandex, Bing）で画像検索実験を行い、以下の手順で評価実験を行い、結果のレポートを提出してください。2 パターンの検索質問での結果は、複数ファイルになる場合、全部をまとめて ZIP してください。スケーラブルなグラフを全部 MS WORD や PDF にまとめられる場合は、1 つのファイルだけの提出でも結構です。締め切りは7月4日までです。

(1) 検索エンジンに以下の3つの表現を含む英語の入力フレーズをインプットしてください。何をクエリとして、インプットしたかわかるようにしてください。

- 特定のオブジェクト（通常、名詞）
- 色（あるいは色を示唆するもの）
- 特定の場所、または動作や状態

例として、

- **“green apples in a basket”**
- **“sliced golden kiwi on the dish”**
- **“black Prius on the road”**
- **“jumping whales in the blue ocean”**

のようなフレーズを各自で 2 パターン考えてください。

(2) 検索結果上位エンジンの実行結果上位 20 位までの画像検索結果を実行し、画面コピー（screen shot）を残してください。

(3) 上位 20 位までの個々の検索結果に対して、主観的な判断で結構ですので、

- ◎（3 点, highly relevant）、
- （2 点, relevant）、
- △（1 点, partially relevant）、
- ×（0 点, non-relevant）

として、結果を表（後述の例参照）として付与してください。

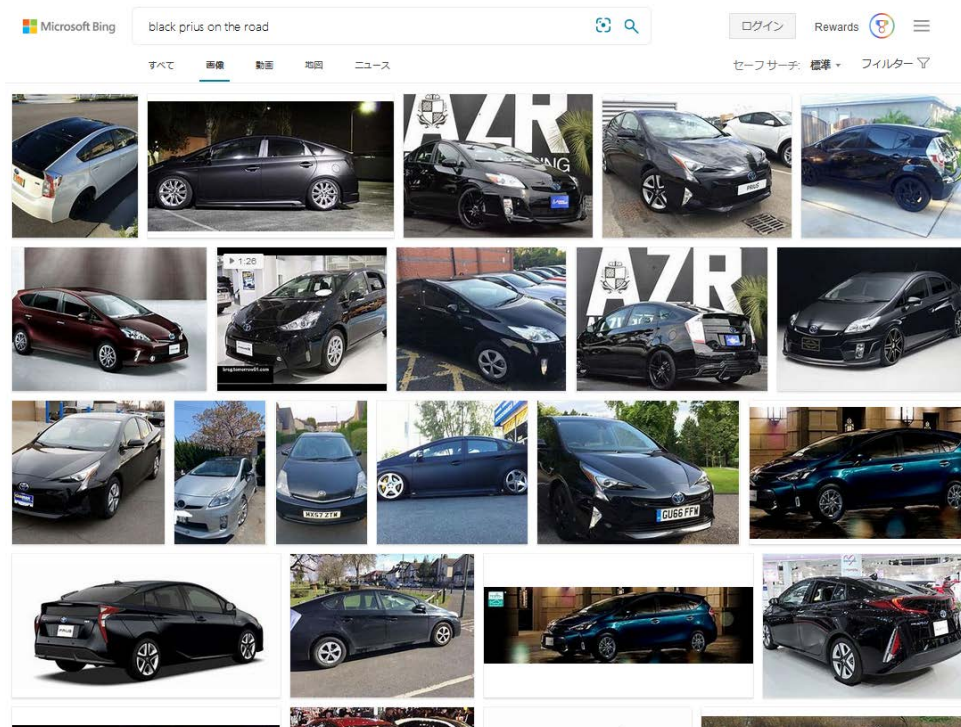
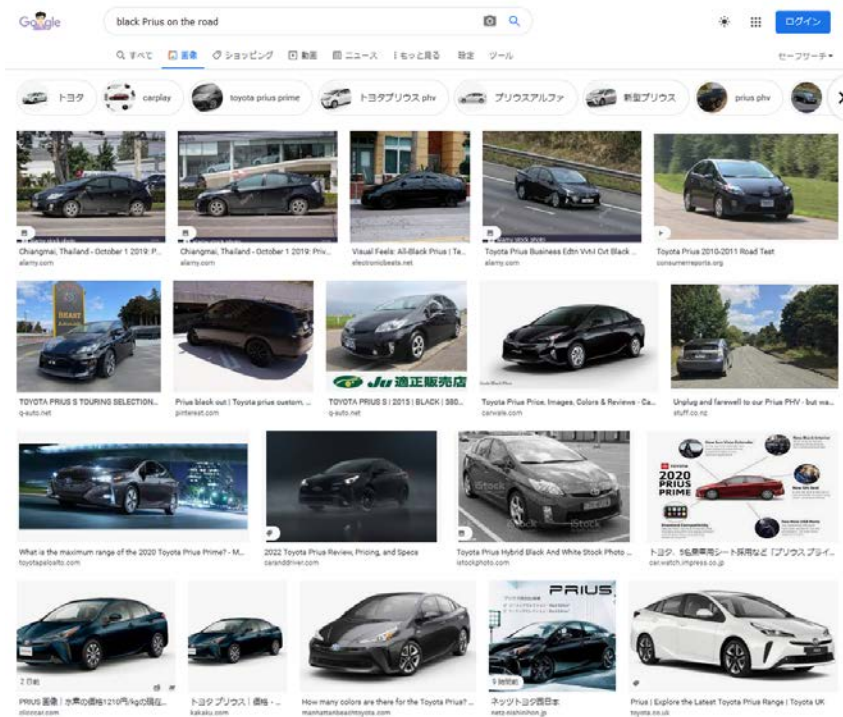
(4) (3)の表をもとに、CG, DCG, nDCG の値を計算し、nDCG のグラフをプロットしてください。（注：プロット結果は PDF, EPS, EMF, SVG など原則（いくら拡大してもギザギザのない）スケーラブルな描画としてください。つまりグラフ自体は JPEG や PNG 等の画像を、原則、貼り付けないよう工夫してください。この際、とりあげた 2 つ（以上）の検索エンジンの nDCG グラフを同時に表示し比較してください。

提出にあたっては、たとえば、WORD に記入したものでも、何らかの文書作成手法で策したものを PDF としたものでもよいとします。

提出するレポートはこの文書の p.2 から p.4 のグラフのところにあるような例までよく、p.4 の後半に描画サンプルとして掲載しているプログラムを提出する必要はありません。

例

(1) (2) “black Prius on the road” (Google 画像検索と Bing 画像検索)



(3)

「道路上の黒いプリウス」なので、黒くないもの、道路上でないものは、適合性を下げるか、不適合としています。

Google

Rank	1	2	3	4	5	6	7	8	9	10
	◎	◎	◎	◎	△	◎	△	◎	△	◎
	11	12	13	14	15	16	17	18	19	20
	○	◎	◎	×	×	×	△	△	×	×

Bing

Rank	1	2	3	4	5	6	7	8	9	10
	×	◎	◎	◎	△	×	△	△	○	△
	11	12	13	14	15	16	17	18	19	20
	○	×	◎	○	◎	×	△	◎	×	△

(4)

CG:Google [3. 6. 9. 12. 13. 16. 17. 20. 21. 24. 26. 29. 32. 32. 32. 32. 33. 34. 34. 34.]

CG:Bing [0. 3. 6. 9. 10. 10. 11. 12. 14. 15. 17. 17. 20. 22. 25. 25. 26. 29. 29. 30.]

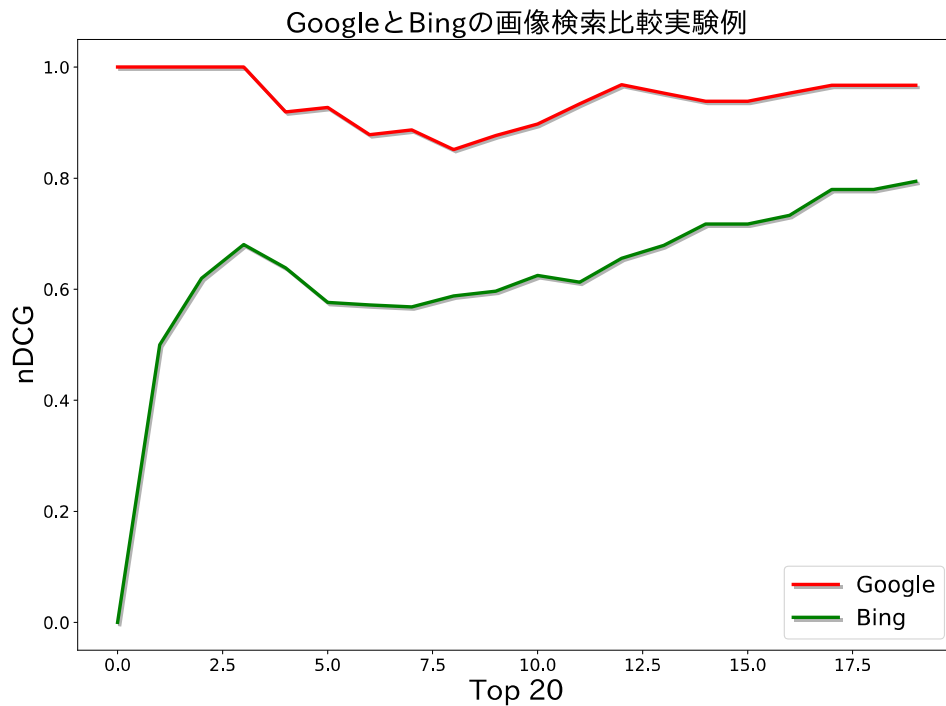
DCG:Google [3. 6. 7.89278926 9.39278926 9.82346582 10.98402424 11.34023143 12.34023143 12.6556963 13.55878629 14.13691594 14.97374478 15.78445924 15.78445924 15.78445924 16.02910979 16.26892225 16.26892225 16.26892225]

DCG:Bing [0. 3. 4.89278926 6.39278926 6.82346582 6.82346582 7.17967301 7.51300634 8.14393609 8.44496609 9.02309574 9.02309574 9.8338102 10.35910927 11.12698335 11.12698335 11.37163389 12.09107129 12.09107129 12.3224495]

nDCG:Google [1. 1. 1. 1. 0.91938533 0.92728361 0.87813477 0.88689323 0.85163938 0.87688453 0.89749544 0.93408072 0.96833018 0.9529751 0.93847263 0.93847263 0.95301844 0.96727661 0.96727661 0.96727661]

nDCG:Bing [0. 0.5 0.61990623 0.68060606 0.63861314 0.57604461 0.57173054 0.56811448 0.58778116 0.59654681 0.62463095 0.6127978]

0.65582064 0.67896024 0.71725575 0.71725575 0.73302616 0.77940176
0.77940176 0.79431662]



参考までに、プロット部分だけ、Python の matplotlib を使う場合の例を示します。なお、以下では日本語フォントの設定部分を含みますが、英語フォントだけの図（のタイトル）でも結構です。

```
# Jupyter notebook(VSCODE 等)を利用すると想定
%matplotlib inline
import numpy as np # Numpy
import matplotlib.pyplot as plt # 描画ライブラリ
import matplotlib.mlab as mlab
import matplotlib.font_manager as fm
import matplotlib.path_effects as path_effects

# 日本語 font の設定例（IPA 日本語フォントの設定例）
from matplotlib import font_manager
font_path='C:/Windows/Fonts/ipaexg.ttf' # Windows の場合
```

```

font_prop = font_manager.FontProperties(fname=font_path)
font_prop.set_style('normal')
font_prop.set_weight('light')
font_prop.set_size('12')
fp2 = font_prop.copy()
fp2.set_size('25')

# For plotting (上位 20 位までのランクと nDCG 値を保持する Numpy 配列)
data1 = np.zeros((2,20))
data2 = np.zeros((2,20))
for i in range(RANK):
    data1[0,i] = data2[0,i] = i

# 以下で、nFCG_Google, nDCG_Bing は事前に計算されていると仮定
data1[1,:] = nDCG_Google
data2[1,:] = nDCG_Bing

plt.figure(figsize=(14,10))
plt.plot(data1[0,:], data1[1:], color='red',label="Google",linewidth=3,
         path_effects=[path_effects.SimpleLineShadow(),
                       path_effects.Normal()])
plt.plot(data2[0,:], data2[1:], color='green',label="Bing",linewidth=3,
         path_effects=[path_effects.SimpleLineShadow(),
                       path_effects.Normal()])
plt.tick_params(labelsize=15)
plt.xlabel('Top 20',fontsize=20,font_properties=fp2)
plt.ylabel('nDCG',fontsize=20,font_properties=fp2)
plt.title('Google と Bing の画像検索比較実験例',fontsize=25,font_properties=fp2)
plt.legend(['Google', 'Bing'], loc='best', fontsize=20)
plt.savefig('Kadai3-nDCG.pdf') # スケーラブルな PDF に出力

```