Content-Based Exclusion Queries in Keyword-Based Image Retrieval

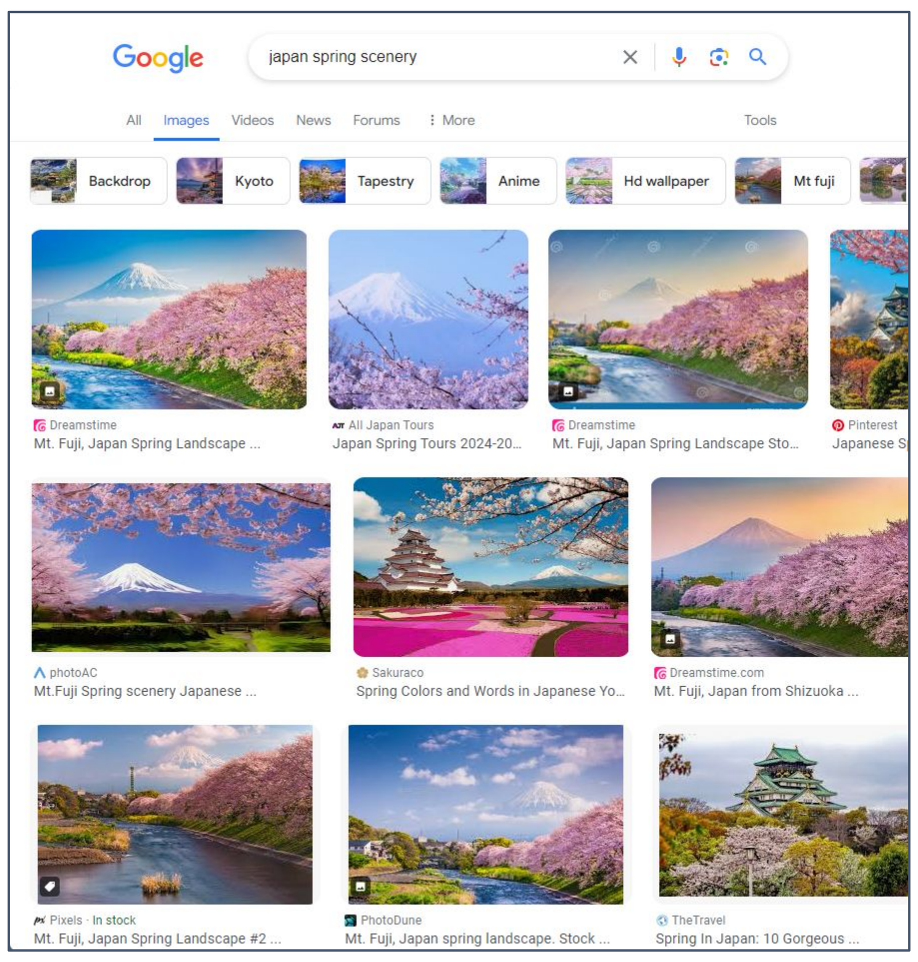
Eisaku Yoshikawa, Keishi Tajima (Kyoto University)

Exclusion Queries "A -B"

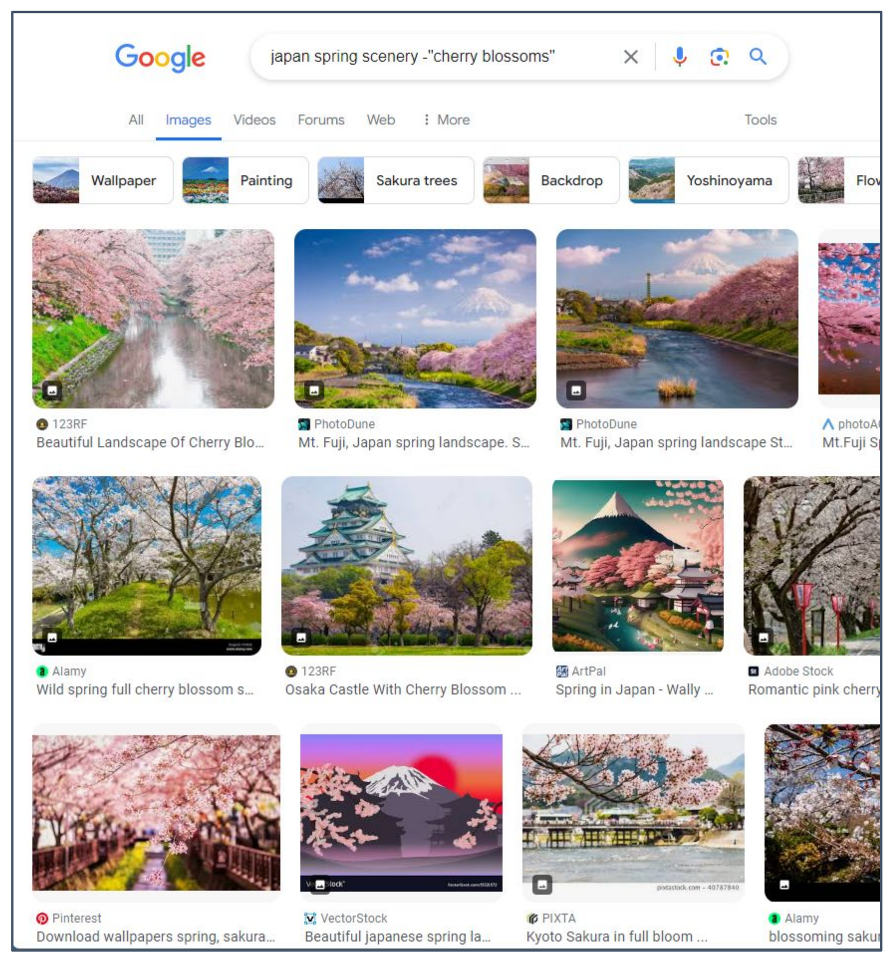
Exclude those including "B" from the results of the query "A".

- ☑ Use exclusion queries for page retrieval
- ☒ Use exclusion queries for image retrieval
  - A popular strategy for implementing keyword-based image retrieval is to use the keywords proximate to the image, which achieves high precision (and low recall).
  - By contrast, exclusion queries for image search based on the proximate keywords result in low precision (and high recall).

➡ We propose exclusion queries **based on image content** rather than textual information.



japan spring scenery 🔎



japan spring scenery -"cherry blossoms" 🔎

The results of the query “A -B” are far from the results of the query “A B”.

⇓

If we **exclude** some of the results of the query “A” **which are close to “A B”** we get the results of the query “A -B”

Appropriate images depend on the query.

⇓

We need to **classify queries** to determine correctness of output.

(1) Retrieve the result of the query “A”, which is a ranked list. Let S_A be the set of images in it, and $\mathrm{rank}_A(a)$ be the rank of $a \in S_A$ in the list.
(2) Retrieve the result of the query “A B”. Let S_{AB} be the set of images in it, and $\mathrm{rank}_{AB}(a)$ be the rank of a in the result.
(3) For each $a \in S_A$, compute $d(a, S_{AB})$, the distance (not in the mathematical sense) between a and the set S_{AB}.
(4) Determine a threshold value θ.
(5) Remove $a \in S_A$ s.t. $d(a, S_{AB}) \leq \theta$ from S_A.
(6) Return the remaining images in S_A in the order of $\mathrm{rank}_A(a)$.

* Mathematical Definition

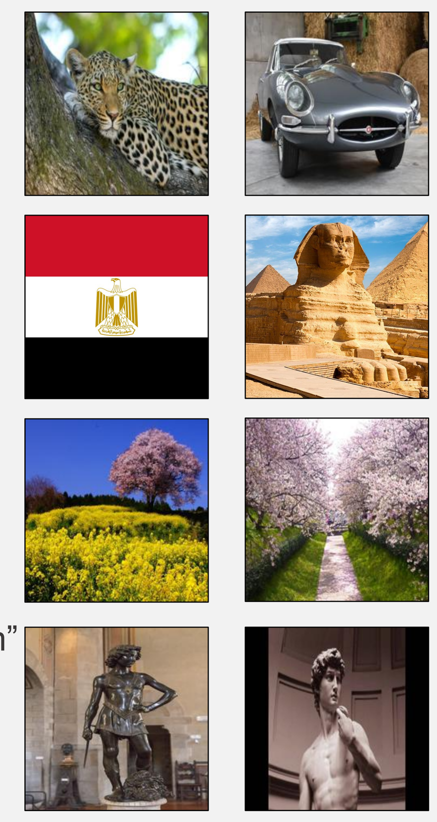
$$d(a, S_{AB}) = \min_{b \in S_{AB}} ||\vec{a} - \vec{b}||$$
$$S_1 = \{a \in S_A \mid d(a, S_AB) \leq \theta\}$$
$$S_2 = \{a \in S_A \mid d(a, S_AB) > \theta\}$$
$$\mathrm{sep}(S_1, S_2) = \frac{|S_1||S_2|}{|S_1| + |S_2|} \frac{||\vec{\mu_1} - \vec{\mu_2}||^2}{\sum_{x \in S_1} ||\vec{x} - \vec{\mu_1}||^2 + \sum_{x \in S_2} ||\vec{x} - \vec{\mu_2}||^2}$$
$$\theta = \underset{\theta \in D, |S_1| \geq 10, |S_2| \geq 10}{\arg\max} \mathrm{sep}(S_1, S_2)$$

Queries to classify polysemous words
- Exclude images of polysemic words A corresponding to the meaning of B
- Example: “Jaguar -car”

Queries without B
- Exclude images that contain B in the image
- Example: “Egypt -pyramid”

Queries where B is not the subject
- Exclude images where the subject of the image is B
- Example: “Japan spring scenery -cherryblossom”

Queries that remove by meta information
- Exclude images with B information (e.g. artist) that does not appear directly in the image
- Example: “David -Michelangelo”



Examples of each qureis

Queries	Imagess	Distance	Search results
“A” “A B”	A = [a1.png, a2.png, …] B = [b1.png, b2.png, …]	||a1 – B|| = 7 ||a2 – B|| = 24 ||a3 – B|| = 13 ||a4 – B|| = 2 ||a5 – B|| = 49 …	~~||a1 – B|| = 7~~ ||a2 – B|| = 24 ||a3 – B|| = 13 ~~||a4 – B|| = 2~~ ||a5 – B|| = 49 …
(1) Collect images “A” (2) Collect images “A B”	(3) Calculate distances	(4) Determine thresholds (5) Filtering	

Schematic of the proposed method

For the exclusion query “A-B
- **relevant**
  - images that should not be excluded from negative searches
- **irrelevant**
  - images that should be excluded from negative search results
  - assumed to be similar to the results of the query “A B”
- **irrelevant to A**
  - image that is not desirable as a search result for “A”
  - the search system will include it in the “A” solution

➡ Proposed method achieves **higher performance** than existing methods (Bing, Google)

* Comparison of 61 queries for Bing and 64 queries for Google

Table 2: Average precision@10 and MRR of methods

	Bing	B+ours (ratio)	Google	G+ours (ratio)
avg. p@10	0.410	0.678 (+65%)	0.547	0.602 (+10%)
σ^2	0.156	0.116	0.144	0.121
MRR	0.470	0.688 (+46%)	0.601	0.763 (+27%)
σ^2	0.205	0.149	0.167	0.157

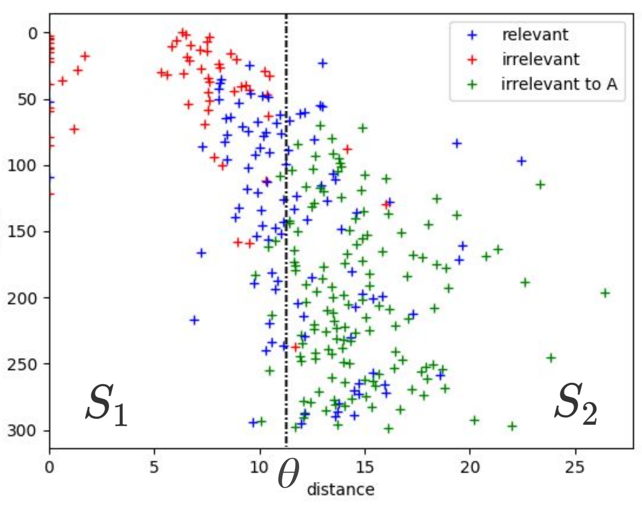
Table 4: comparison in precision@10 for each query

	base < ours	base = ours	base > ours	
Bing vs B+ours	41	16	4	61
Google vs G+ours	28	14	22	64

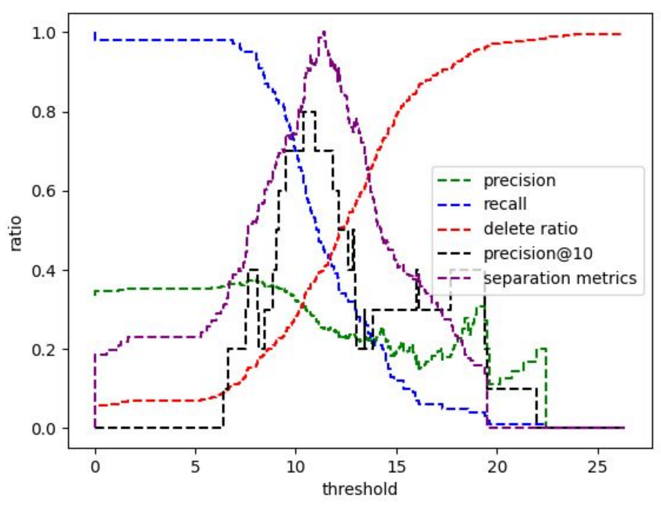
Table 3: The p-values of the paired t-tests

avg.p@10	Bing	Google
B+ours	$< 10^{-9}$	0.002
G+ours	$< 10^{-4}$	0.183

MRR	Bing	Google
B+ours	0.0006	0.085
G+ours	$< 10^{-4}$	0.003



Images in image set A and thresholds



We choose θ where **the separation metrics** takes its beak.

CONCLUSION

Considerations

- Negative image search using image content is effective
  - Negative image search with image content is effective and also improves the exclusion performance.
- The proposed method cannot achieve high performance if the dataset has few matching solutions
  - The query with low performance in this study could be considered a dataset problem.

Future works

- Since the proposed method has many elements that can be selected or adjusted, consider whether there are more appropriate elements.
  - Feature vector extraction for images other than CNN
  - Mechanism for determining threshold values
- Consider what is appropriate as a performance measure
  - Both precision@10 and MRR are indicators of whether or not there is a top matching solution, and evaluation of reproducibility and diversity is also desirable for “exclusion queries”.