

Inclass Exercise 3: Rank Merging

Suppose you are working for an information exploitation company, and one day your boss comes to you excited. “Look at this”, she says, “We’ve discovered a new algorithm that is good at classifying user queries: it can reliably separate them into type X and type not-X. This is good news, because type X queries are important to our business, and since we also know that queries of type X are generally handled better by mamesearch.com, than by any other search engine!”

“Great,” you say. “Then we can just route all X-type queries to mamesearch.com and make our analysts happy. It seems too good to be true.”

“It is,” she replies. “On most type X queries, google still returns some useful documents not seen in mamesearch, and ranks some good ones higher. We think that google is doing a better job of finding value in long documents. So here’s your next assignment: find a way to order the results for type X queries that uses all the information. Here’s some data to get you started. Give me a first-pass solution in 20 minutes. Oh, and by the way, whatever solution you come up with, give me a rough analysis of how likely it is to work for general type X queries, and tell me what you could do to improve the solution if we had more time.”

She leaves, dropping the following data on your desk, and saying over her shoulder “sometimes people say that utility is proportional to the reciprocal of rank, for some queries on some engines, in case that helps.”

query: “snowboarding vacations near West Texas” (a typical type X query)

<u>mamesearch</u> <u>results</u>	<u>google</u> <u>results</u>	<u>desired</u> <u>results</u>	<u>document</u> <u>lengths (kilobytes)</u>
1. A	V	A	A 1
2. B	Y	B	B 2
3. C	D	V	C 9
4. D	A	D	D 6
5. E	F	Y	E 1
6. F	G	F	F 1
7. G	B	G	G 2
8. H	Q	E	H 2
9. I	N	Q	I 1
10. J	Z	C	J 1
			N 4
			Q 3
			V 9
			Y 26
			Z 1