Query-based Configuration of Text Retrieval for Software Engineering Tasks

- Text Retrieval (TR) approaches are used to search information in software based on a given query, much like Google searches the web.
- To lead to good results, TR approaches need to be configured for the software system and the task they are supporting.
- Current approaches for automatic TR configuration apply a single TR configuration for all queries. This leads to suboptimal results.
- We present the first approach, ***QUEST***, bringing TR configuration selection to the ***query level*** in software engineering.
- ***QUEST*** recommends the best TR configuration for each given query; it uses a ***supervised machine learning approach*** which determines the TR configuration that performs the best for each query based on its ***properties***.
- We evaluated QUEST in the context of ***feature and bug localization*** and found that it is able to recommend one of the top three TR configurations for a query with a 69% accuracy, on average.
- We compared the results obtained with the configurations recommended by QUEST for every query with those obtained using a single TR configuration for all queries in a system and in the entire dataset. We found that using QUEST we obtain better results than with any of the considered TR configurations.

Table 1. Percentage of queries per system for which QUEST improved, preserved, and worsened the retrieval performance compared to the best TR approach

System	Improved	Preserved	Worsened
apache-nutch-1.8	29.2%	66.7%	4.2%
apache-nutch-2.1	38.5%	41.0%	20.5%
bookkeeper-4.1.0	30.6%	53.2%	16.1%
commons-math3-3.0	19.4%	69.4%	11.1%
derby-10.9.1.0	25.3%	58.7%	16.0%
mahout-0.8	25.0%	72.7%	2.3%
openjpa-2.0.1	42.2%	40.0%	17.8%
pig-0.11.1	33.8%	43.8%	22.5%
pig-0.8.0	28.6%	39.7%	31.7%
solr-4.4.0	30.1%	47.0%	22.9%
tika-1.3	37.8%	37.8%	24.4%
zookeeper-3.4.5	10.4%	76.4%	13.2%
Average	*29.2%*	*53.9%*	*16.9%*

Table 2. Top-k recommendation accuracy achieved by QUEST when using within-and cross-project training.

Training strategy	System	Top 1	Top 2	Top 3
Within-project	apache nutch 1.8	57%	75%	86%
	apache nutch 2.1	39%	62%	69%
	bookkeeper 4.1.0	23%	58%	69%
	commons-math3 3.0	62%	75%	79%
	derby 10.9.1.0	31%	53%	73%
	mahout 0.8	53%	70%	78%
	openjpa 2.0.1	39%	70%	73%
	pig-0.8.0	20%	34%	48%
	pig-0.11.1	29%	45%	58%
	solr-4.4.0	22%	39%	56%
	tika 1.3	53%	63%	65%
	zookeeper-3.4.5	30%	45%	70%
	Average	38%	***57%***	69%
Cross-project		13%	24%	35%

Table 3. Percentage of queries for which QUEST improved, preserved, and deteriorated the retrieval performance compared to each baseline.

Baseline	Improved (med. improv.)	Preserved	Deteriorated (med. deter.)
E1C1	27.8% (-84)	53.1%	19.1% (233.5)
E2C4	67.8% (-367)	18.2%	14.0% (763)
E2C3	71.9% (-412)	16.6%	11.5% (159)
E5C1	51.5% (-42.5)	21.4%	27.1% (75)
E6C1	35.7% (-48)	31.0%	33.3% (61)
E6C2	41.8% (-60.5)	29.0%	29.2% (66)
E3C1	44.6% (-40)	25.5%	29.9% (66)
E4C3	41.3% (-53)	30.5%	28.2% (99)
Average	*47.8% (-138.4)*	*28.2%*	*24.0% (190.3)*

Sonia Haiduc | Dept. of Computer Science | shaiduc@cs.fsu.edu