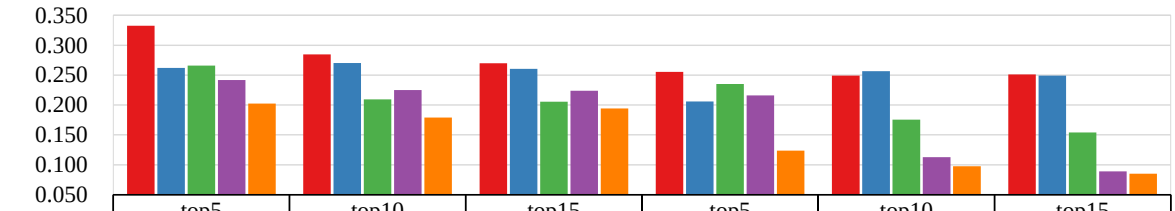

	McGill AK dataset			Android AK dataset		
	top5	top10	top15	top5	top10	top15
CK	0.333	0.285	0.270	0.255	0.249	0.251
Lucene	0.262	0.270	0.261	0.206	0.257	0.249
WE	0.266	0.210	0.205	0.235	0.176	0.154
SOTU	0.242	0.225	0.224	0.216	0.113	0.089
PLAN	0.202	0.179	0.194	0.124	0.098	0.085

Figure R1: Average JSD of PLAN and the baselines on two datasets (the lower the better)

1 Jensen Shannon Divergence Values

RQ1: How effective is PLAN in returning complementary API knowledge?

Motivation: In this RQ, we would like to verify whether the $\langle API, QA\rangle$ pairs and $\langle API, FRA\rangle$ pairs returned by PLAN contain the complementary API knowledge.

Approach: We compare the Jensen Shannon Divergence (JSD) of PLAN with Lucene, WE, CK, and SOTU on two datasets. JSD employs the probability distribution of words to measure the distance between documents [**?**], which has been widely used to measure whether two documents contain the complementary information [**? ? ?**]. For example, **?**] employed JSD to evaluate whether the test report and its summarization contain complementary information.

In this paper, we used the JSD value to assess whether the returned $\langle API, QA\rangle$ pairs and the returned $\langle API, FRA\rangle$ pairs contain the complementary API knowledge. For the top-k returned results of a given query, if any two pairs contain the same API and come from different sources (e.g., one pair from API tutorial and one pair from SO), we used these two pairs to construct the tuple. We first calculated JSD_{tup} between the $\langle API, QA\rangle$ pair and the $\langle API, FRA\rangle$ pair in each tuple. Then, we adopted the average JSD_{tup} of tuples as the JSD_q of the query. Finally, the average JSD_q of queries was used to measure an approach. The JSD_{tup} can be written as:

$$JSD_{tup} = H(\frac{p_s + p_t}{2}) - \frac{H(p_s) + H(p_t)}{2} \quad (1)$$

where s and t separately denote the $\langle API, QA\rangle$ pair and the $\langle API, FRA\rangle$ pair. $H(p) = \sum(h(\frac{n(\omega,d)}{T_d}))$ and $h(x) = -xlogx$. $n(\omega, d)$ is the frequency of a word ω appearing in a pair d (d is s or t) and T_d is the total number of words appearing in d. The value of JSD is between 0 and 1. The lower the better. **Results:** Figure R1 presents the average JSD values of different approaches on two datasets, with k being 5, 10, and 15. In general, PLAN achieves better performance than the baseline approaches. For example, when returning the top-5 results, PLAN outperforms the baseline approaches in terms of average JSD at least by 0.132 on

the two datasets. Experimental results indicated that the complementariness of API knowledge provided by PLAN is higher than the baselines. Besides, PLAN could provide more complementary API knowledge than the baselines. Taking the top-10 returned results as an example, for each query on the McGill dataset, PLAN could provide 5.671 tuples containing complementary API knowledge on average, while CK could only find 2.062 tuples. For each query on the Android dataset, on average 8.050 tuples can be returned to a query by PLAN, whereas only 1.895 tuples can be returned by CK. Clearly, PLAN provides more complementary API knowledge for developers to learn APIs. All the p-values are smaller than 0.04, which confirms that the improvement achieved by PLAN is statistically significant. Besides, $|\delta|$ values are all greater than 0.5, which means a large effect size.