

# AnswerBot

Automated Generation of Answer Summary  
to Developers' Technical Questions

**Bowen Xu**, Zhenchang Xing, Xin Xia, David Lo



# How to find solutions to our problems?

Problem

Query




Relevant  
Documents

Target  
Information

Solved



# Relevant Documents

 <input type="text" value="difference hashtable hashmap java"/>  			
Rank	Title	#Answers	#Words
1	Differences between HashM	37	4135
2	Difference between Hashtable and ConcurrentHashMap(HashMap)	5	847
3	What are the differences between Hashtable and hashmap? (Not specific to Java)	3	747
4	Difference between ConcurrentHashMap and Hashtable purely in Data Structures	3	565
5	What are the differences between Hashmap vs Hashtable in theory?	3	477

51 answers | 6,771 words in total!

Skip reading?



Reading all these answers?



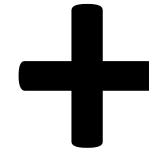
30 minutes to read

# Target Information

Aspect	#Answers
Synchronization/Thread Safety	23
Null Keys/Null Values	14
Performance	8
Evolution History	10
Iteration	13

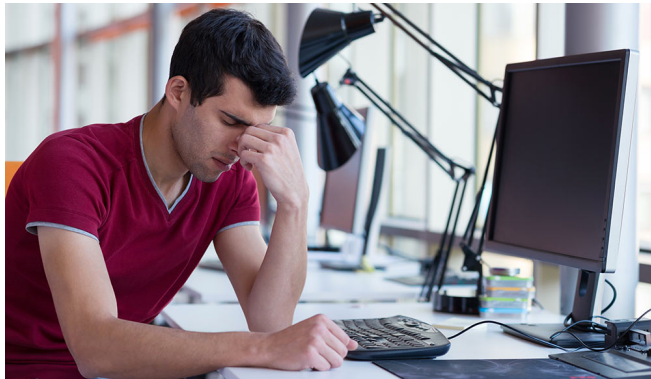


**Redundant** Information



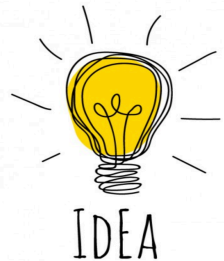
**Irrelevant / Low-quality**  
Information

# Things are more difficult than we imagined



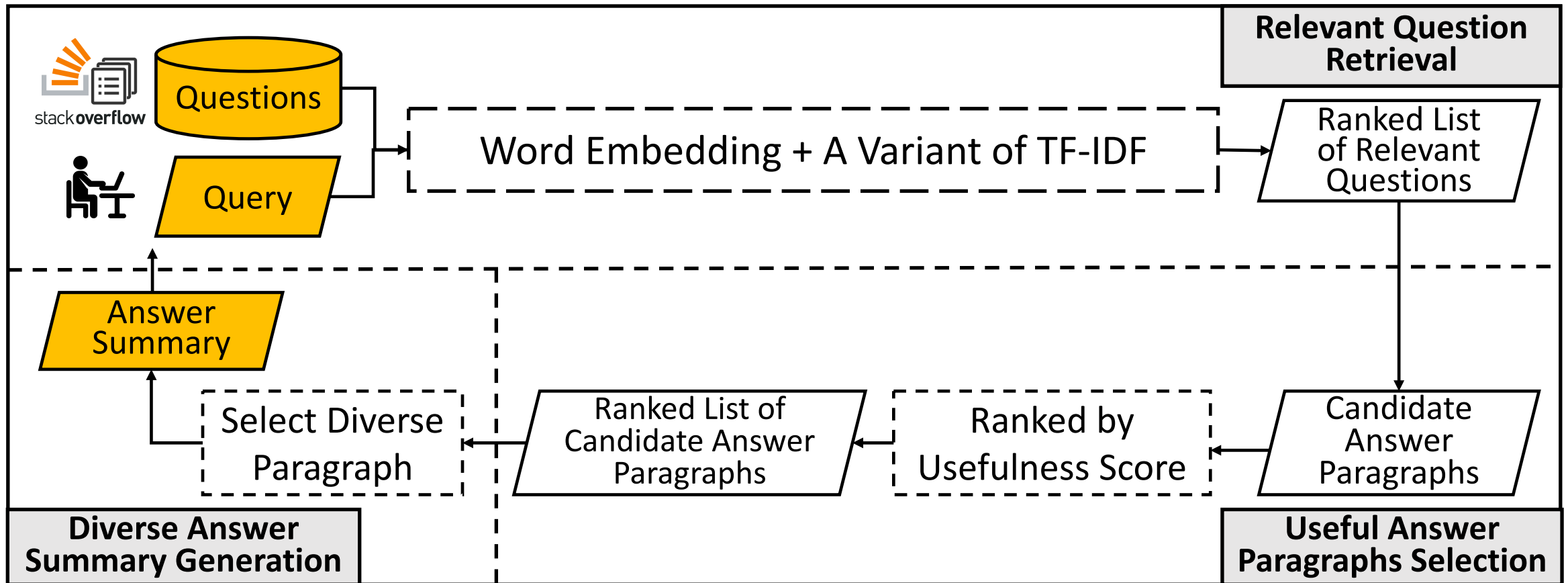
Survey based on 72 developers

- “Some questions received **too many long answers**, and many of these answers have **redundant content...**”
- “I notice **even the best answers** in Stack Overflow often answer the questions **only in one aspect...** Thus, I expect the tool should **provide a diversity of potential answers...**”
- “Google will return a **number of “relevant” links** for a query, and I have to click into these links, and read a number of paragraphs. It is really **time-consuming**. A tool which generates potential answers can **save my time wasted on reading a lot of irrelevant content...**”

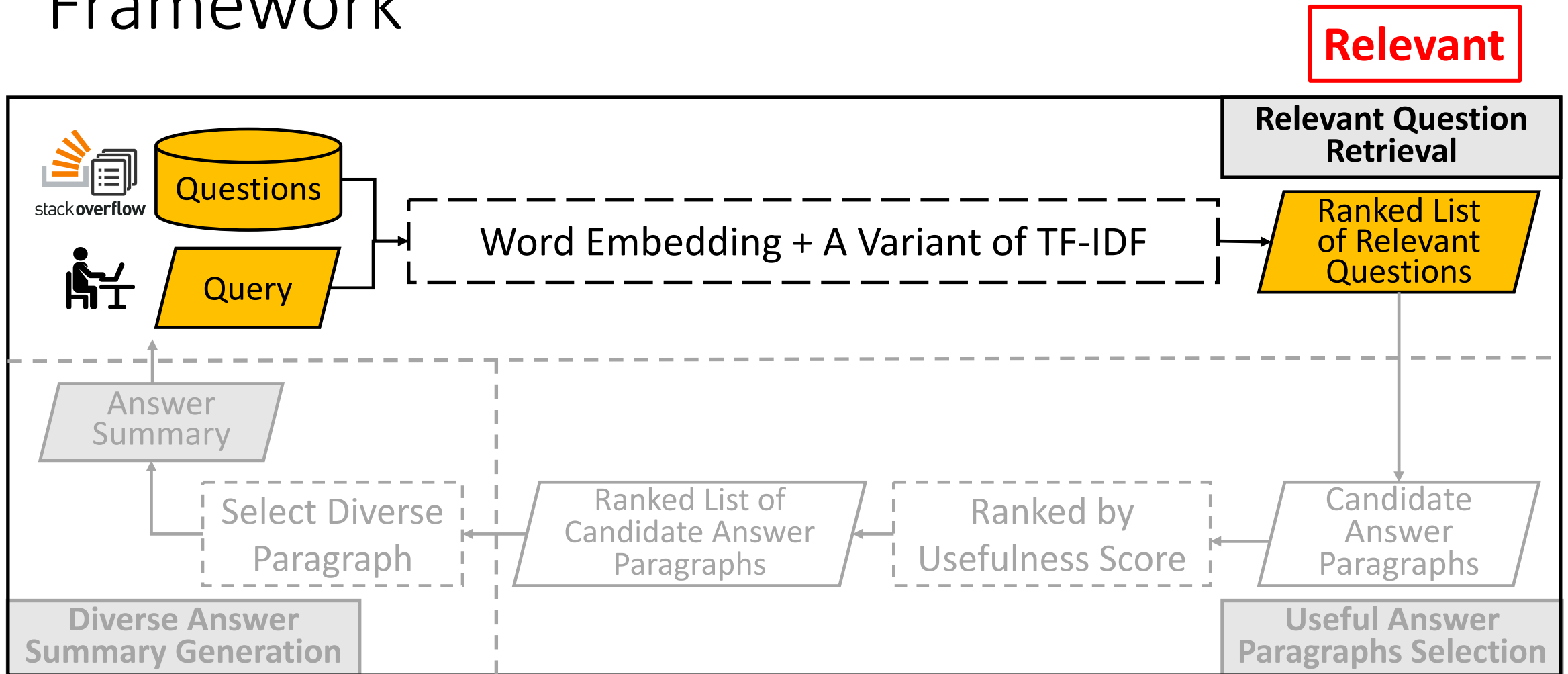


Developers really need an effective technique to generate an answer summary with **diverse**, **relevant** and **high-quality** information from answer posts.

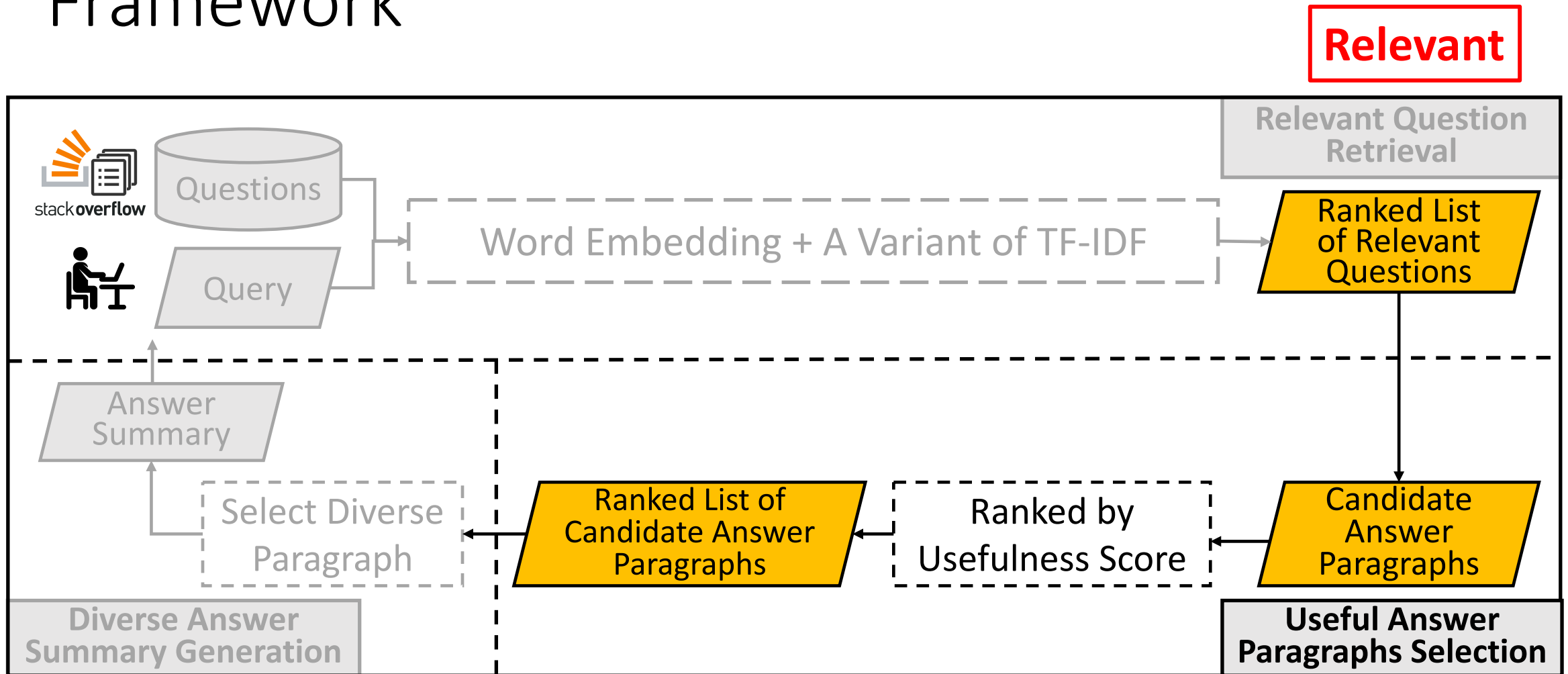
# Framework



# Framework



# Framework

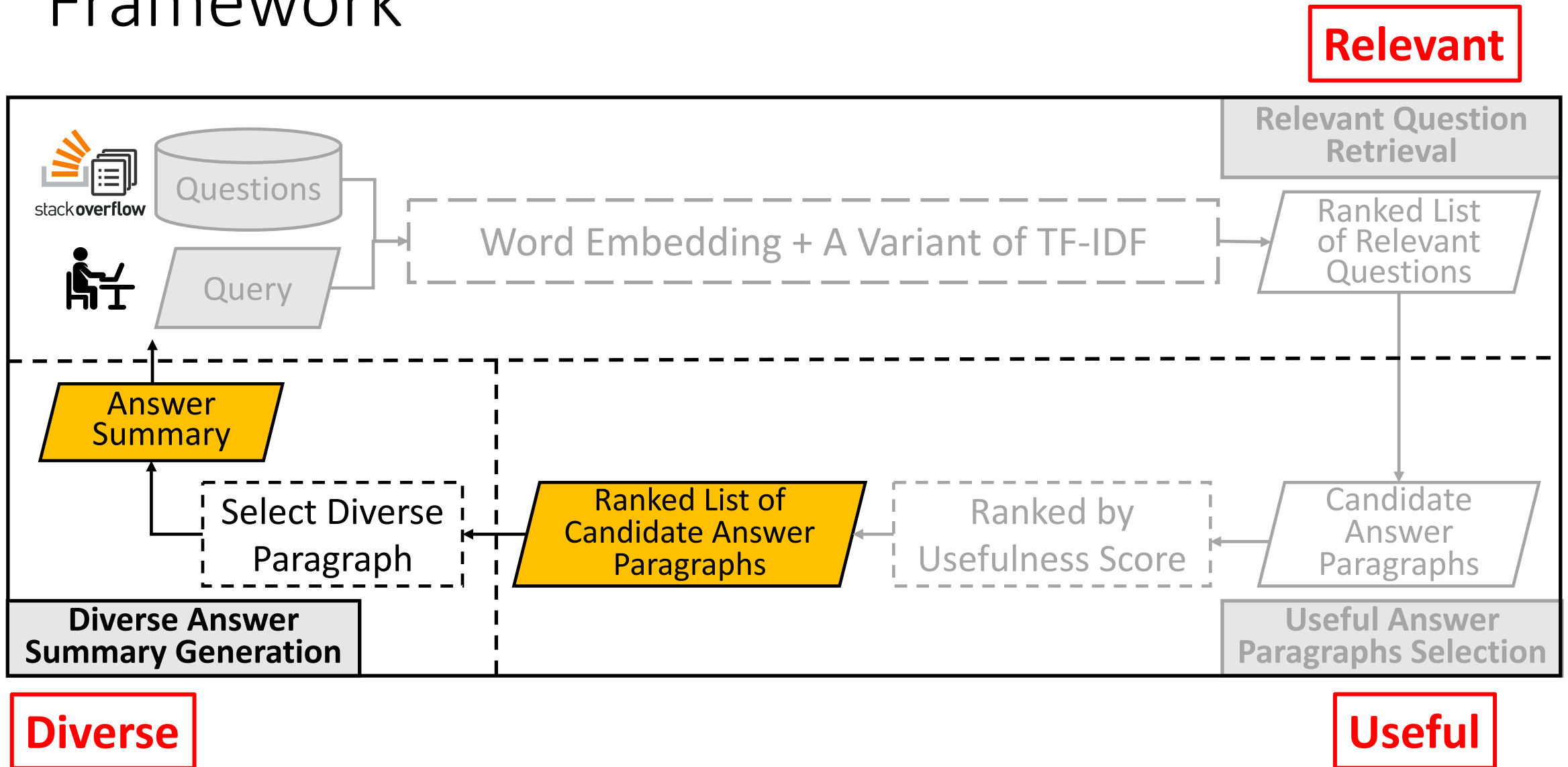


**Relevant**

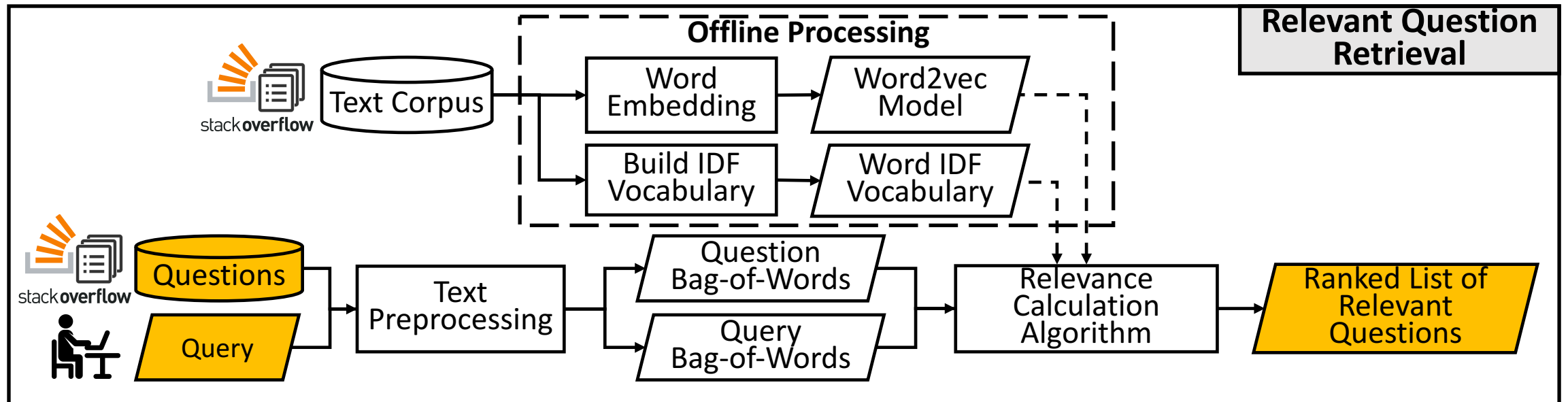
**Useful**



# Framework

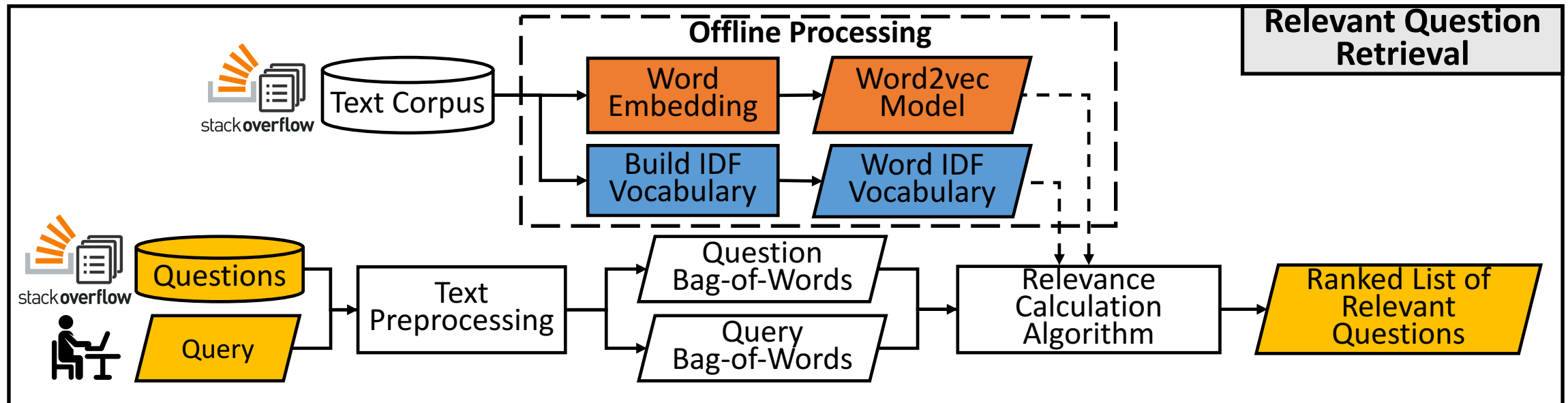


# Relevant Question Retrieval



# Relevant Question Retrieval

- An approach combine **Word Embedding + A Variant of TF-IDF**

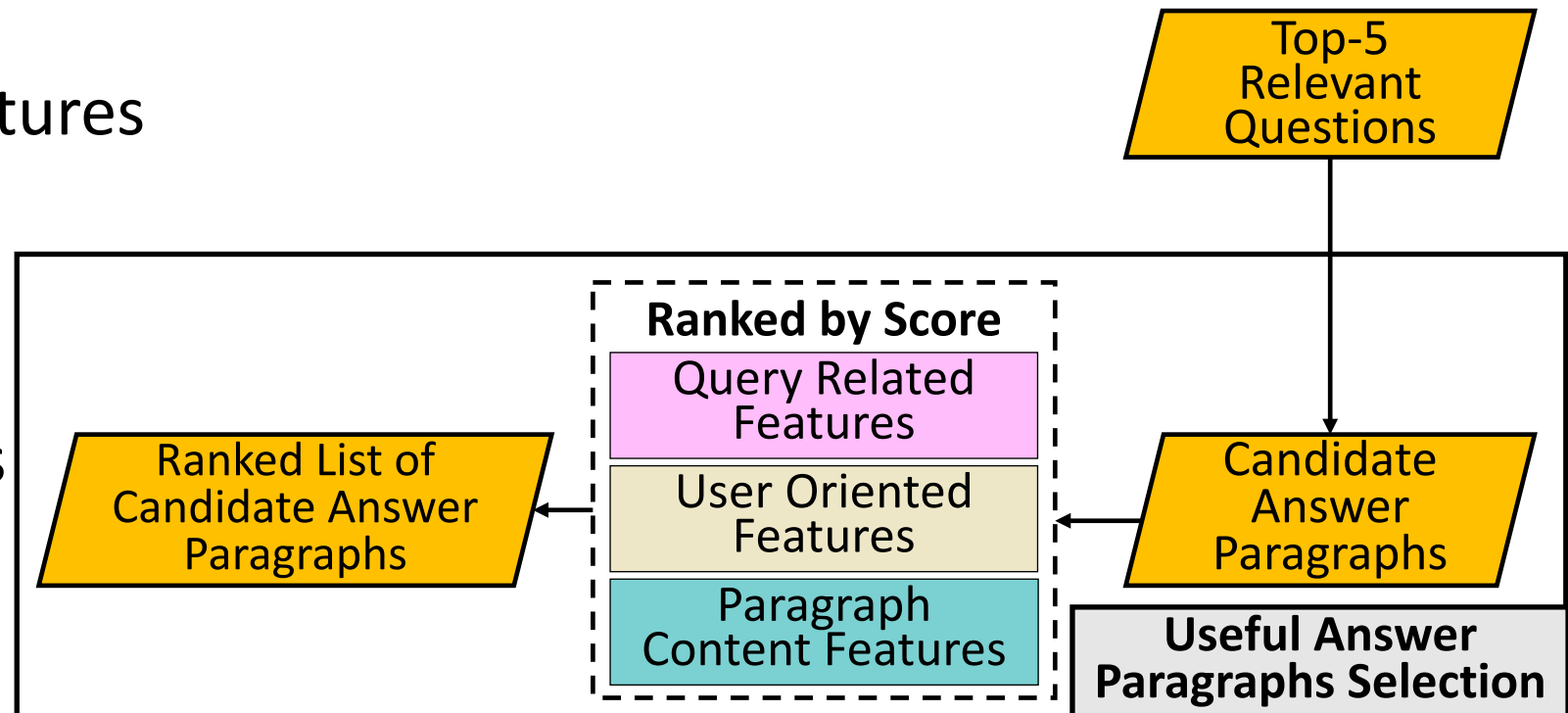


**Word Embedding: Word-level Similarity**

**A Variant of TF-IDF: Document-level Similarity**

# Useful Answer Paragraphs Selection

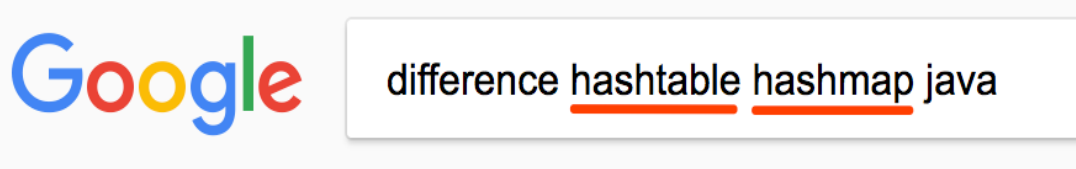
- Query related features
  - Relevance to query
  - Entity overlap
- Paragraph content features
  - Information entropy
  - Semantic patterns
  - Format patterns
- User oriented features
  - Paragraph position
  - Vote on answer



# Query related features

## *Entity Overlap (EO)*

Query



Paragraph

1. Hashtable is synchronized, whereas HashMap is not. This makes HashMap better for non-threaded applications, as unsynchronized Objects typically perform better than synchronized ones.
2. Hashtable does not allow `null` keys or values. HashMap allows one `null` key and any number of `null` values.
3. One of HashMap's subclasses is LinkedHashMap, so in the event that you'd want predictable iteration order (which is insertion order by default), you could easily swap out the HashMap for a LinkedHashMap. This wouldn't be as easy if you were using Hashtable.

$$Score_{EO} = \frac{|Eq \cap Eap|}{|Eq|}$$

# User oriented features

## Paragraph Position (PP)

### summary first and then sub details

Apart from the differences already mentioned, it should be noted that since Java 8, `HashMap` dynamically replaces the Nodes (linked list) used in each bucket with `TreeNodes` (red-black tree), so that even if high hash collisions exist, the worst case *when searching* is

$O(\log(n))$  for `HashMap` Vs  $O(n)$  in `Hashtable`.

\*The aforementioned improvement has not been applied to `Hashtable` yet, but only to `HashMap`, `LinkedHashMap`, and `ConcurrentHashMap`.

FYI, currently,

- `TREEIFY_THRESHOLD = 8` : if a bucket contains more than 8 nodes, the linked list is transformed into a balanced tree.
- `UNTREEIFY_THRESHOLD = 6` : when a bucket becomes too small (due to removal or resizing) the tree is converted back to linked list.

share edit flag

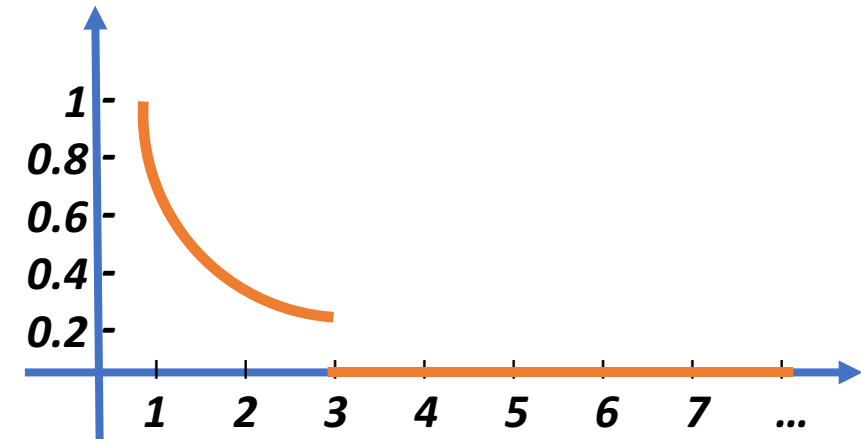
answered May 4 '16 at 15:04



Konstantinos Chalkias

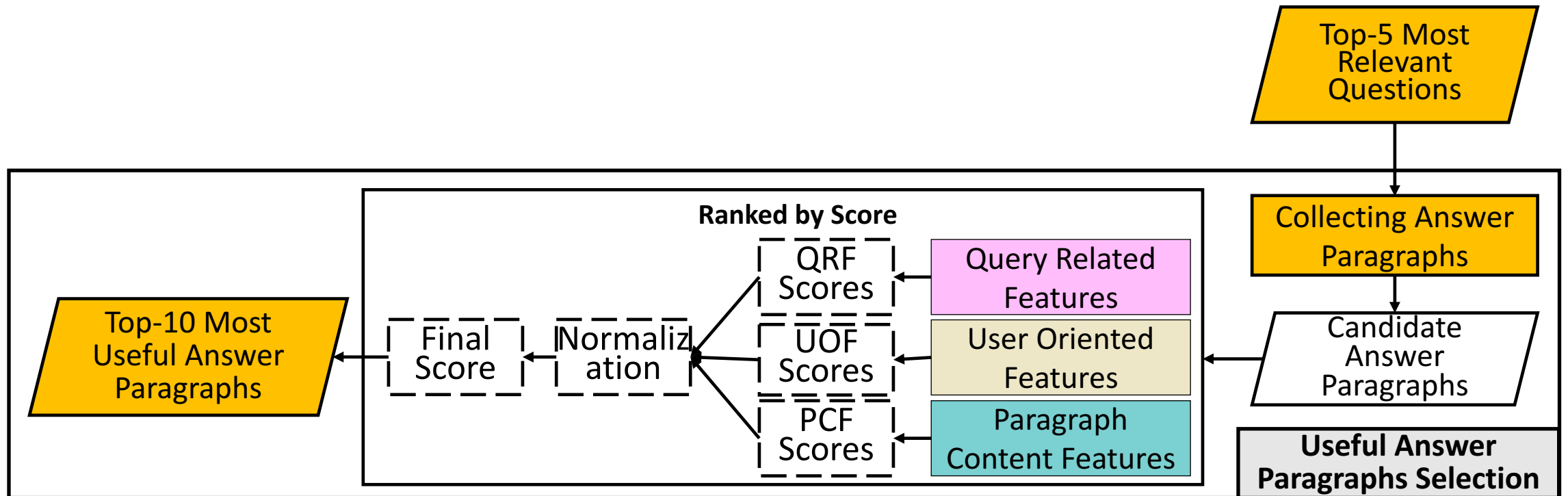
2,621 ● 2 ● 13 ● 19

[add a comment](#)

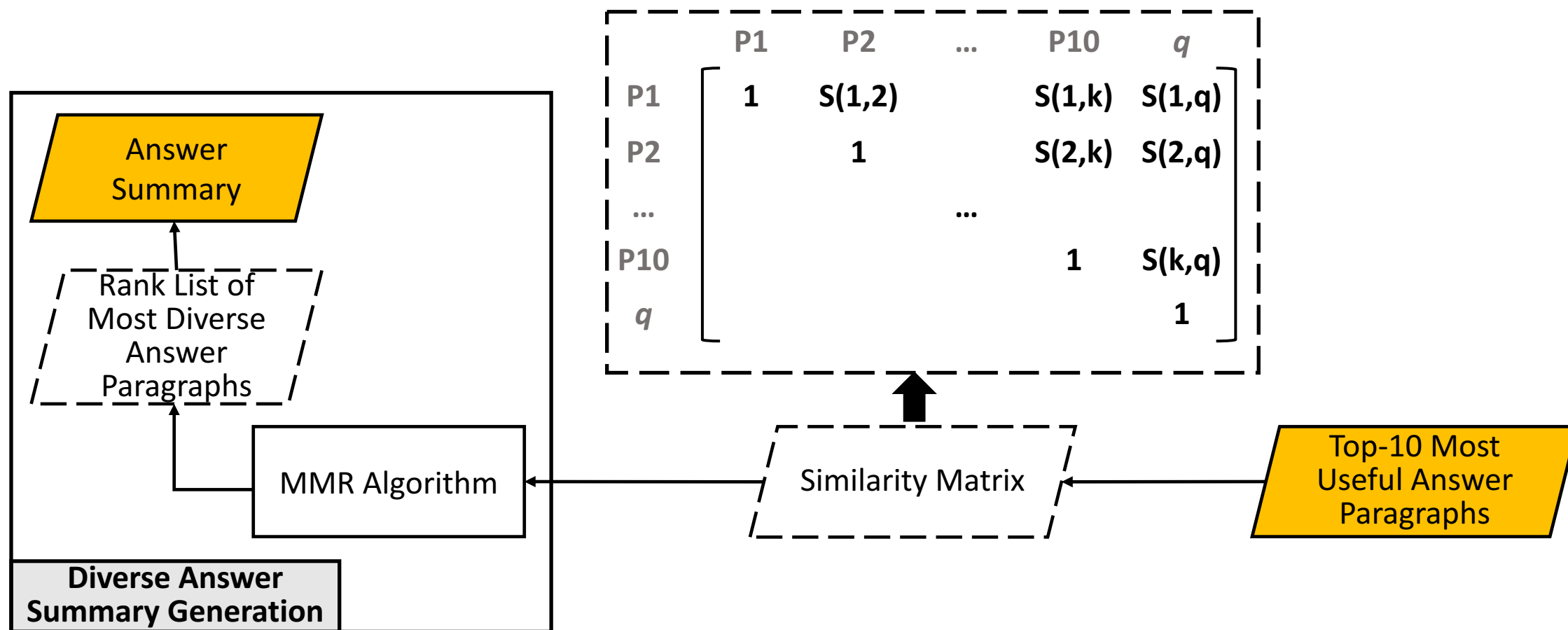


$$Score_{IE} = \begin{cases} \frac{1}{pos}, & 1 \leq pos \leq 3 \\ 0, & pos > 4 \end{cases}$$

# Ranking by score



# Diverse Answer Summary Generation





# Research Questions

## ➤ *Research Question 1*

### **Overall Performance**

## ➤ *Research Question 2*

Effectiveness of **relevant question retrieval** component

## ➤ *Research Question 3*

Effectiveness of **useful answer paragraph selection** component

# Experiment Method

## ➤ Dataset

- **228,817 Java questions** from Stack Overflow Data Dump of March 2016.
- Randomly selected **100 query questions** cover a diversity of aspects of Java programming.

## ➤ Evaluation

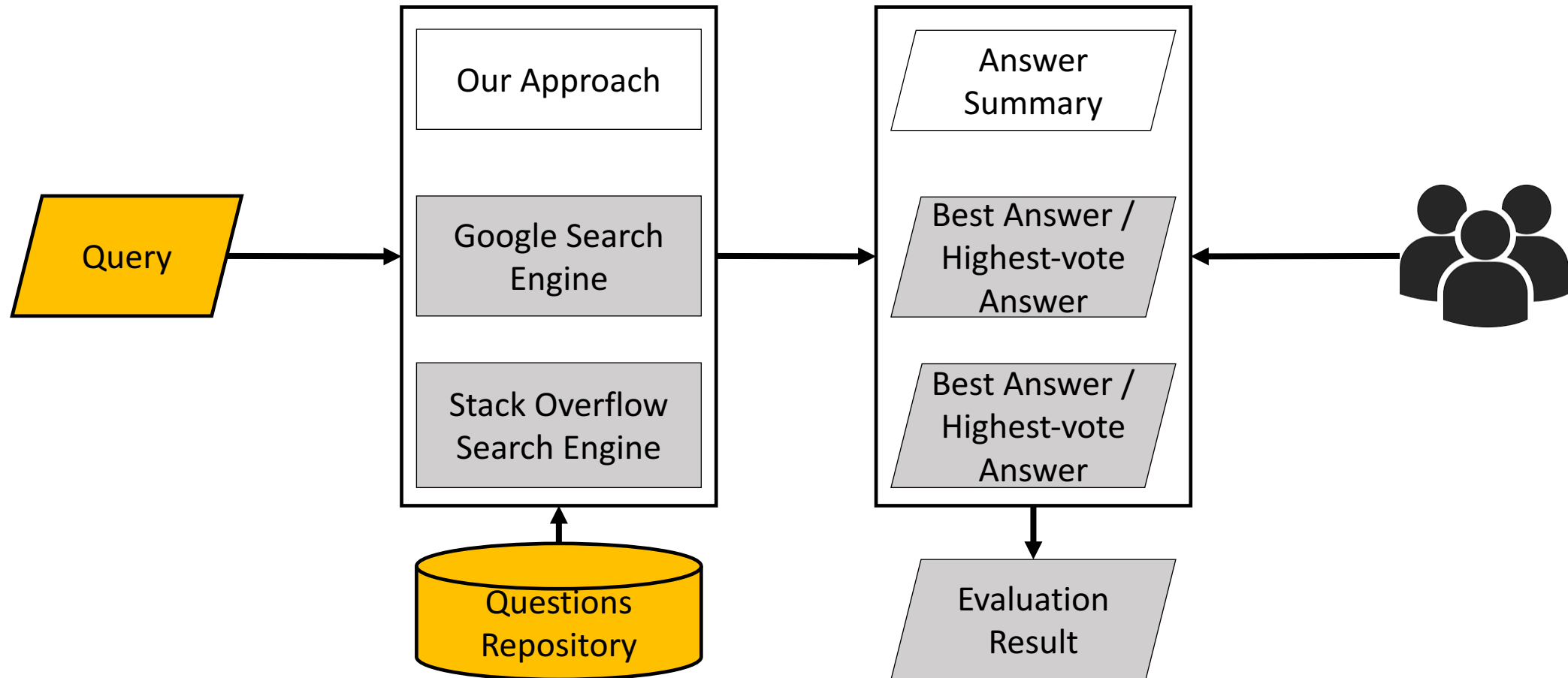
RQs	Group1 (P1, D1, D2, D3)	Group2 (P2, D4, D5, D6)
RQ1	Q1-Q50	Q51-Q100
RQ2	Q51-Q100	Q1-Q50
RQ3	Q51-Q100	Q1-Q50

1. working experience on Java are vary from 2 to 8 years, with an average 4.6 years
2. participants do not know which result is generated by which approach
3. None of them is the author of this paper

P: Postdoctoral Fellows

D: PhD Students

# RQ1 Evaluation on Overall Performance



# RQ1 Evaluation on Overall Performance

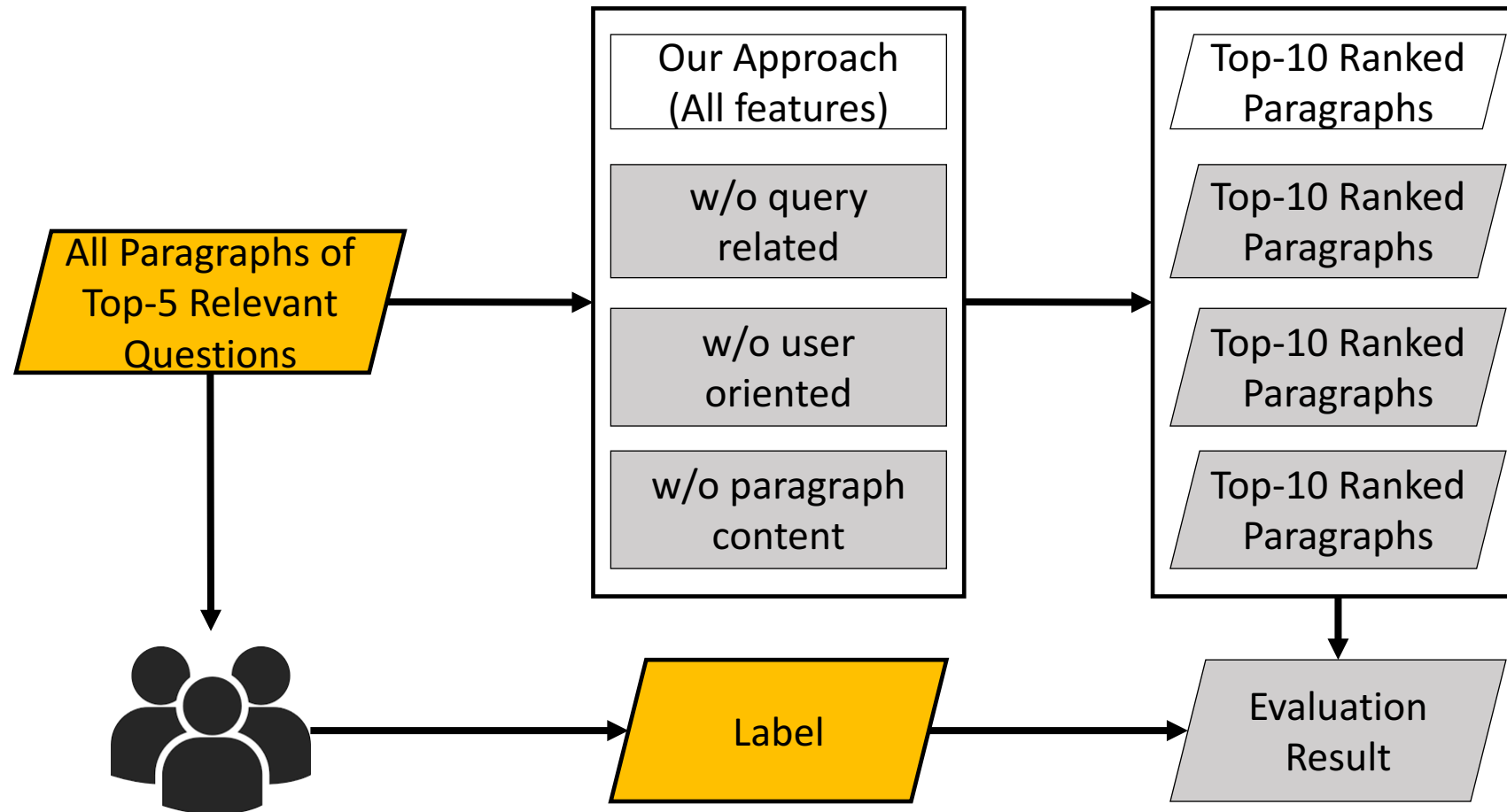
	Relevance	Usefulness	Diversity
Our Approach	3.450	3.720	3.830
Baseline_Google	3.440	3.480*	2.930***
Baseline_Stack Overflow	2.576***	2.712***	2.305***

\*\*\*p<0.001, \*\*p<0.01, \*p<0.05

## Result:

- Best or highest-vote answers may not cover as diverse information as the developers need.
- It is worth reading more answer posts to summarize more complete information.

# RQ3 Evaluation on Useful Answer Paragraph Selection



# RQ3 Evaluation on Useful Answer Paragraph Selection

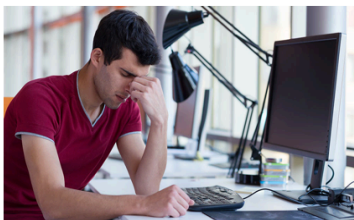
	Top@1	Top@5	Top@10	MRR
all features	0.660	0.990	1.000	0.803
w/o query related	0.610*	0.970	1.000	0.758**
w/o user oriented	0.500***	0.980	1.000	0.699***
w/o paragraph content	0.570*	1.000	1.000	0.744**

\*\*\*p<0.001, \*\*p<0.01, \*p<0.05

## Result:

- **User-oriented** features > **Paragraph content** features > **Query-related** features
- General effectiveness of all answer paragraph selection features

# Things are more difficult than we imagined



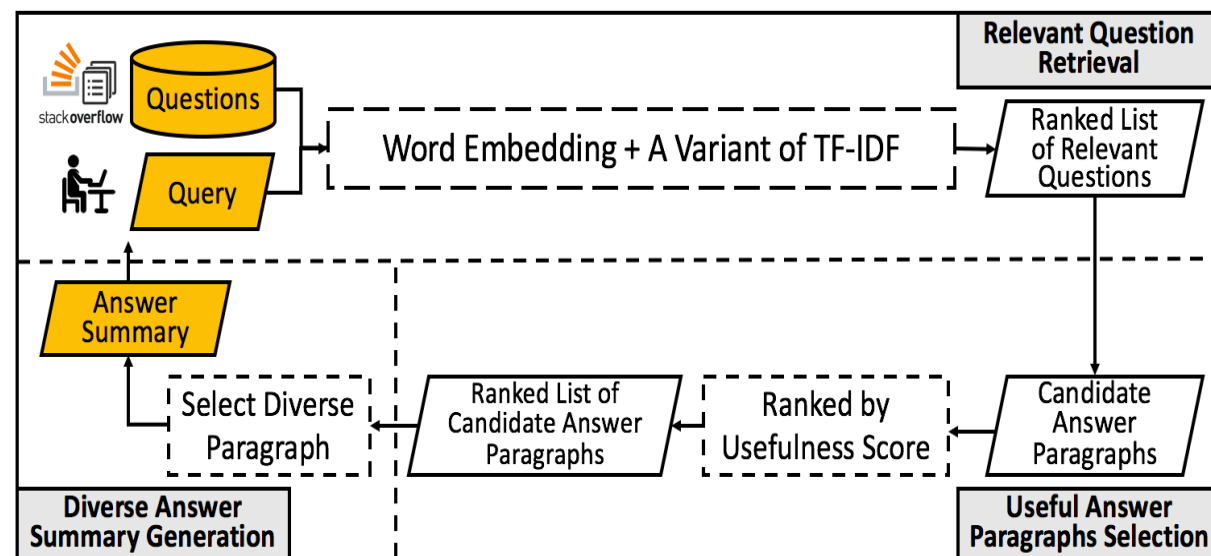
Survey based on 72 developers



Developers really need an effective technique to generate an answer summary with **diverse**, **relevant** and **high-quality** information from answer posts.

- “Some questions received **too many long answers**, and many of these answers have **redundant content...**”
- “I notice **even the best answers** in Stack Overflow often answer the questions **only in one aspect...** Thus, I expect the tool should **provide a diversity of potential answers...**”
- “Google will return **a number of “relevant” links** for a query, and I have to click into these links, and read a number of paragraphs. It is really **time-consuming**. A tool which generates potential answers can **save my time wasted on reading a lot of irrelevant content...**”

# Framework



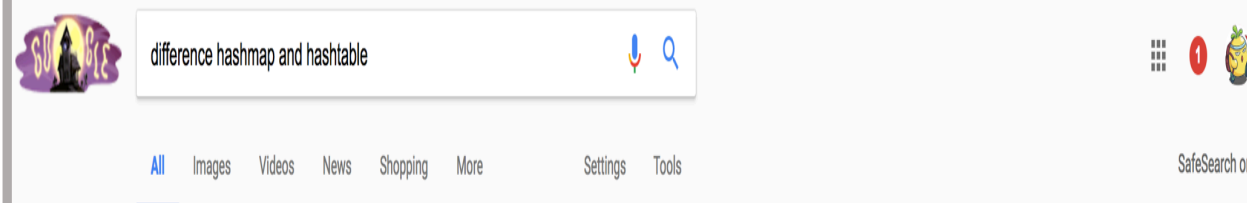
# RQ1 Evaluation on Overall Performance

	Relevance	Usefulness	Diversity
Our Approach	3.450	3.720	3.830
Baseline Google	3.440	3.480*	2.930***
Baseline Stack Overflow	2.576***	2.712***	2.305***

\*\*\*p<0.001, \*\*p<0.01, \*p<0.05

## Result:

- Best or highest-vote answers may not cover as diverse information as the developers need.
- It is worth reading more answer posts to summarize more complete information.



About 87,900 results (0.49 seconds)

java - Differences between HashMap and Hashtable? - Stack Overflow

<https://stackoverflow.com/questions/.../differences-between-hashmap-and-hashtable>

Sep 2, 2008 - The HashMap class is roughly equivalent to Hashtable, except that it is non synchronized and permits nulls. (HashMap allows null values as key and value whereas Hashtable doesn't allow nulls). HashMap does not guarantee that the order of the map will remain constant over time.

java - Differences between HashMap and Hashtable? - Stack Overflow

<https://stackoverflow.com/questions/.../differences-between-hashmap-and-hashtable/2534815...>

Sep 2, 2008 - The HashMap class is roughly equivalent to Hashtable, except that it permits nulls. (HashMap allows null values as key and value whereas Hashtable doesn't allow nulls).

## Answer Summary:

HashMap is non-synchronized whereas Hashtable is synchronized. Hashmap can store one key as null. Hashtable can't store null...

# Thanks and QA!

[bowenxu.2017@phdis.smu.edu.sg](mailto:bowenxu.2017@phdis.smu.edu.sg)

