Mining Duplicate Questions in Stack Overflow

What are duplicate questions?

Stack Overflow recommends that users search previous posts before asking a new question. This is to avoid asking a question that has already been asked and that may already have been answered. Stack Overflow also suggests links to questions whose title matches the new question. Despite these efforts, the site constantly faces duplicate questions; questions that are asked to solve the same problem. When two questions are duplicates of each other, one of them will be marked as a duplicate and go through the closing process. The other question will be marked as the master. Usually, the recent question will be closed as a duplicate of the older question because the older question typically contains the best answer.

Related Links:

Stack Exchange database schema: https://meta.stackexchange.com/questions/2677/database-schema-documentation-for-the-public-data-dump-and-sede

How to detect duplicate questions:

The postlinks.csv file contains the pair of questions that are duplicate to each other. You need to check the LinkTypeId to determine the duplicate questions.

PostLinks

- `Id` primary key
- `CreationDate` when the link was created
- `PostId` id of source post
- `RelatedPostId` id of target/related post
- `LinkTypeId` type of link

  1 = Linked (`PostId` contains a link to `RelatedPostId`)

  3 = Duplicate (`PostId` is a duplicate of `RelatedPostId`)

Questions:

1. **Determine the Number of Tags Per Question**
2. **Determine the Total Number of Unique Tags**
3. **Determine the top-25 Tags appearing frequently**
4. **Determine the nature of the distribution of top-500 tags**
5. **Determine the ratio of duplicate questions asked in each month**

For the given dataset, generate a line graph as follows. You need to consider the time between 2008 and 2018.

6. **Determine the percentage of duplicate questions associated with different tags**
   Each Stack Overflow question has some tags associated with it which is an identification of the topic of the question content. For each tag in your dataset, determine the number of duplicate questions associated with the tag over the total number of questions (calculate the percentage). Determine the top-20 tags based on the previously calculated numbers and then show the percentage in a bar chart.

7. **Determine the time required to close duplicate questions**
   You need to generate a bar chart

8. **Distribution of the reputation of users whose questions are closed as duplicates**

   The goal of this analysis is to verify whether duplicated questions are only asked by low-reputation users or not. If that is the case, we need to educate novice or low-reputation users to avoid creating duplicate questions. **You can generate a bar chart to show the number of users for different reputation categories. Use your own judgement to answer the question properly (e.g., break the reputation into different ranges).**

9. **Consider the first 500 tags and determine how many percentage of questions are been covered**
10. **Repeat Q.9 for first 5000 tags**