Related Questions Recommendation System for Stackoverflow Community

Shreya Bhatia
Stony Brook University
shbhatia@cs.stonybrook.edu

Arpita Sheth
Stony Brook University
arsheth@cs.stonybrook.edu

ABSTRACT

Stack Overflow is a language-independent collaboratively edited question and answer community site for programmers. It has accumulated large volumes of knowledge through the voluntary services of people across the globe.It leverages the knowledge and expertise of users to provide answers to technical questions. Majority of the times, similar queries to the current query being asked are already solved and it is useful to read the answers to such queries. We propose to recommend to the user, previously solved similar queries and posts which have the most relevant answers. We first employ similarity measures like Jaccard similarity and document retrieval techniques like TF-IDF. Later on, we propose advanced models based on Topic Modeling to map the semantic relationship between different posts.

Keywords

Stack Overflow, Question Recommendation, Similarity Measures, Jaccard Similarity, TF-IDF, Topic Modeling, LDA, Model Boosting

1. INTRODUCTION

The aim of this project is to recommend previously answered relevant questions that already have good answers. The motivation behind this is that the user can seek information faster if he is pointed in the correct direction for seeking answers. Usually, in a huge ecosystem like StackOverflow, getting the question correctly answered takes more time and this wait time can be reduced if we have relevant answers for contextually similar posts. We propose to employ similarity measures and text analysis approach to find the most relevant questions not just according to tags and title but also according to contextual relevance using Topic Modeling. We are employing multifarious models in the areas of Distance Similarity between documents, Text Analysis techniques and advanced models using Topic Modeling techniques, using features like Tags,Title, Question Body, View Count etc. We present an in depth discussion about the performance of various models in this paper.

2. PRIOR WORK

Work has been done in the area of analyzing the question response time in the stack overflow data and to identify those factors that have clear relation with response time. Prior work shows that Tags of a question have strong correlation with and are extremely indicative of response time[4]. In our approach we will be using tags information for predicting similar posts and also will be showing that tags, title tokens and context of the post plays a significant role in ranking the relevant posts. We will be using LDA for predicting topic of the posts and then assigning higher scores to similar topic scores. In [5], the author proposes similar questions using LDA model and inverted word indexing since computing similarity between all questions is a computationally intensive task. With LDA we will also be using tag information to get more relevant posts.

3. DATA COLLECTION AND ANALYSIS

For this project, we are using Stack Overflow Data [1] and we are focusing on Computer Science and Programming field Data for our model and analysis. The data is in Xml format segregated as Posts.xml, Users.xml, Badges.xml, PostLinks.xml etc. For our analysis purpose, we will be concentrating on Posts.xml and PostLinks.xml data. Posts.xml data contains the Questions and Answers data. PostLinks.xml contains the relevant and duplicate posts links suggested by StackOverflow which we will use for evaluation purposes. Posts data contains 463,533 posts which are segmented into 157,072 Questions and 306,451 Answers. The dataset contains following information:

Table 1: Data Attributes

Name	Attributes
Posts	Body, Id, Title, tags, View Count, etc.
PostLinks	Id,Creation date, Related Id, LinkType Id etc.

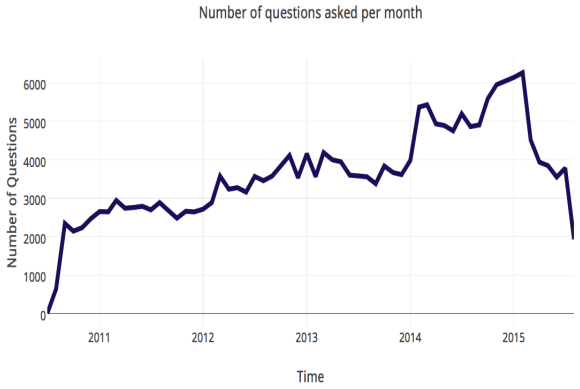


Figure 1: Number of Questions asked per month.

We have used PostLinks file for getting related posts and duplicates for our model evaluation. For the major part we have focused on the type 1 i.e Questions data in the posts file because we are recommending similar questions to the user. Fig 1. indicates the increase in number of questions per month since 2010 to 2015 May.

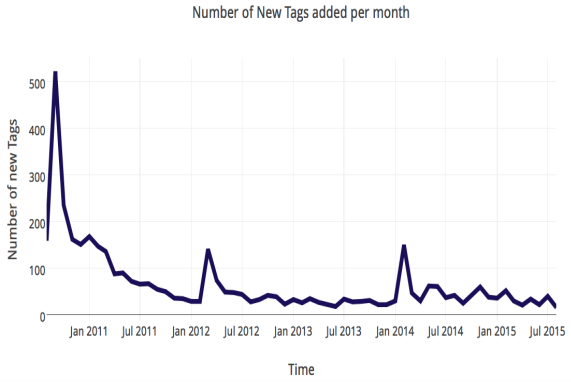


Figure 2: New tags added per month

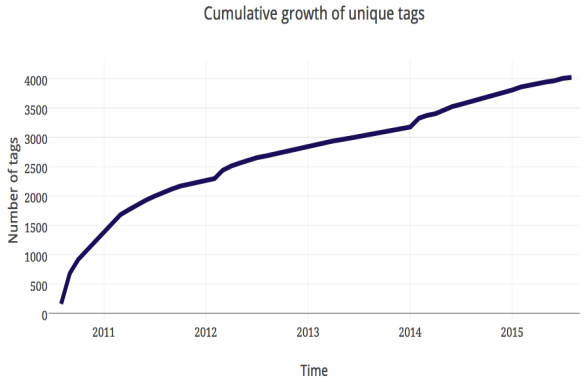


Figure 3: New Tags and Cumulative growth

For tag-based modeling, we also visualize the statistics of tag level data. [2] Fig 2 describes the addition of new tags to system every month and Fig 3 signifies the cumulative growth in tags.

4. MODELS AND EVALUATION

4.1 Evaluation Methodolgy

Duplicate Question Example:
Question Title: How to stream audio_out from PC to android using wifi?
Duplicate Question Title: Can I use my phone as an audio device via Wifi?
These type of questions are marked duplicate in the Stack Overflow and the data is available in the PostLinks file. We use these questions to evaluate our models with an assumption that duplicate questions are the most relevant ones for a new post. We measure the performance of various models according the distribution of similar questions returned by the model. A model is more efficient if it return greater percentage of duplicate records in first 10 results for each query.

4.2 Baseline Model

Jaccard similarity model was used as a baseline model for Title and Tags Similarity Matching. We processed the title and tag tokens and calculated Jaccard similarity of the combined set with the New Question. Topmost scores n posts are returned.
Jaccard Similarity:

$$JS(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

where J = set of tokens in the new question, S = set of tokens in each document of Stack Overflow.
We evaluate this model against the Duplicate Questions listed in the Stack Overflow data. Fig. 4 shows the distribution of relevant results returned by the model. We consider

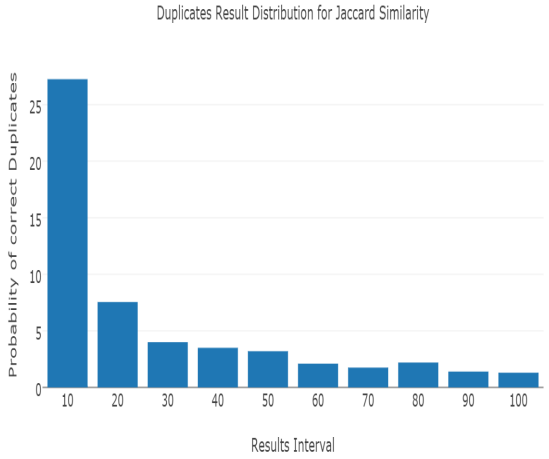


Figure 4: Performance Results for Baseline model

the accuracy of a model directly proportional to the percentage of queries for which duplicate posts were returned in the first 10 results. Jaccard Similarity model fetches duplicate questions for around 27.4% of the total questions in the first 10 results.

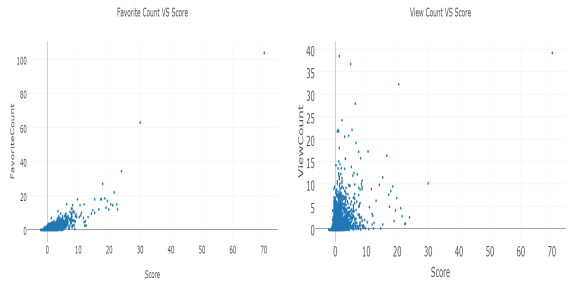
Table 2: Baseline Model Example

Question	How do I reset the launcher app?
Suggested Posts	Physical-keyboard driven app launcher?
	Can I reset the contacts app in ICS?
	Reset individual app settings
	Automatic factory reset and app installation
	How do you move an app shortcut between launcher screens?

4.3 Advanced Models

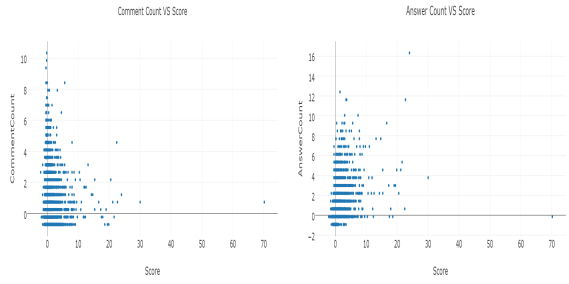
4.3.1 *Jaccard Similarity Score boosted with Additional features*

It was observed that apart from title and tags, other factors like View Count, Stack Overflow score and Favourite



(1)

Figure 5: Correlation of View and Favorite Count with score



(2)

Figure 6: Correlation of Comment and Answer Count with score

Count also play a significant role in determining the similarity between different posts. Fig. 5 and 6 show the correlation between different factors and their impact on the score.

From this analysis, it can be seen that the score of a post is directly impacted by Favourite Count and View Count, whereas Comment Count and Answer Count do not have any plausible impact on the score. This is radical in the sense that score of a post should be boosted by how many times it has been referred and liked by different users rather than the number of answers or comments it gets. Number of answers does not play an empirical role because posts with even 1 relevant answer as compared to a post with 20 substandard answers is more relevant for the task at hand. Thus 'View Count','Score' and 'Favourite Count' were used in this model for boosting the scores returned by Jaccard Similarity model. As depicted by Fig.7, this model retrieved duplicate posts for around 35.4% of the total questions in the first 10 results showing a marginal improvement over the traditional baseline model.

Table 3: Boosted JS Model Example

Question	How do I reset the launcher app?
Suggested Posts	Physical-keyboard driven app launcher?
	Automatic factory reset and app installation
	Reset individual app settings
	setting a default launcher
	How do you move an app shortcut between launcher screens?

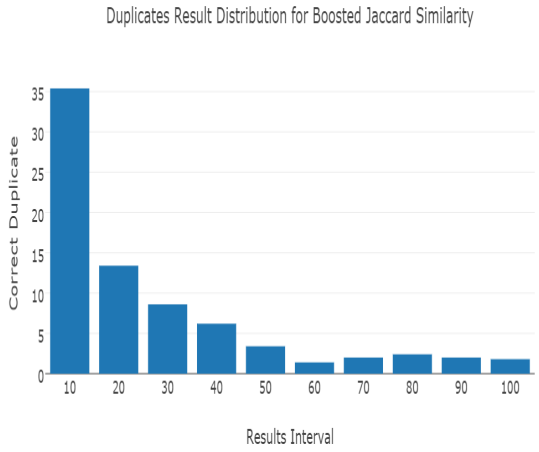


Figure 7: Performance Results for Boosted Jaccard Similarity model

4.3.2 *TF-IDF with Cosine Similarity*

In the above models, the body content of posts was not taken into consideration. The question body is a prominent aspect in recommending similar question and conveys the aesthetics of different posts. It denotes a better contextual relationship between different posts.
So we use an advanced model incorporating Term Frequency and Inverse Document Frequency combined with cosine similarity measures on the text body.Term frequency combined with Inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus and is used as a central tool in scoring and ranking a document's relevance given a user query. TF-IDF:

$$\mathrm{tf}(t,d) = 0.5 + \frac{0.5 \times f_{t,d}}{\max\{f_{t,d} : t \in d\}}$$
$$\mathrm{idf}(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$
$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \times \mathrm{idf}(t,D)$$

with N: total number of documents in the corpus N = |D|
$|\{d \in D : t \in d\}|$: number of documents where the term t appears (i.e., $\mathrm{tf}(t,d) \neq 0$).
If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to $1 + |\{d \in D : t \in d\}|$.

Cosine Similarity:

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

where A_i and B_i are components of vector A and B respectively.

The TF-IDF model applied to text corpus,titles and tags of all posts returns the posts marked duplicate for 63.2% of the total questions in the first 10 results and returns the duplicate posts for almost 80% of the queries in the first 20 results as shown by the distribution in Fig.8. This indicates

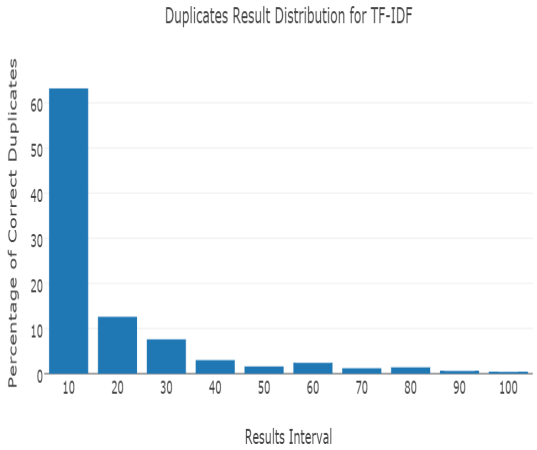


Figure 8: Performance Results for TF-IDF model

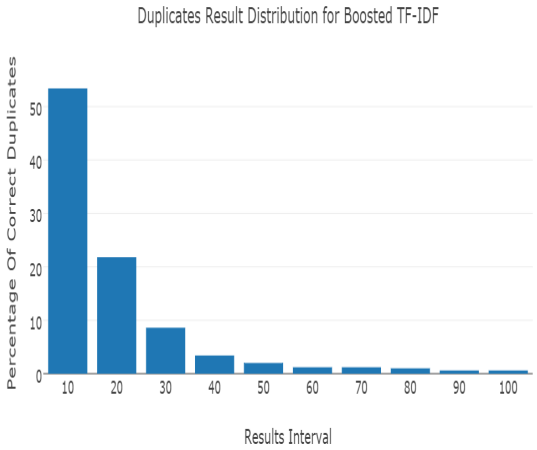


Figure 9: Performance Results of Combined TF-IDF model

that including the text body of a post in calculating similarity score gives a substantial improvement over using just the title and tags.

Table 4: TF-IDF Example

Question	How do I reset the launcher app?
Suggested Posts	I checked 'use as default' when choosing the default launcher, but now I can't start ADW.Launcher any more
	setting a default launcher
	Why aren't some launchers compatible with some devices?
	Automatic factory reset and app installation
	What is launcher.ebproductions.android.launcher?

4.3.3 *TF-IDF Score combined with Additional features*

As explained in the previous section, View Count and Favourite Count are important factors in boosting the relevance score of a post. So, we used these features in addition to TF-IDF scores. We observed that body content of a document and the other features of the post are not very inter-related.
Fig.9 shows that combining other features with the TD-IDF model does not achieve any substantial improvement over the TF-IDF model. This model is comparable to the above model when top 20 results are considered for each query but it yields an accuracy of 53.4% when top 10 results are examined.

4.3.4 *Topic Modelling - LDA*

It might be inefficient to use the entire corpus of posts for recommending similar questions in a huge ecosystem like Stack Overflow where the number of posts is inexhaustible. It is intuitive to extract a small set of words per post according to the contextual relevance. This calls forward the use of Topic Modelling where we extract topics as key features from each post. To extract topics, we used Latent Dirichlet

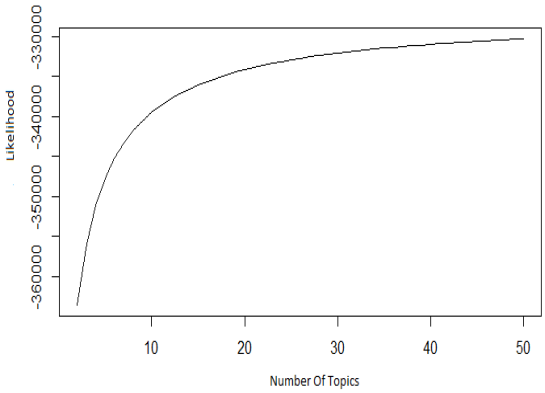


Figure 10: Maximum likelihood estimation for LDA

Allocation[3] proposed by Blei. combined with the probabilistic TF-IDF model on the corpus of text body,title and tags of the posts. We performed the maximum likelihood estimation to determine the number of topics for LDA. As it can be seen in the figure, it converges at 30 so we developed LDA model on the documents using 30 topics. Converting each query in a topic vector using LDA model developed on top of TF-IDF corpus and ranking it against all other posts gives a distribution with 31.2% accuracy.

4.3.5 *Supervised Learning*

Supervised learning is the machine learning task of inferring a function from labeled training data. A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. Stack Overflow provides the data for duplicate questions. We used Naive Bayes classifier for predicting if a post is duplicate of another post.
Following table shows the feature vector used for classification. Feature vectors of the two posts which are duplicate to each other were combined and used this combined vector

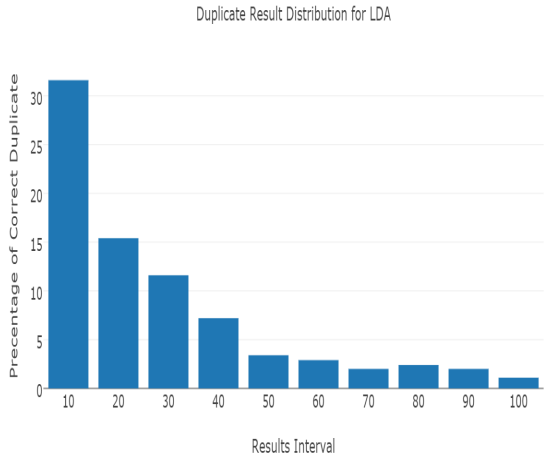


Figure 11: Performance Results for LDA

Table 5: Features for Supervised Learning

Features Used
Title Length,Tag Length,Body Length,View Count Favorite Count,Answer Count,Comment Count Score,OwnerUserId

for training the naive classifier model with the label 1, signifying a duplicate Question.
We generated pairs of non related Question and combined their feature vectors for training the model with label 0, signifying a non related post.
Data was divided into the ratio of 70-30% for training and testing purposes.
We tested our model on two sets of data:
1. Test Data
2. New post data set : Posts which has only Title_Length, Tag_Length, Body_Length and Owner UserId as its features.
Accuracy Of the model:

Table 6: Model Evaluation Results

	precision	recall	F-measure
Test Data	0.83	0.63	0.72
New Posts Data	0.58	0.54	0.56

5. FUTURE WORK

In our paper, we extracted topics and combined it with TF-IDF model for recommending similar posts. We did supervised learning using naÃŕve bayes classifier but just considered features like tag length, body length, view count etc. But it can further be extended by using textual features of the post, and combining it with feature vector from TF-IDF and LDA models. It would be interesting to observe the similar posts results using that model. These models can also be extended to recommend the most relevant questions to a user according to his technical interests. This helps in finding relevant questions for experts that they are more likely to answer.

6. CONCLUSIONS

The motivation for this paper is to come up with a methodology to recommend similar questions for a question in the Stack Overflow data. This has applications in tasks such as information retrieval, user recommendations, text analysis and many other problems which involve large amount of textual data. In this report, we have discussed our baseline approach using Distance Similarity measures and a more advanced model using TF - IDF and cosine similarity. Performing similarity operations on the entire text corpora might be inefficient for large text corpus like the Stack Overflow data so we extracted key features i.e topics from the posts and used these as a representation of the document for similarity measures.

7. REFERENCES

[1] *https://archive.org/details/stackexchange*. Stack Overflow public data.

[2] A. Barua, S. W. T., and Hassa, A. E. An analysis of topics and trends in stack overflow. *ACM*, 3 (2012).

[3] Blei, David M., A. N., and Jordan, M. Latent dirichlet allocation. *the Journal of machine Learning research 3*, 2 (2003), 993–1022.

[4] Vasudev Bhat, Adheesh Gokhale, R. J. J. P., and Akoglu, L. Analysis of question response time in stackoverflow. *ASONAM*, 1 (2014).

[5] Xiance Si, Edward Y. Chang, Z. G., and Sun., M. Confucius and its intelligent disciples: Integrating social with search. *Google Research*, 2 (2009).