Analyzing Stack Overflow Posts using LDA method

Ankit Kumar
Sridhar Chimalakonda
Akhila Sri Manasa Venigalla
cs20m004@iittp.ac.in
ch@iittp.ac.in
cs19d504@iittp.ac.in

ABSTRACT

Software developers the frequently solve development issues using of online platform like stack overflow(SO). it is most useful website for the developers for asking the doubts it is question and answer web forums. it is knowledge sharing platform. we are helping the users to helping the finding the question like tags works. For the developer benefit we classify posts into the three question categories. Furthermore, our approach can improve existing research on analyzing and finding topics which was discussed on SO posts.In this paper we found out three topic question categories which is-

- *Topic*1 - Web development
- *Topic*2 - Cloud containers
- *Topic*3 - Cloud computing services

KEYWORDS

Stack Overflow (SO), Latent Dirichlet Allocation (LDA), natural language processing

1 INTRODUCTION

Stack Overflow (SO) is one of the most frequently used websites with about 15M visits every day. so our study based on classifying questions on stackoverflow (SO) based on their keywords and arrived at almost similar taxonomies of categories. it automatic categorization by topic modeling using LDA. and also we manually categorize. by question category. For example, developers can use our models to find web development specific challenges by question category. the main contributions of this paper are - A taxonomy of 3 question categories. A manually labeled data set that maps 100 questions to 3 question categories. It automatically classify posts into the 3 question categories and we use NLP techniques - Latent Dirichlet Allocation (LDA) and Machine learning (ML) classifier Support Vector Classifier (SVC) to classify SO posts.
Behind the motivation is I read paper that title was -Learning with comments: An analysis of comments and community on Stack Overflow they categories the comments of stackoverflow post so I thought that I will experiment on stackoverflow question and I will find the topic which is depend on LDA techniques. I also read the paper title was - SOTagger - Towards Classifying Stack Overflow Posts through Contextual Tagging which was done by my mentor, in this paper they use LDA technique and they come up SOTagger application.
The contributions of this work are as follows:

- We investigate the distribution of questioners and answerers and investigate their behaviors.
- We employ topic modeling to assign topics to tens of thousands of questions from StackOverflow.
- we calculate Performance of our models that is classifing the stackoverflow SO posts into the 3 question categories?

The structure of this paper is as follows. In Section 2, we describe Dataset information on StackOverflow and topic modeling (Latent Dirichlet Allocation). In Section 3, we describe Experiments in the Empirical Study. in Section 4, we describe Results. in Section 5. we describe Limitation of the study. in Section 6. we describe Related Work in section 7. Finally, we describe conclusion with future work.

2 DATASET

Extract DataSet- I extract the first 500 Stack Overflow (SO) post because of with my IP address only allow 500 post per day. So I download each day 500 Stack Overflow (SO) post until I got 10k Stack Overflow (SO) post.For every call, the API would return 500 randomly elected Stack Overflow (SO) post or we can choose by date. We use these questions and their answers in our study. I scrape the some specific data so I fetch five data item from Stack Overflow (SO) posts.

- question Title
- question(full)
- tags
- Question(UserId)
- Answer(UserId)

after scraping data I combine the all csv file using *mergecsv_code.py* using this program it will combine into one csv file which name is *merge_data_stackoverflow.csv* file. I use StackAPI library for the data scraping. I build code using python language using of this I use the StackAPI library and scraping the data.
To perform categorization of SO posts we downloaded datascrapper.csv file which get by the file datascrapper.py. We considered a subset of this file that constituted 10K Stack Overflow posts under Body column that resulted in a dataset of 10K posts. and filtered out questions based on question column.
Data Pre-processing: Data present in question column in csv file was considered for preprocessing. We will perform the following steps:

Unpublished working draft. Not for distribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA

© 2021 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Unpublished working draft.
Not for distribution.

- Tokenization: Split the text into sentences and the sentences into words. Lowercase the words and remove punctuation.
- All stopwords are removed.
- We used Gensim here, use (deacc=True) to remove the punctuations.
- Stemming(lemmatization using spaCy) Words are stemmed words are reduced to their root form. The advantage of this is, we get to reduce the total number of unique words in the dictionary. As a result, the number of columns in the document-word matrix will be denser with lesser columns. From this we can expect better topics to be generated in the end.

Topic modeling

Step 1- Latent Dirichlet Allocation Model- We applied LDA to perform topic modeling. The LDA topic model algorithm requires a document word matrix as the main input. So I created this using using CountVectorizer. I have configured the CountVectorizer to consider words that has occurred at least 10 times, remove built-in english stopwords, convert all words to lowercase, and a word can contain numbers and alphabets of at least length 3 in order to be qualified as a word.Everything is ready to build a Latent Dirichlet Allocation (LDA) model. Let's initialise one and call $fit\_transform()$ to build the LDA model. We use here online variational Bayes algorithm for findout the best parametres. LDA model that categorizes given data into 4 topics.

Step 2- Naming Topics. We find the contextually useful keywords in each of the 3 topics, and used them to identify and name topics.

Step 3- Append Labels to Dataset. The LDA model provided us with a topic-document correlation matrix, where document refers to content of one post. This matrix contained probabilities of every identified topic for each document. We then classified posts in the Dataset into topics based on the dominant topic from correlation matrix which had the highest probability. **Step 4-** Prepare a Machine Learning model.

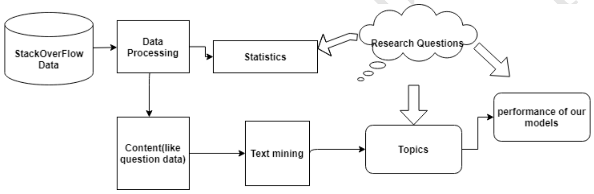


Figure 1: Empirical Study Framework

3 EXPERIMENTS IN THE EMPIRICAL STUDY

Research Question (RQ1)-analyzing the distribution of questions related to user in stack-overflow. We plot the graph of questioners in Figure 3. The graph shows the number of users that ask a given number of questions, and its y-axis is count. From the graph, we notice that most users only ask one question. Only some of the uses ask two or more questions. The number of users that ask questions reduces exponentially as we consider a higher number of posted questions. Only very less users ask more than 13 questions.

Experiment-

I took the data from *merge_data_stackoverflow.csv* file and from 4th column(*question_user_id*) we take that column all values and draw the graph using Microsoft Excel.

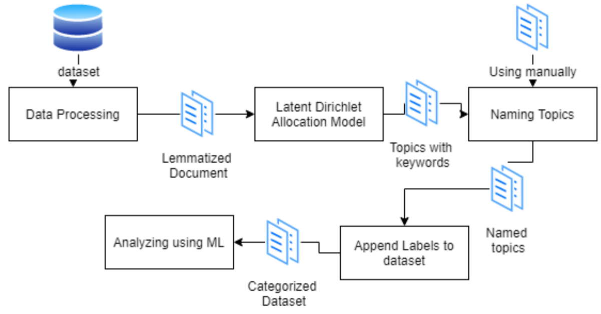


Figure 2: Overview of this research

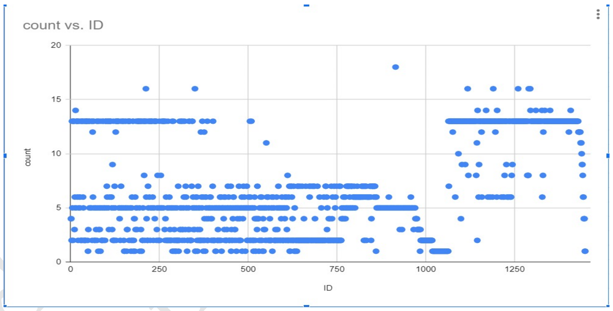


Figure 3: Count vs users question id

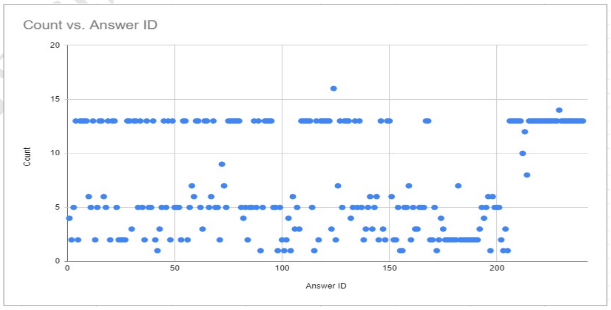


Figure 4: Count vs users answer id

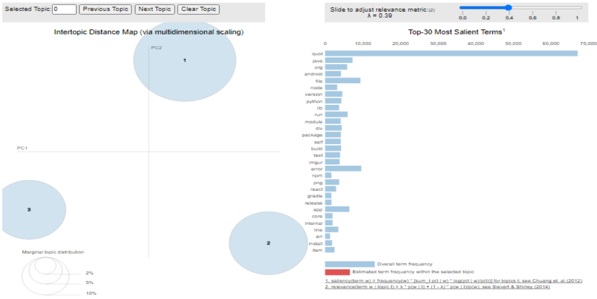

Selected Topic: 0
Previous Topic
Next Topic
Clear Topic
Intertopic Distance Map (via multidimensional scaling)
PC2
PC1
1
2
3
Marginal topic distribution
2%
5%
10%
Slide to adjust relevance metric:(2)
λ = 0.39
0.0
0.2
0.4
0.6
0.8
1
Top-30 Most Salient Terms¹
0
10,000
20,000
30,000
40,000
50,000
60,000
70,000
quot
java
org
android
file
node
version
python
lib
run
module
div
package
self
build
text
imgur
error
npm
png
react
gradle
release
app
core
internal
line
err
install
item
Overall term frequency
Estimated term frequency within the selected topic
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 5: Occurrence of keywords visualization

Unpublished working draft. Not for distribution.

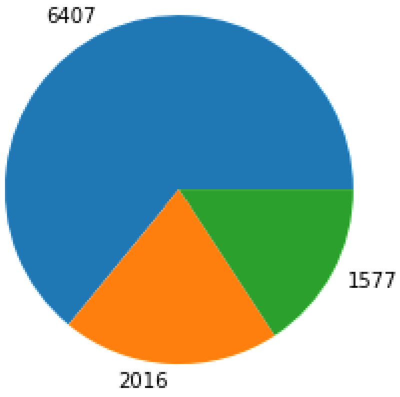
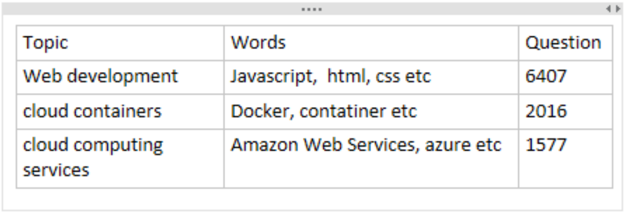


Figure 6: Statistics of the question which distributed into topics



Topic	Words	Question
Web development	Javascript, html, css etc	6407
cloud containers	Docker, contatiner etc	2016
cloud computing services	Amazon Web Services, azure etc	1577

Figure 7: Question categories

Research Question (RQ2)-analyzing the distribution of answers related to user in stack-overflow. We plot the graph of answers in Figure 4. The graph shows the number of users that answer a given number of questions and its y-axis is count. From the graph, we notice that most users only answer one question. Only about some of the users answer two or more questions. Very less users give answer more than 10 questions.

Experiment-

I took the data from *merge_data_stackoverflow.csv* file and from 5th column(*answer_user_id*) we take that column all values and draw the graph using Microsoft Excel.

Research Question (RQ3)- What is the performance of our models to classify SO posts into the 3 question categories? A model with higher log-likelihood and lower perplexity is considered to be good. using this we can make better version of the LDA models and we can improve the model using different function, here we are going to find out the performance of this model.

Research Question (RQ4) What are the topics of question categories of stackoverflow (SO) posts? here I run the our LDA model on given data and it will give us 4 topics using of then we manually read the keywords and decide what topic name it should be and we build the model like if we give the question data in text variable it will give us the probability of all four topics which will get the higher probability then we take the high probability topic and assign that topic to the particular question. and I create also which is automatically assign the topic to each question.

4 RESULTS

Explain the results of each research questions.

Research Question (RQ1) result-

analysing the distribution of questions related to user in stack-overflow. The result shows that there are few regular questioners on Stack Overflow. the behind the reason is some questions already asked. or they were using other question answer websites.

Research Question (RQ2) result-

analysing the distribution of questions related to user in stack-overflow. The result shows that there are few regular questioners on Stack Overflow. the behind the reason is some answer already given. or they were using other question answer websites.

Research Question (RQ3) result- The results show that our models can classify SO posts into the 3 question categories with performance of 1.Log Likelihood Score: -1584582.5417850607, 2.Model Perplexity: 743.7041895978797 we can improve this by the taking the large datasets.

Research Question (RQ4) result-

so our result is, fisrt we ar egoing to applying the topic modeling technique LDA on the on stackoverflow posts. after we set the 3 number of topics, our outputs is giving 3 topics- each topic, it has a set of words sorted in terms of their likelihood of belonging to the topic. then our model not generate meaning full label for each topics then we manually read the words of each topics and assign label to the topic in manual label file which data set length is number is row is 100. so from of this in the questions categories table we assign and give the Figure 7 (naming topics). We assign only one topic with the highest probability to a question based on the topic probabilities assigned by LDA to the question and count how many questions belong to each topic shown in the in pie chart we were Figure 6- (distributed the question) provide in three categories and provide some statistics. Figure 5 occurrence of keyword in particular topics we can se which have more repetition.

5 LIMITATIONS OF THE STUDY

Explain the limitations about the study talking about technical challenge related to experiments, data set creation. You should also mention any assumption you that you consider while conducting experiments.

Limitation is that I apply on my model only 10k stackoverflow posts instead of I have to take more large dataset. And I have to apply more better model in comparison of LDA techniques(unsupervised learning), we could also try supervised learning may it will give us high accuracy. Limitation of this model

- Uncorrelated topics (Dirichlet topic distribution cannot capture correlations)
- Non-hierarchical (in data-limited regimes hierarchical models allow sharing of data)
- Static (no evolution of topics over time)
- Bag of words (assumes words are exchangeable, sentence structure is not modeled)
- Unsupervised (sometimes weak supervision is desirable, e.g. in sentiment analysis)

I was facing lots of difficulties like scrapping the data it is very difficult to scraping the more than 500 posts so instead of this I

Unpublished working draft. Not for distribution.

scrape the daily 500 posts till it reach to 10k posts. I read documentation of stackAPI library and I manually scrape the data and labeling them. and for merging the all posts in the one csv file so I build the python code with this help I could merge the all csv file into one csv file. for my model coding part there is two library which they provide to build LDA model one is gensim and other one is scikit-learn I try both of them then I choose scikit-learn library for my LDA model. And installing spacy also difficult I try so many command installing the spacy, while data cleaning I also read the text that what I have to take/leave. From this I learn NLP techniques and ML techniques. So many error I face while building the LDA model.

6 RELATED WORK

Automatically Classifying Posts Into Question Categories on Stack Overflow after reading this paper they done manually created data set and of 500 posts and classified into seven categories Using this data set, they apply machine learning algorithms. after getting best model they show that his models can classify posts into then correct question category.

7 CONCLUSION AND FUTURE WORK

We analyzing here is that questions posted by users could be grouped into 3 categories based on the topic modeling technique. In this research, we analyzing on StackOverflow posts and categories each question into one of the three topics. recently analyze 200 questions manually and label them into 3 categories. lots of users did not posts any question and answers. very few users give the answer on stackoverflow posts and very few users post the question on stackoverflow.

- $Topic1$ - Web development
- $Topic2$ - Cloud containers
- $Topic3$ - Cloud computing services

for future work- My research on future work is the we can extend this study by finding the more questions from StackOverflow(SO) and from other question and answer web sites. we also thinking that we can applying the various topic modeling techniques. we can also find the topic from comment.

8 ARTIFACTS

GitHub link of my Empirical study.
https://github.com/ashupipalia/Analyzing-Stack-Overflow-Posts

9 REFERENCES

1.https://ksiresearch.org/seke/seke19paper/seke19paper_67.pdf
2.https://dl.acm.org/doi/abs/10.1145/3196321.3196333
3.https://www.researchgate.net/publication/262239821_An_empirical_study_on_developer_interactions_in_StackOverflow
4.https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf
5.https://core.ac.uk/download/pdf/286030414.pdf

REFERENCES

Unpublished working draft.
Not for distribution.