INFORMATICS OF STACK OVERFLOW DATA

Under Guidance of:

Prof. Emily Liu

By:

Yash Mahendra

Yash Shah

Hardeep Singh

Swapnil Panchal

Introduction

Stack Overflow is a Knowledge Market Website founded in 2008. It is the largest, most trusted online community for developers to learn, share their knowledge and build their careers. It is visited by more than 50 million professional and aspiring programmers each month to help solve coding problems, develop new skills and find job opportunities.

Stack Overflow is a question and answer site for professional and enthusiast programmers. There is no chit – chat done on this site neither it is a discussion forum, it is all about getting answers.

Motivation and Objectives

All four of us have either computers or information technology as our background. Being engineers we have frequently seek answers to questions from Stack Overflow. Whenever, we searched for answers on Stack Overflow there were some questions that did not have answers, or the solution was not to the point or 100% correct. This motivated us to do some analysis on this website.

The site has more than 14M questions. Thus, to limit the scope of the project we selected only 20000 Python tagged questions. The main aim of our project is to cluster similar type of questions and propose answer for unanswered or new questions based on the cluster they belong. The secondary goal of our project is to get Top users for each type of clusters based on their frequency and reputation score. This will help individuals ask questions, directly to these users if they are unanswered for a long time.

Related Work

There is already some analysis done on Stack Overflow questions by:

Miltiadas Allamanis, Charles Sutton, School of Informatics, University of Edinburgh, Edinburgh, UK named:

Why, When and What: Analyzing Stack Overflow Questions by Topic, Type and Code.

In their paper they perform 3 steps:

1. They categorize questions according to concepts.
2. Move beyond the first analysis by categorizing questions by type.
3. They connect question concepts and types to perform analyses like: “What types of questions are mostly asked about the Date object in Java?”

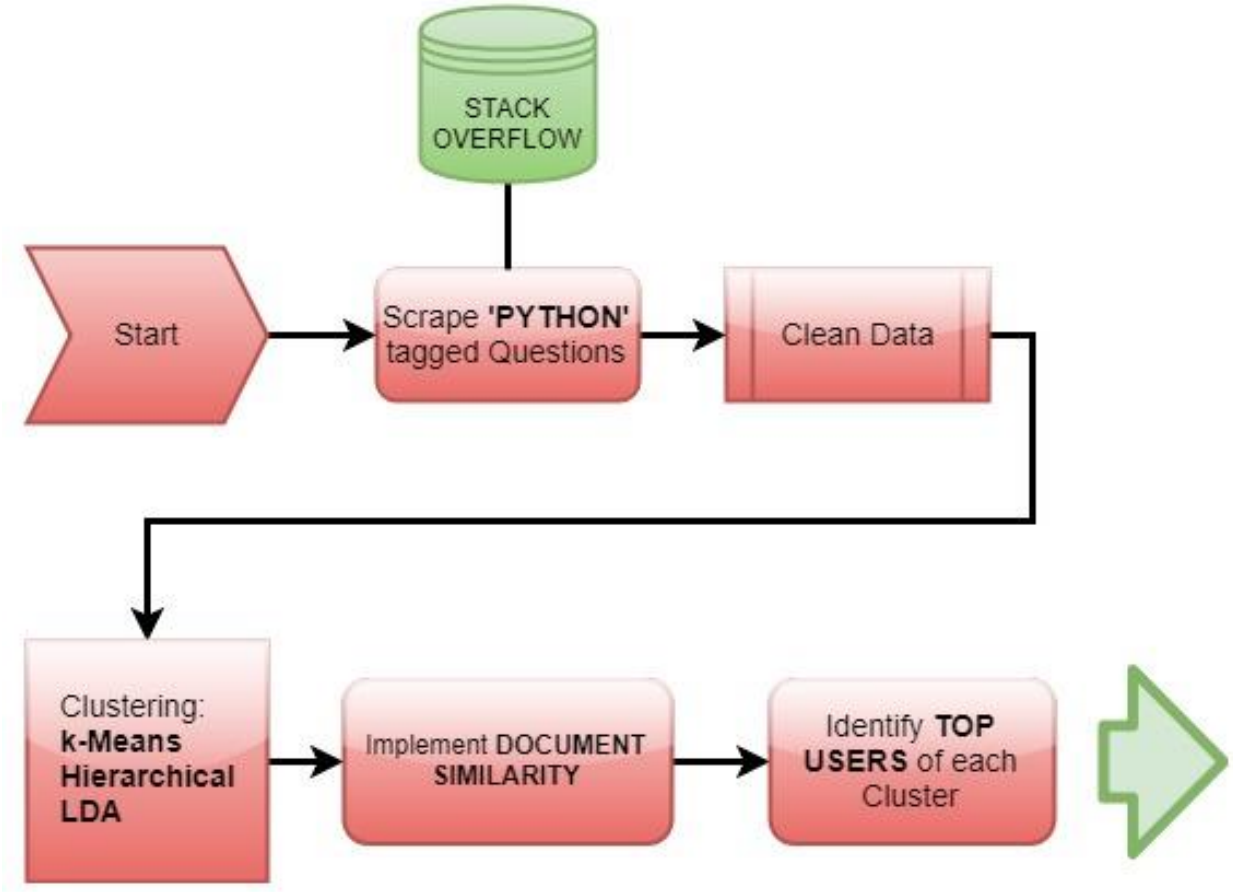
The paper uses Topic Modelling for their analyses because it helps them associate programming concepts and identifiers with particular types of questions such as, “how to perform encoding”.

What’s New

In addition to the existing research, the add – ons to our project are:

1. To propose questions related to the new question asked.
2. To evaluate user expertise based on the frequency and reputation score.
3. To determine the unsolved questions and forward it to top users.

Methodology



1. Scraping data (Question and Answers), posted by users from stack overflow related to python which consists of:
    a. Question ID
    b. Question Description
    c. User
    d. Reputation Score
    e. Gold Badge Count

f. Silver Badge Count
g. Tags
h. Answer
i. Votes

2. Then cleaning data by stripping, removing html tags, etc.
3. Applying clustering algorithms Hierarchical clustering, K-means, LDA (Latent Dirichlet Allocation) on Question Description.
4. Implementing code to help find similar clusters for any new question submitted.
5. Finding top users in each cluster.

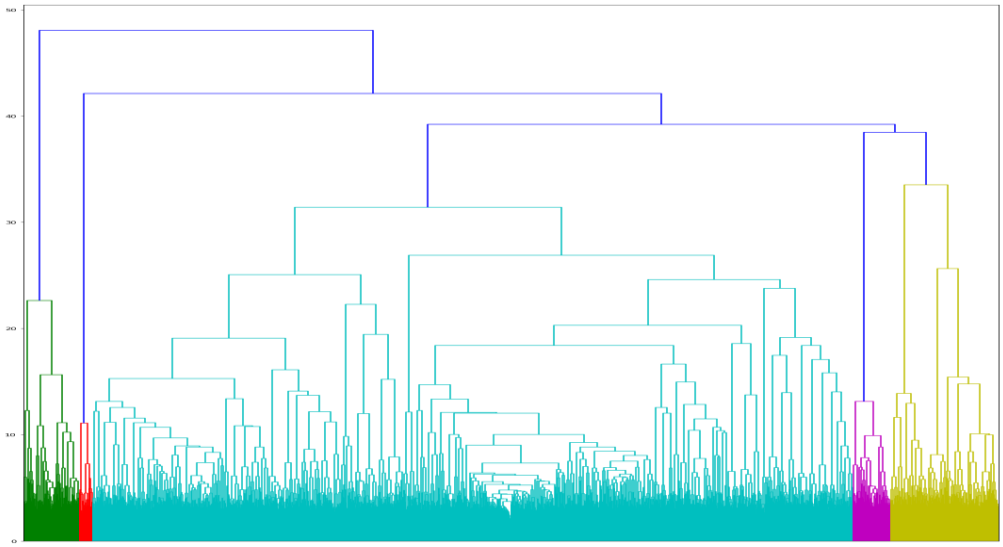
Clustering

We are using the following steps for each cluster model.

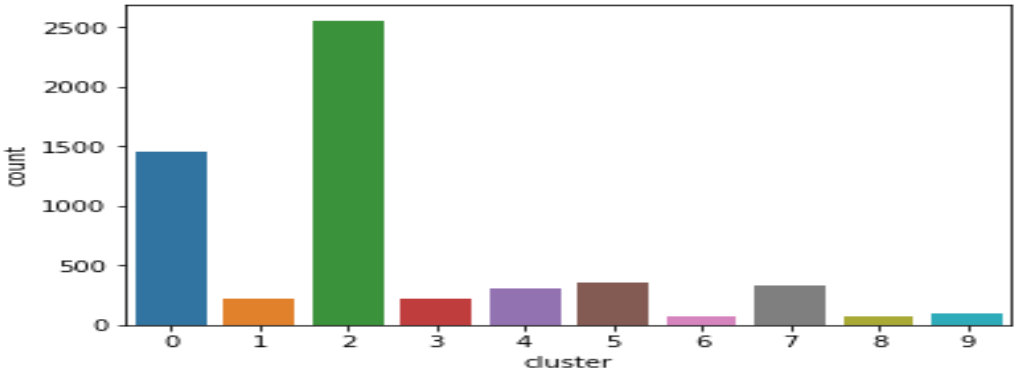
1. Load the data.
2. Calculate tf_idf of text data using tfidf vectorizer.
3. Run the model.
4. Check the cluster distribution and top words.
5. Evaluate the model using internal and external evaluation.

Hierarchical clustering:

We first used the Hierarchical clustering to understand how many clusters can be formed for further evaluation in K-means as we need to define the value of K to implement K-means. We used this clustering only on the question description that contains answers.



We also found the data to be unbalanced with most of the values concentrated in cluster 0 and 2.

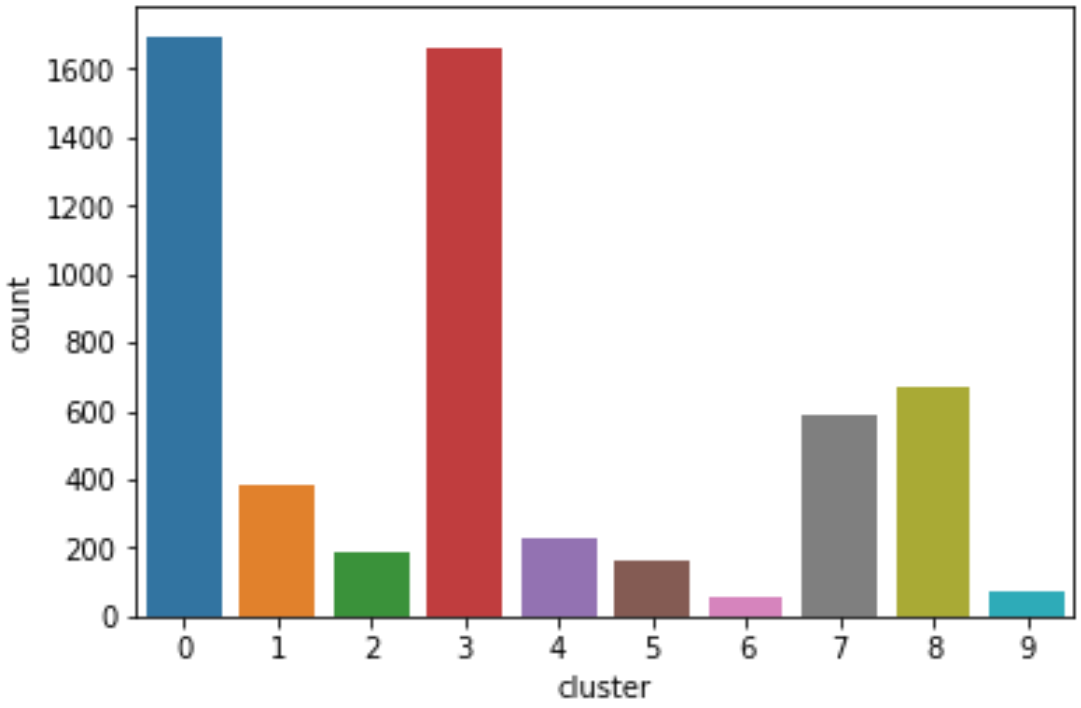


Cutting the dendogram at level 20 we were able to see 10 clusters being formed. Using K=10 can be used to implement K-means.

K Means

Then we used K-Means algorithm, to cluster Question Description based on euclidean distance. By setting the value of k = 10, we are able to get a better output from Hierarchical clustering.

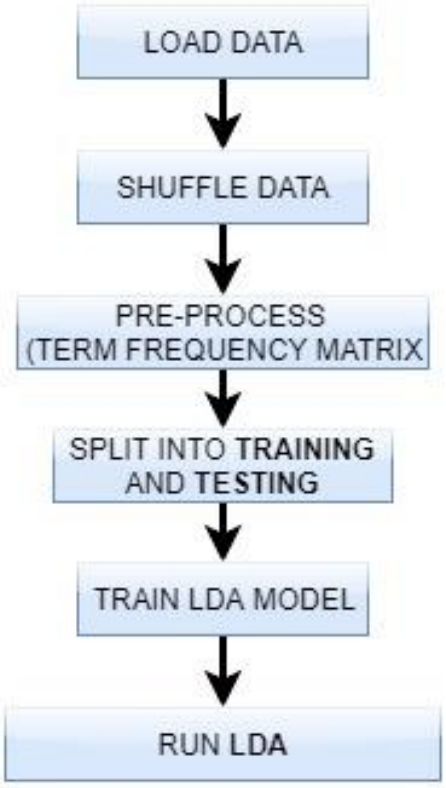
But the clusters formed were imbalanced like Hierarchical clustering. Cluster 0 and 3 had the majority of questions and rest of the had few questions, based on count plot below.



LDA(Latent Dirichlet Allocation):

Latent Dirichlet allocation (LDA) is a particularly popular method for fitting a topic model. It treats each document as a mixture of topics, and each topic as a mixture of words.

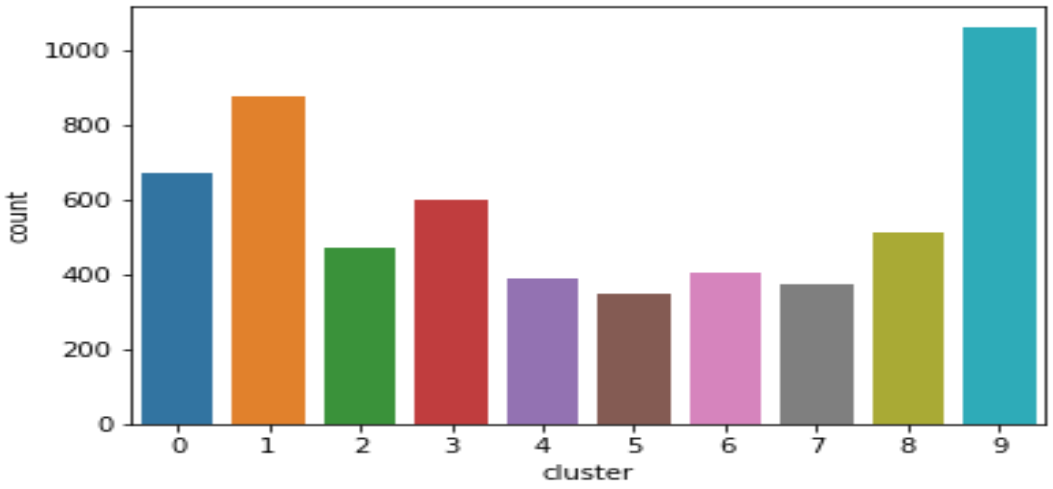
The steps we followed for LDA:



Using the following steps with iterations set to 50 to reduce perplexity and make the model more stable.

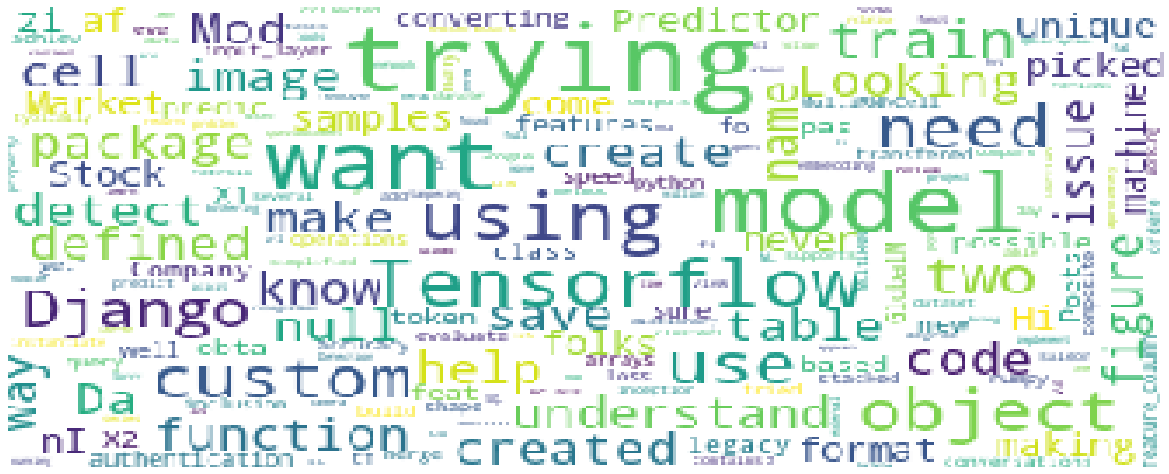
For 1st iteration the perplexity was 1384. And it was reduced to 902 in 47th iteration.

Using the LDA we found the data to be better distributed than the K-means model



Also figure below will help us understand with the help of Wordcloud of top 200 words that are included in some of the clusters formed.

Word Cloud of Cluster 5 – PANDAS



Word Cloud of Cluster 6 – MACHINE LEARNING

Word Cloud of Cluster 9 – MODULE IMPORT & ERROR

The labels of identified clusters:

Sr no.	Cluster Name	Cluster No.
1.	Python	0
2.	Python 3.x	1
3.	Database	2
4.	Web Scraping	3
5.	File Handling	4
6.	Pandas	5
7.	Machine Learning	6
8.	Class	7
9.	Data Structure	8
10.	Module, Import and Error	9

Evaluation of Clusters:

Clusters are evaluated using:

- **Internal Evaluation**
  - **Silhouette Co-efficient :** It is defined for each sample and is composed of two scores:

    a: The mean distance between a sample and all other points in the same class.

    b: The mean distance between a sample and all other points in the next nearest cluster.

  - **Calinski-Harabaz Index:** For k, clusters, the Calinski-Harabaz score s is given as the ratio of the between-clusters dispersion mean and the within-cluster dispersion.

$$s = \frac{b - a}{max(a, b)}$$

Following are the results of internal evaluation.

Hierarchical Clustering

- metrics.silhouette_score(data2D, clusters) = -0.0083045139386758639
- metrics.calinski_harabaz_score(data2D, clusters) = 719.41549160766886

k-Means Clustering

- metrics.silhouette_score(data2D, cluster_labels) : 0.36023708541084698
- metrics.calinski_harabaz_score(data2D, cluster_labels) : 4202.5395393652852

LDA Clustering

- metrics.silhouette_score(data2D, clusters) = -0.13033888139828545
- metrics.calinski_harabaz_score(data2D, clusters) = 311.55332614057988

Figure

From this we found that K-means have the highest silhouette score and Calinski Harabaz score among the three.

- **External Evaluation**
  - Obtain 'GROUND TRUTH'
  - Randomly label data (manually) if labelling absent
  - Assign each cluster to a 'True' class by majority vote rule
  - Calculate 'Precision' and 'Recall'

Below is the table containing 1000 Rows of Random data. We have manually labelled the target, based on Tags. For example, the fifth tag is "web-scraping beautifulsoup html-lists". We can relate that tag to our cluster "Web Scraping".

Tags	Labels
python documentation read-the-docs	Python
python	Python
python text nlp data-cleaning	Machine Learning
python	Python
python web-scraping beautifulsoup html-lists	Web Scraping
python heap priority-queue	Data Structure
python sqlalchemy	Web Scraping
python arrays numpy	Data Structure
python python-2.7 sorting tuples unique	Data Structure

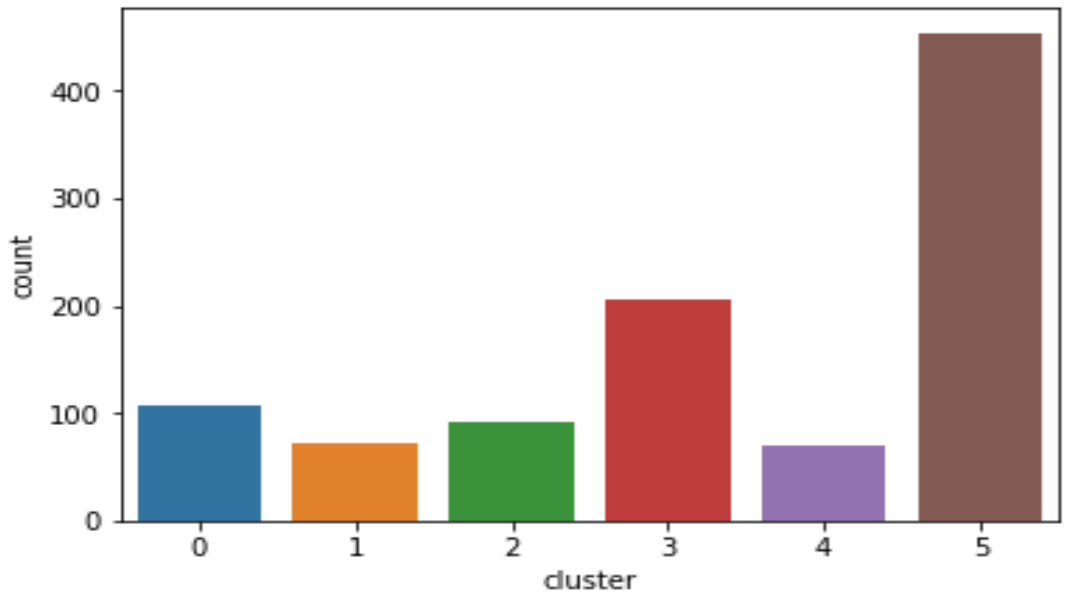
Figure

Below are the results of K-means crosstab and countplot barchart.

- As we can see from the cross tab and Figure 15 count plot, we cannot identify topics for each cluster.
- Cluster 5 contains maximum number of questions(Figure 16) and the rest of the clusters have few questions.
- Since the clusters assigned are quite imbalance and there is no majority , we are not able to assign questions to clusters.
- Thus, we are not able to calculate precision or classification report for k-Means.

actual_class	Class	Data Structure	File Handling	Machine Learning	Python	Web Scraping
cluster						
0	21	5	10	2	53	17
1	2	21	3	9	19	17
2	11	51	2	20	2	6
3	13	14	24	28	105	21
4	13	14	2	25	10	6
5	50	31	103	107	53	110

Figure



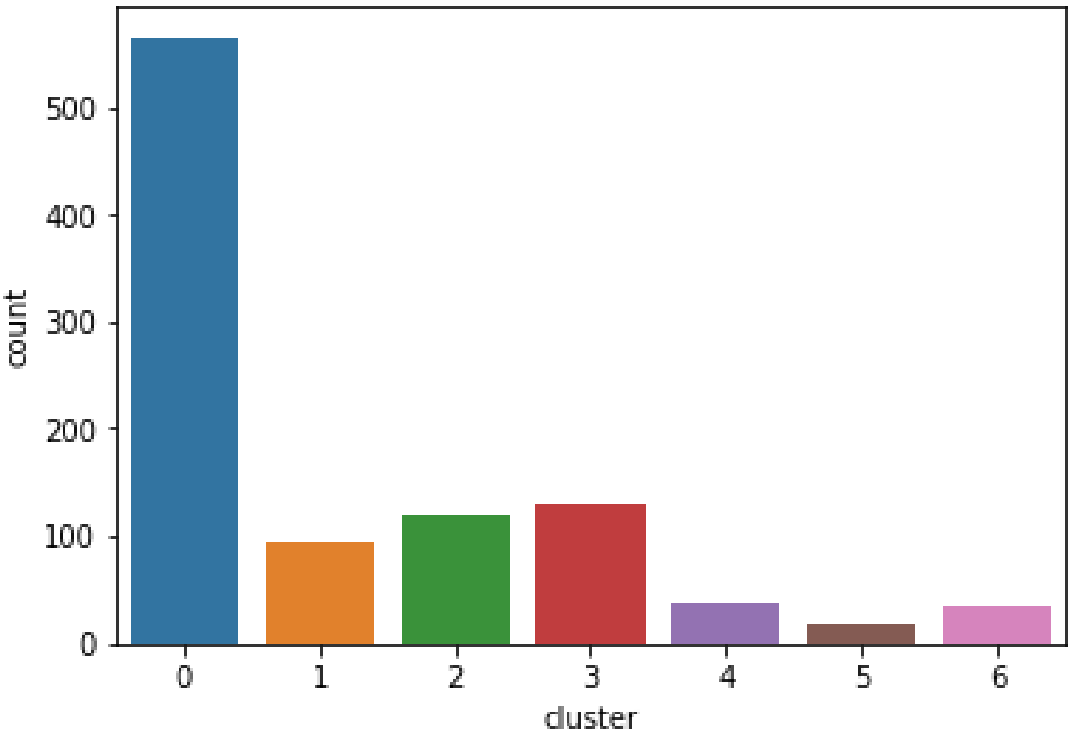
Figure

Below are the results of Hierarchical Clustering crosstab and countplot - barchart.

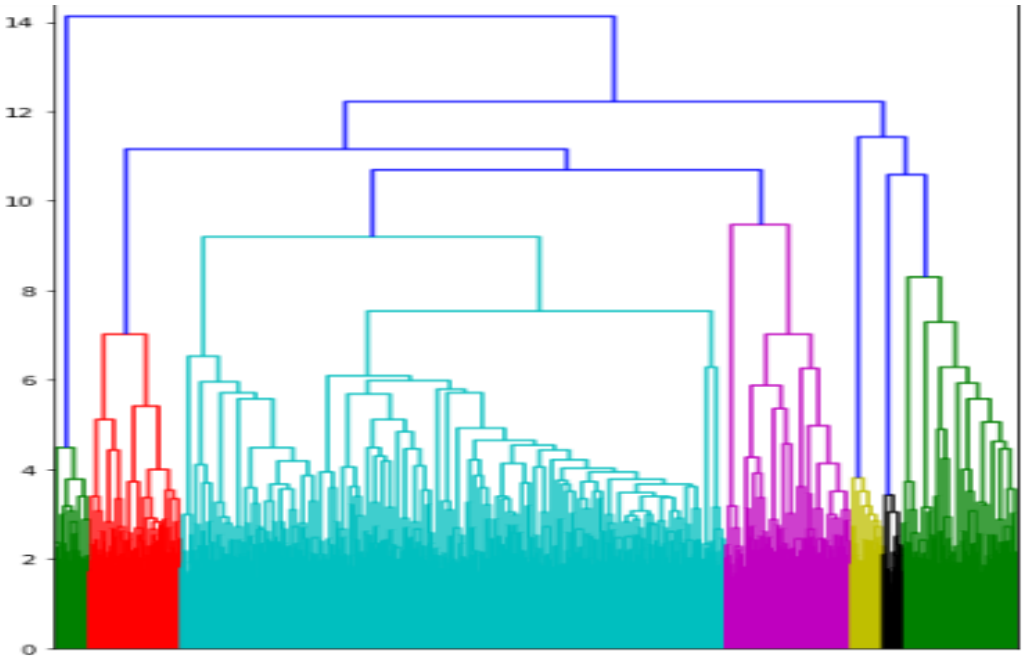
- From the cross tab and count plot for hierarchical clustering, we cannot identify topics for each cluster.
- Cluster 1 contains maximum number of questions(Figure 18) and the rest of the clusters have few questions.
- Since the clusters assigned are quite imbalance and there is no majority , we are not able to assign questions to clusters.
- Thus, we are not able to calculate precision or classification report for Hierarchical Clustering.

actual_class cluster	Class	Data Structure	File Handling	Machine Learning	Python	Web Scraping
0	61	47	100	95	153	109
1	12	6	10	7	44	16
2	25	54	13	15	5	8
3	7	8	20	37	24	34
4	0	15	1	5	8	8
5	0	2	0	15	2	0
6	5	4	0	17	6	2

Figure



Figure



Figure

Below are the results of LDA crosstab and countplot - barchart.

Perplexity reduces from approx.. 570 to 424.13

Iterations:

```
iteration: 1 of max_iter: 50, perplexity: 569.8423
iteration: 2 of max_iter: 50, perplexity: 523.2481
iteration: 3 of max_iter: 50, perplexity: 495.2135
iteration: 4 of max_iter: 50, perplexity: 478.4297
iteration: 5 of max_iter: 50, perplexity: 467.5348
iteration: 6 of max_iter: 50, perplexity: 459.2701
iteration: 7 of max_iter: 50, perplexity: 453.0113
iteration: 8 of max_iter: 50, perplexity: 447.6762
iteration: 9 of max_iter: 50, perplexity: 442.6869
iteration: 10 of max_iter: 50, perplexity: 438.1928
iteration: 11 of max_iter: 50, perplexity: 435.6378
iteration: 12 of max_iter: 50, perplexity: 433.5272
iteration: 13 of max_iter: 50, perplexity: 431.4718
iteration: 14 of max_iter: 50, perplexity: 429.7839
iteration: 15 of max_iter: 50, perplexity: 428.4748
iteration: 16 of max_iter: 50, perplexity: 427.6282
iteration: 17 of max_iter: 50, perplexity: 426.9317
iteration: 18 of max_iter: 50, perplexity: 426.3018
iteration: 19 of max_iter: 50, perplexity: 425.6146
iteration: 20 of max_iter: 50, perplexity: 425.1461
iteration: 21 of max_iter: 50, perplexity: 424.8318
iteration: 22 of max_iter: 50, perplexity: 424.6201
iteration: 23 of max_iter: 50, perplexity: 424.3744
iteration: 24 of max_iter: 50, perplexity: 424.2095
iteration: 25 of max_iter: 50, perplexity: 424.1360
```

Figure 1 Perplexity

actual_class	Class	Data Structure	File Handling	Machine Learning	Python	Web Scraping
cluster						
0	103	10	5	5	2	3
1	1	3	2	8	19	145
2	0	2	124	5	3	12
3	3	11	1	157	7	4
4	1	5	10	6	209	9
5	2	105	2	10	2	4

Figure: Crosstab LDA

```
<matplotlib.axes._subplots.AxesSubplot at 0x20799764390>
```

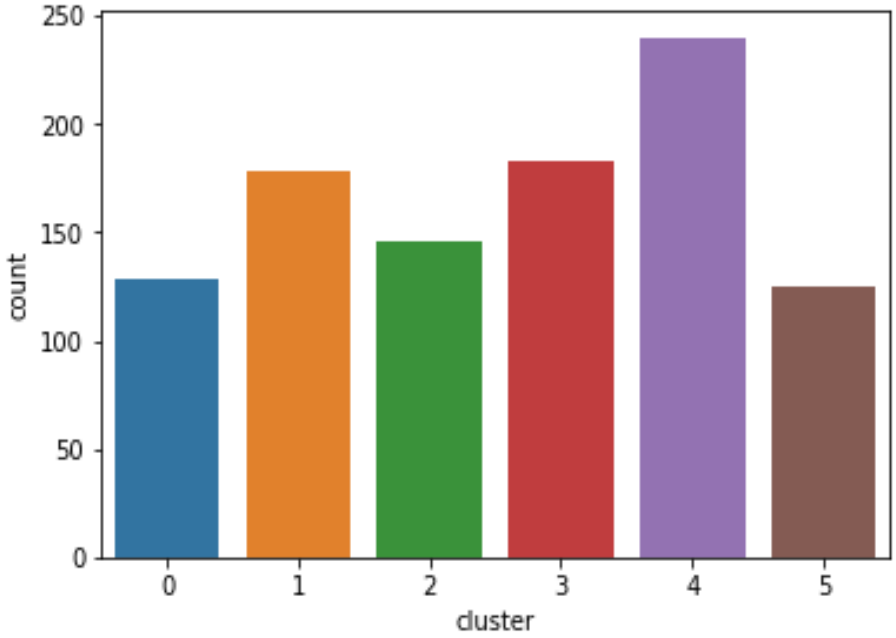


Figure: Bar chart LDA

Figure below shows the report of LDA approach:

	precision	recall	f1-score	support
Class	0.80	0.94	0.87	110
Data Structure	0.84	0.77	0.80	136
File Handling	0.85	0.86	0.86	144
Machine Learning	0.86	0.82	0.84	191
Python	0.87	0.86	0.87	242
Web Scraping	0.81	0.82	0.82	177
avg / total	0.84	0.84	0.84	1000

Figure 2 Report LD

Below is the finding from internal and external evaluation:

- From Internal Evaluation, the k-Mean model has the best value for both the co-efficient.
- But from External Evaluation, the LDA model is the most efficient approach of clustering data.
- Since External Evaluation is much more realistic and efficient in many real-world scenarios as it does not refer to any assumed references from outside which is not always feasible to obtain.
- Other Advantages of using LDA for text-clustering,
    - **Fast**
    - **Highly Modular (Easily Extended)**
    - **Shorter and Cleaner Code**
    - **Has Interpretable Topics**
    - **Less Expensive**
    - **Get Free Categories with any dataset**

Analysis of Experiment Results

1. What part of our methodology worked or didn't work?

   The methodology that we used for our project is Clustering. We tried Hierarchical Clustering, K Means Clustering and Latent Dirichlet Allocation (LDA) on our dataset. We evaluated the clusters using internal and external evaluation.

   The only thing that we tried and didn't work for our methodology is that when a new question is asked, instead of LDA directly predicting the cluster, we had to do document similarity to check, in which cluster does the new question belongs. This seem to be a expensive way of finding the cluster for the new question than LDA predicting it.

2. Why did your methodology work or didn't work?

   Our methodology worked because given the objective of our project, this seem to be the only apt method. Evaluating the clusters helped us know that for our dataset clustering using LDA was better than K means and Hierarchical Clustering. Clustering helped us in fragmenting the dataset into chunks. This further allowed document similarity, although not the best and cheap way to check the cluster associated to the new question, these small chunks helped us do it faster than expected.

3. How to improve?

   We will try to improve the time and space complexity of our project by making LDA predict the clusters instead of using document similarity for it.

4. How to utilize your results? What business insights can be derived from your analysis?

The results generated by us in the project can help Stack Overflow generate solutions to the unanswered questions by asking those questions to the top 5 users generated by us based on the frequency and reputation score.

Stack Overflow can also use the idea of clustering similar type of questions and whenever a new question is asked, they can identify the cluster to which it belongs and provide an approximate answer based on the cluster solutions.

Conclusion

To conclude our project, we would like to show you the demo of what we have analyzed:

1. New Question=**"Specific type of webscraping [on hold]"**
2. [0.02365704809214167, 0.02273894798279715, 0.024369553824805695, 0.018350023480873275, 0.04019069545628393, 0.027863872698812842, 0.035337866765671765, 0.025909023763576768, 0.050765908090397716, 0.024345085332260548]'-**Mean Similarity Values**
3. The Cluster number 8

   Mean Similarity Value- 0.050765908090397716

4. Top 10 Questions

	Index	Similarity Values
63	63	0.470746
101	101	0.416636
70	70	0.412094
87	87	0.402996
43	43	0.375636
34	34	0.331571
83	83	0.308178
85	85	0.286638
30	30	0.273958
40	40	0.241262

	Question Id	Votes	Answer Count	Views	Question	QDescription	User	Reputation Score	Gold Badge Count	Silver Badge Count	Bronze Badge Count	Tags	QDescription length	cluster
0	49303054	-3	1	21	Retrieve all the values of one column of Datas...	class Publisher (models.Model):\n name = mod...	RISHABH BANSAL	1	0	0	6	python django django-models	204	8.0
1	49278902	-1	1	43	optimize nested loop in python [on hold]	I am writing 2-opt algorithm and want to optim...	Tomonaga	4	0	0	1	python nested-loops	204	8.0
2	49297302	-4	2	33	Tkinter progress bar [on hold]	I looked through the internet and couldn't rea...	Ben	3	0	0	2	python tkinter python-multithreading	203	8.0
7	49287997	0	2	51	What Have I Done Wrong? | Python Code Not Runn...	Why python code dose not seem to want to run a...	Steelingsword94	2	0	0	1	python	204	8.0
8	49346981	-5	1	40	Convert binary files (malwares) into RGB images...	I'm working on a malware classification proble...	hardik0	7	0	0	3	python numpy machine-learning deep-learning co...	204	8.0
9	49338413	-4	1	42	Can we put in ascending order a list (input ()...	I made an exercise with France IOI, but i woul...	Yacine Alloul	1	0	0	0	python python-3.x	188	8.0

Some of the questions are:

Question 0: Retreive all the values of one column of Dataset in Django [on hold]

Question 1: Optimize nested loop in python [on hold]

Question 9: Can we put in ascending order a list (input()) in Python3 [on hold]

5. Top 5 Users for the similar cluster

Index	User	Frequency
10	Joe Iddon	2.0
4	Ajax1234	1.0
5	TMichel	1.0
11	Gareth Latty	1.0
12	Michael Robellard	1.0

Based on Frequency

Index	User	Reputation Score
218	Martijn Pieters	638000.00
27	Abarnert	210000.00
607	Alasdair	154000.00
977	ImportanceOfBeingErnest	84200.00
28	Martineau	58400.00

Based on Reputation Score

Future Scope

1. Intelligent Systems could be developed, which will provide solutions for different encountered problems.
2. We can find Co-relation between Questions and Answers and gain useful insights.
3. Deep Learning can be implemented to Rank and Categorize new questions based on historical data.
4. Out of 20,000 Questions scraped,
    a. Only 6,000 had Answers

b. Thus, percentage of available answers or solutions is approximately **30%.**

5. Using the analysis, we can forward the unsolved questions to corresponding **Experts**.
    a. Improve the percentage of answers and make Knowledge Market Websites like **STACKOVERFLOW/ GITHUB/ SLACK** more **Fruitful and Juicy**.