VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELGAUM-590014

MINI-PROJECT ENTITLED

“Autonomous Tagging of Stack Overflow Questions”

For the academic year 2019-2020
Submitted by:

AYUSHI	1MV17CS022
DHANYA N. NAIK	1MV17CS032
DIKSHA BHARTI	1MV17CS034
G. ARYA REDDY	1MV17CS036

Project carried out at

Sir M. Visvesvaraya Institute of Technology
Bangalore-562157.

Under the Guidance of
Mrs. SUSHEILA SHIDNAL
Assistant Professor,
Dept. of CSE
Sir MVIT, Bangalore.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Sir M. Visvesvaraya Institute of Technology
HUNASAMARANAHALLI, Bangalore-562157.

SIR M. VISVESVARAYA INSTITUTE OF TECHNOLOGY
BENGALURU -562157

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

It is certified that the project work entitled "**Autonomous Tagging For Stack Overflow Questions**" is a bona fide work carried out **by AYUSHI(1MV17CS022) DIKSHA BHARATHI(1MV17CS036), DHANYA N NAIK(1MV17CS032), G. ARYA REDDY(1MV17CS036)** in partial fulfilment for the requirements of mini project for the VI semester curriculum, Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the academic year 2019-2020.The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the course of Bachelor of Engineering.

Name & Signature of Guide	Name & Signature of HOD
Mrs. Susheila shidnal	**Dr.G C Bhanu Prakash**
Assistant Prof. & Internal Guide	HOD, Dept of CSE
Dept. Of CSE,	
Sir MVIT Bengaluru -562157	Sir MVIT Bengaluru - 562157

ABSTRACT

Online question and answer forums such as Stack Exchange and Quora are becoming an increasingly popular resource for education, whereby a user labels his/her post with an appropriate set of topics that describe the post, such that it is more easily retrieved and organized. This project aims in providing a system that is capable of automatically assign tags to question. We implement a one-vs-rest classifier for a Stack Overflow dataset, using a linear SVM and a carefully chosen subset of the entire feature set explored.

ACKNOWLEDGEMENT

It gives us immense pleasure to express our sincere gratitude to the management of **Sir M. Visvesvaraya Institute of Technology**, Bangalore for providing the opportunity and the resources to accomplish the project work in their premises.

On the path of learning, the presence of an experienced guide is indispensable and we would like to thank our **Mrs. Susheila shidnal**, Associate Professor, Dept. of CSE, for her invaluable help and guidance.

We would also like to convey our regards and sincere thanks to **Dr. G C Bhanu Prakash,** HOD, Dept. of CSE for his suggestions, constant support and encouragement. Heartfelt and sincere thanks to **Dr. V. R. Manjunath,** Principal, Sir MVIT for providing us with the infrastructure and facilities needed to develop ourproject.

We would also like to thank the staff of the Department of Computer Science and Engineering and lab-in-charges for their cooperation and suggestions. Finally, we would like to thank all our friends for their help and suggestions without which completing this project would not have been possible.

-Ayushi(1MV17CS022)
-Diksha Bharti(1MV17CS034)
-G. Arya Reddy (1MV17CS036)
-Dhanya N. Naik(1MV17CS032)

DECLARATION

We hereby declare that the entire mini project work embodied in this dissertation has been carried out by us and no part has been submitted for any degree or diploma of any institution previously.

Place: Bengaluru
Date: Signature of Students:

AYUSHI(1MV17CS022)

G. ARYA REDDY(1MV17CS036)

DIKSHA BHARATHI(1MV17CS034)

DHANYA N NAIK(1MV17CS032)

CONTENTS

Chapter 1

Introduction

The question-answering site StackOverflow allows users to assign tags to questions in order to make them easier for other people to find. Further experts on a certain topic can subscribe to tags to receive digests of new questions for which they might have an answer. Therefore it is both in the interest of the original poster and in the interest of people who are interested in the answer that a question gets assigned appropriate tags. StackOverflow allows users to manually assign between one and five tags to a posting. Users are encouraged to use existing tags that are suggested by typing the first letter(s) of a tag but they are also allowed to create new ones, so the set of possible tags in infinite. While the manual tagging by users generally works well for experienced users, it can be challenging for inexperienced users to find appropriate tags for their question and by letting users add new tags it is likely that different users use different orthographic versions of tags that mean the same thing such as “php5” and ”php5”. For these reasons it is desirable to a have a system that is able to either automatically tag questions or to suggest relevant tags to a user based on the question content. In this project we are developing a predictor that is able to assign tags based on the content of a question.

Chapter 2

Specifications

2.1 Software Requirements:

SOFTWARE	: Python 3.3 or greater
SUPPORTED BROWSERS	: Google Chrome / Mozilla Firefox / Internet Explorer
EDITOR	: Atom / Visual Studio Code
FRAMEWORK	: Flask 1.1.1
OPERATING SYSTEM	: Windows, or MACos, or Linux (32/64 bit)
PACKAGES	: sklearn

2.2 Hardware Components:

- Processor – Dual Core
- Hard Disk – 50 GB
- Memory – 1GB RAM

Chapter 3

Implementation

3.1 Installation

1. Clone the repository-git clone https://github.com/ayushianan/Tagging-of-stackoverflow-questions.git
2. move to folder cd stack project
3. Install packages as per instructed on your terminal
4. to run the application python app.py

3.2 Algorithm

3.2.1 LinearSVC

Similar to SVC with parameter kernel='linear', but implemented in terms of liblinear rather than libsvm, so it has more flexibility in the choice of penalties and loss functions and should scale better to large numbers of samples.

This class supports both dense and sparse input and the multiclass support is handled according to a one-vs-the-rest scheme.The underlying C implementation uses a random number generator to select features when fitting the model. It is thus not uncommon to have slightly different results for the same input data

3.2.2 Dummy classifier

A dummy classifier is a type of classifier which does not generate any insight about the data and classifies the given data using only simple rules. The classifier's behavior is completely independent of the training data as the trends in the training data are completely ignored and instead uses one of the strategies to predict the class label.

3.3 Code

1. In google-colab or jupyter-notebook-download dataset from kaggle https://www.kaggle.com/stackoverflow/stacksample.

2. **Extracting the data**
   - df = pd.read_csv("/home/ayushi/Downloads/stacksample/Questions.csv", Encoding="ISO-8859-1")

- tags = pd.read_csv("/home/ayushi/Downloads/stacksample/Tags.csv", Encoding="ISO-8859-1", dtype={'Tag': str})
- grouped_tags = tags.groupby("Id")['Tag'].apply(lambda tags: ' '.join(tags))
- grouped_tags_final = pd.DataFrame({'Id':grouped_tags.index, 'Tags':grouped_tags.values})

3. **Delete unnecessary columns**
   - df.drop(columns=['OwnerUserId', 'CreationDate', 'ClosedDate'], inplace=True)
   - df.drop(columns=['Id', 'Score'], inplace=True)
   - df = df.merge(grouped_tags_final, on='Id')
     **finally it contains three columns Title,Body,Tags

4. **Remove duplicates if any**
   - print ('Duplicate entries: {}'.format (df.duplicated().sum()))
     df.drop_duplicates(inplace = True)

5. **Clean the data**
   - For Title,Body using these functions
   - def lemitizeWords(text):
   - def stopWordsRemove(text):
   - def clean_punct(text):
   - def strip_list_noempty(mylist):
   - def clean_text(text):

6. **Spilt the dataset**
   - X1 = new_df['Body']
     X2 = new_df['Title']
     y = new_df['Tags']
   - multilabel_binarizer = MultiLabelBinarizer()
     y_bin = multilabel_binarizer.fit_transform(y)
     vectorizer_X1 = TfidfVectorizer(..)
   - X1_tfidf = vectorizer_X1.fit_transform(X1)
     X2_tfidf = vectorizer_X2.fit_transform(X2)
     X_tfidf = hstack(..)
   - X_train, X_test, y_train, y_test = train_test_split(X_tfidf, y_bin, test_size = 0.2, random_state = 0)

7. **Train the model**
   - Tuning the hyper-parameters of an estimator
   - svc = OneVsRestClassifier(LinearSVC())

     CV_svc = model_selection.GridSearchCV(estimator=svc, param_grid=param_grid, cv=
     5, verbose=10, scoring=make_scorer(avg_jacard,greater_is_better=True))
     CV_svc.fit(X_train, y_train)

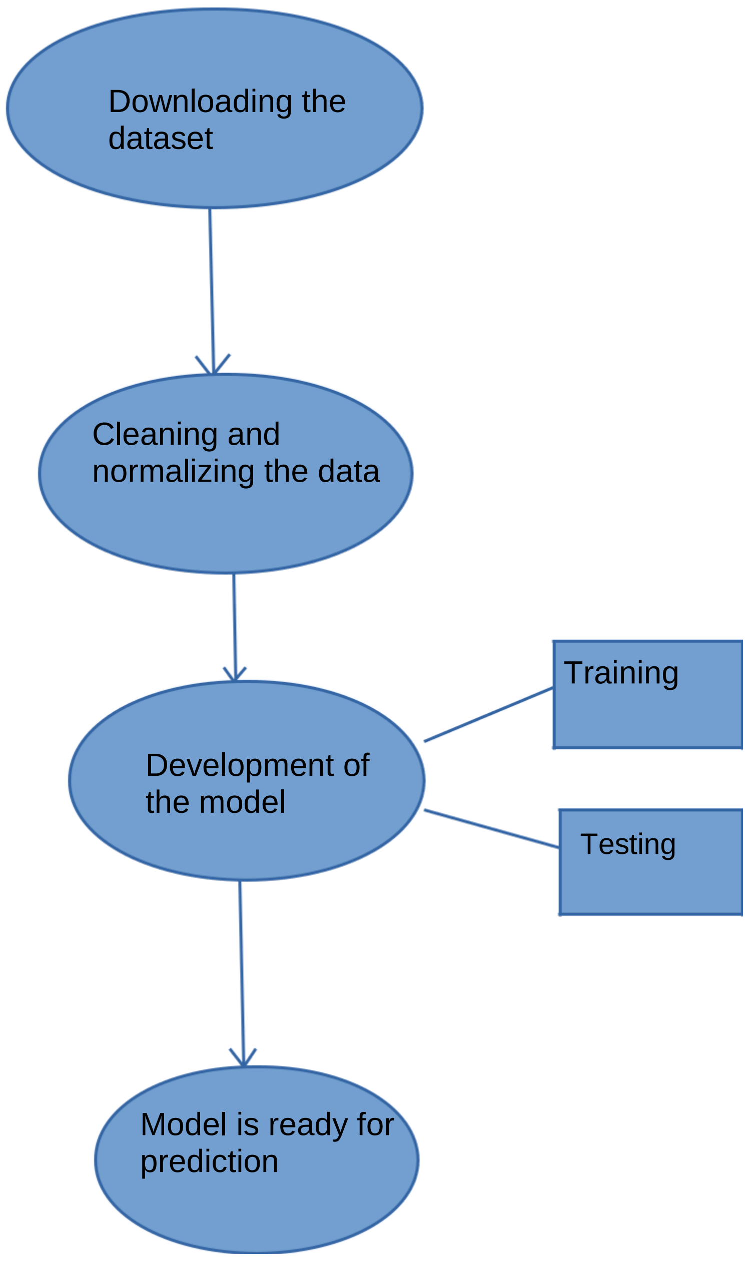
   jacard score: 52.356208115666725
   Hamming loss: 0.9552794047807505

- dummy = DummyClassifier()
```
        For classifier in [dummy]:
        clf = OneVsRestClassifier(classifier)
        clf.fit(X_train, y_train)
        y_pred = clf.predict(X_test)
jacard score: 2.5994136507433954
Hamming loss: 3.00411587779009
```

8. **Download the model**

- import pickle
```
With open('model.pkl','wb') as f:
pickle.dump(best_model, f)
```

3.4 Flow diagram

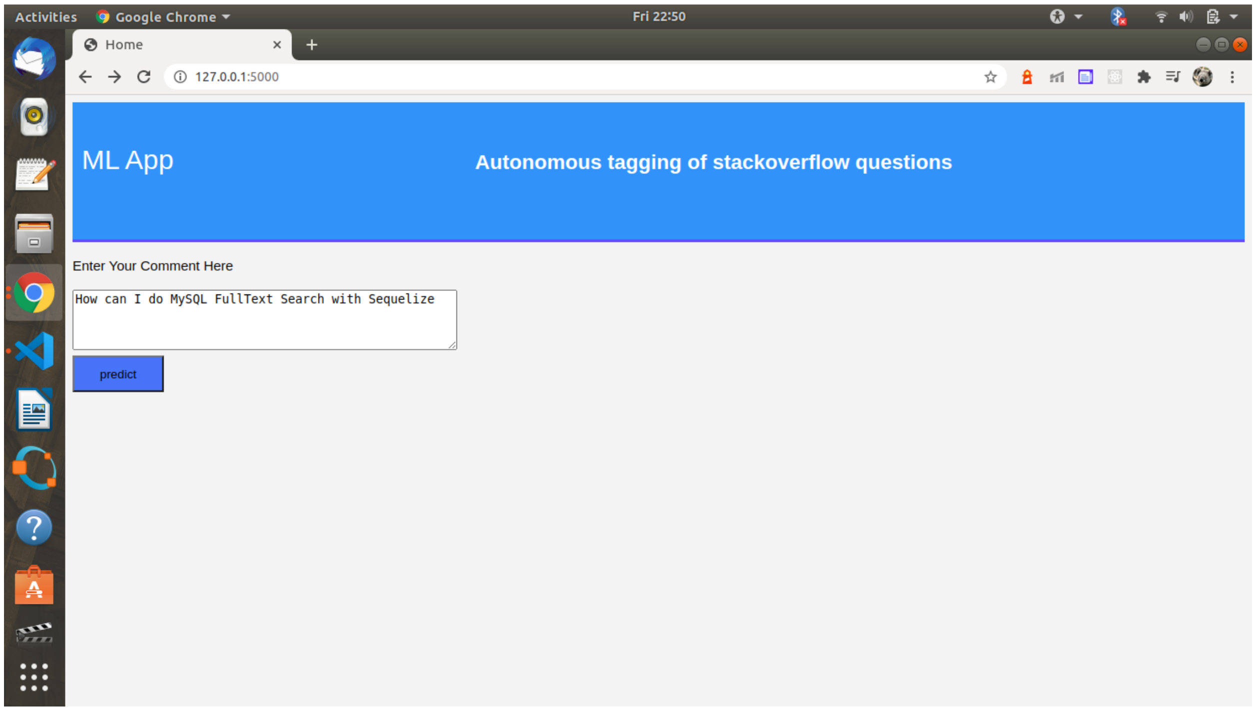

Downloading the dataset
Cleaning and normalizing the data
Development of the model
Training
Testing
Model is ready for prediction

3.5 Link

- https://github.com/ayushianan/Tagging-of-stackoverflow-questions

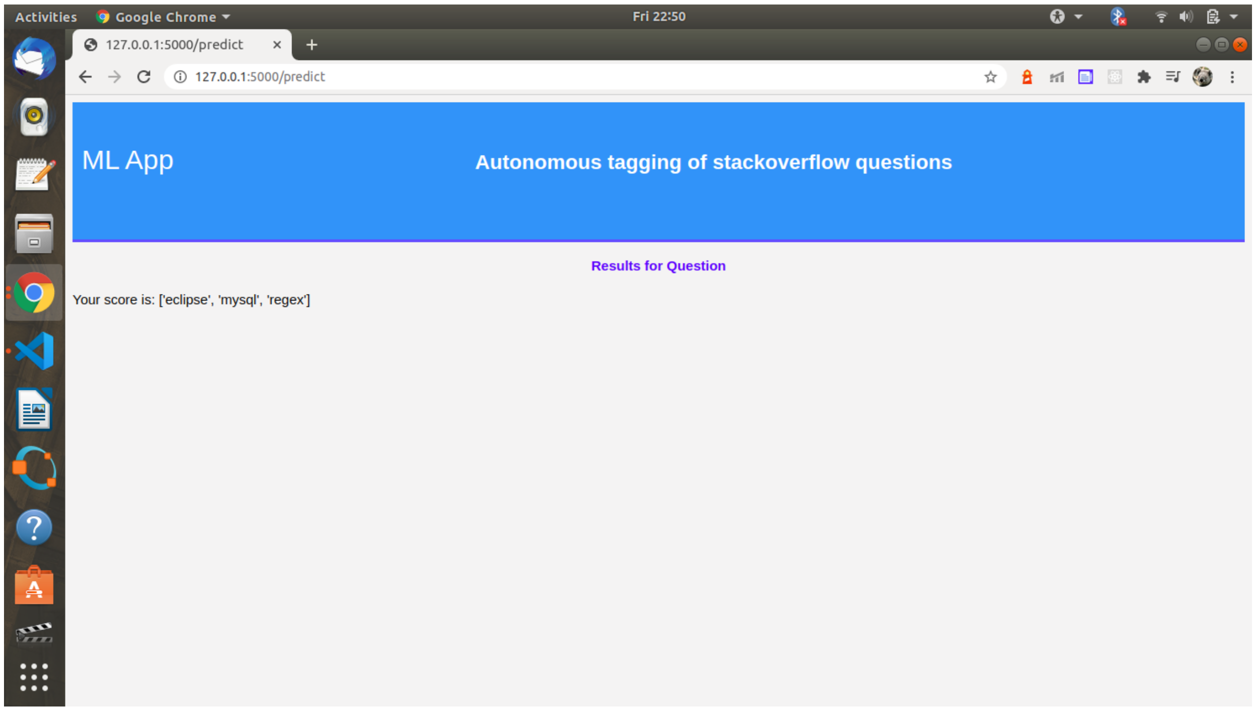
Chapter 4

Results



Enter Your Comment Here

```
How can I do MySQL FullText Search with Sequelize
```



Results for Question

```
Your score is: ['eclipse', 'mysql', 'regex']
```

Chapter 5

Conclusions

We described a simple classifier that has the capability of predicting tags to StackOverflow questions given only the question title and body. Besides engineering more sophisticated features, future work should focus on optimizing the runtime of our model, as it is currently too slow to be used in practice. We also assume that incorporating more data to estimate the PMI values, could further improve recall as it seems that our model heavily relies on that feature. Further, one might want to investigate whether co-occurrence statistics of tags could further improve the model, as there are a lot of tag pairs such as JavaScript and jQuery that co-occur very often.

Chapter 6

References

- http://cs229.stanford.edu/proj2014/Mihail%20Eric,%20Ana%20Klimovic,%20Victor%20Zhong,MLNLP-Autonomous%20Tagging%20Of%20Stack%20Overflow%20Posts.pdf
- https://towardsdatascience.com/auto-tagging-stack-overflow-questions-5426af692904