ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)
Vol. 2, Issue 3 (July - Sept. 2014)
International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)

Context-Aware Computing with Semantic Data

[I]Sanjay Rao H N, [II]Pushpa H G
[I]Dept. of CSE, School of Engg. & Technology, Jain University, Kanakapura Road, Bangalore, India
[II]Professor & H.O.D, Dept. of CSE, School of Engg. & Technology, Jain University, Kanakapura Road, Bangalore, India

Abstract
Web content tagging has been rising in recent years and perpetual, most of which are currently done with human intervention. Contents tagged with appropriate tags/keywords are used in search engines extensively. Application of Machine Learning methods with NLP for text data mining in identifying keywords/tags is hot and trending area. Q&A sites like StackOverflow, Quora, StackExchange provide tags for the questions to segregate the area to which questions belong, thus improving the visibility of questions, user friendly, and also helps in search and explore relevant data quickly. Current research focused on the dataset provided by StackOverflow released through its Creative Commons data Dumps which is licensed under cc-by-sa. Challenge is to identify the appropriate tags from questions title and questions body which can be composition of code as well within standard code tags, tags prediction is based on Training. Training dataset contains over 6 million of questions with total of 42,000 unique tags, test data consists of over 2 million questions using the model developed from training dataset. Developed model is based on tf-idf numerical statistic which reflects how important a word is to corpus combined with word cooccurrence distribution.

Keywords
NLP, TF-IDF, Machine Learning, Cooccurence Distribution, Rooter

I. Introduction

The problem of correct tagging is not only complicated but also highly challenging with limited human power and oceanic ever-increasing data. One such site is StackOverflow, which is a Q&A site for Professionals and enthusiastic Programmers. Questions are posted continuously at an average of 300 Questions per hour1 given the instance of this training dataset of 7GB , the data can be classified as Big Data due to following features exhibited which conforms to 3Vs model defined by Gartner[2]:

1. Volume : More than 7.5 million questions which has more than 14 million answers
2. Velocity : More than 63 million unique visitors per month
3. Variety : Data to be processed includes different forms like Normal English sentence, HTML Tags, links, URLs, line breaks, punctuations, Programming Codes. Each record in the training dataset contains four fields: Question ID, Title, Body, Tags/Keywords

example post is shown above in Figure 1, the tags defined are user defined while posting the question, website rules make it mandatory to specify tags while posting questions. In order to find appropriate tags we require Context-Aware computing, Context-aware Software: Software that examines and reacts to an individual's changing context, that scales across different contexts of data, in this case like the StackExchange site which hosts different websites including stackoverflow.com, music.stackexchange.com. The data to be analyzed should be refined and remove signal variance, Semantic data : Purpose of semantic data would be to allow computers to understand and figure out information without the help of a human user. In this research study due to availability of the data dump from stackoverflow.com alone we consider the context to be from different areas of programming which includes languages like C, C++, PHP, Java, JavaScript, JQuery etc., Operating systems like Windows, Unix, Linux, editors like eclipse, NetBeans, however the developed system can be positioned to work even with varying contexts as along data is in textual.

II. Related Work

Processing is broken into following stages.

A. Data Processing

Removal of Noise Data and Processing

Noise data here includes the XML and HTML tags which are present, like <p>, &lt, also removal of whitespaces. Code, contents within <code> blocks are extracted and kept in separate section of record.

Stop-Word Removal

Common English words like the, if, or are moved from title and body fields, as they are low-predictors and for finding the unique words mapping it is required to "stop" these words from

[1]From StackOverFlow post http://meta.stackexchange.com/questions/201651
[2]Gartner Technology Research Website, http://www.gartner.com/technology/home.jsp

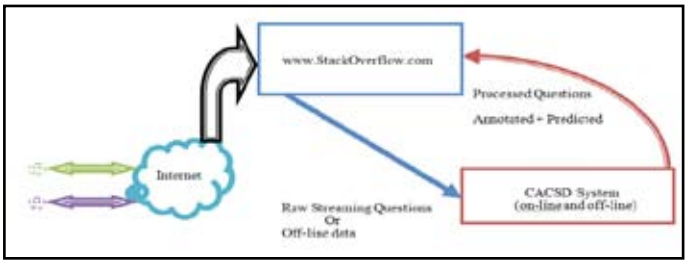
ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

analyzing

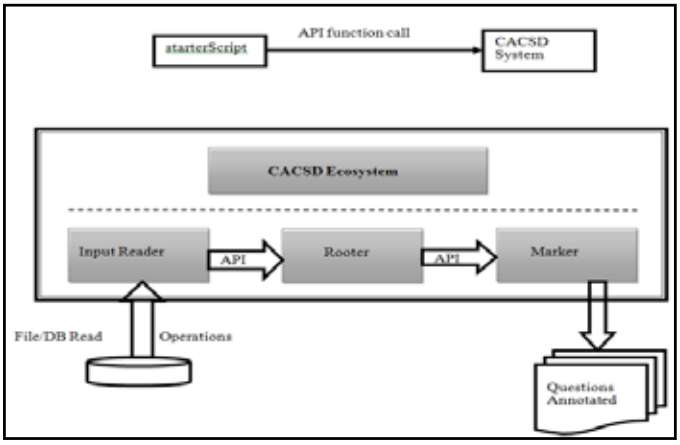
Stemming

It is the common practice to use stemming process in NLP which reduces the words to their condensed form, used Stemmer algorithm was PorterStemmer Algorithm which removes commoner morphological and in flexional ending in words. PorterStemmer Function(*PSF*) PSF (weak, weakness, weakened, weakening, weakener) = (weak, weak, weaken, weaken, weaken)

B. Proposed Model



The Fig. 2 above shows the possible deployment of the model in the real world where the sophisticated model can learn from off-line data and parallel try predicting the questions posted online.



Core features of the system in Fig. 3

Input Reader

Does the Data processing and stores the result back in the CSV file itself, initial plan of storing the preprocessed data into the Hbase had performance issues since the data read and analyzed are in the sequential form, javacsv.jar was used for reading the CSV files, it handles multi lines fields well, and handles the column and record delimiters within data using quoted text qualifiers.

Rooter

Performs the task of learning from the Training data, the algorithm used is TF-IDF (Term-Frequency and Inverse Document Frequency) together with Co-occurrence distribution between the terms and the corresponding tags learnt during the Training. Term-Frequency - Inverse Document Frequency assigns values to the terms to signify how good the terms describe the document in the corpus. Term-Frequency highlights the words that occur lot in a document, and considers it to represent the meaning of the document well. For the term (t) and document (d), d∈ D, and t appears in n of N documents in D. TFIDF function is of form

$$TFIDF(t,d,n,N) = TF(t,d) \times IDF(n,N)$$

There are many possible functions, in general

$$TF(t,d) = \begin{cases} 1 & \text{if } t \in d \\ 0 & else \end{cases}$$

Term frequency maybe normalized with some range. Only terms that appeared in at least one training document are used during comparison.

Currently used TF function during study is defined as follows:

If is the count of term t in character sequence then TF is defined as:

$$IDF(t,N) = \sqrt{\log\left(N \div df(t,N)\right)}$$

IDF function weights the term negatively relative to the number of documents which contain the term, the terms occurring in many documents don't discriminate the documents well.

If be the document frequency of token t, that is, the number of documents in which the token t appears. Then the inverse document frequency () of t is defined as:

$$IDF(t,N) = \sqrt{\log\left(N \div df(t,N)\right)}$$

Co-occurrence refers to pair of words which appear together in document, there may be other words in between, but the relevance of these words occurrence are of high probability.

Marker

Tags the fresh set of questions based on the knowledge gained from the training data, for this purpose Rank of a tag with relevance to the question was considered, and Top 5 ranked Tags were predicted. Rank for the tags increase when more unique terms in the Question contains the same tags or when unique terms identified as features contain fewer tags. Finally the scores identified are massaged with the IDF value to remove the most occurring tag to uniquely identify the tags.

III. Results

A. Evaluation Criteria

Evaluation was done on Macro-F1 Score, Macro average of F1 score is found best to evaluate the system performance across different data sets. F1 Score has following terminologies

True Positives (TP)
Tags which are predicted and which are correct

False Positives (FP)
Tags which are predicted and are incorrect

False Negatives (FN)
Tags which are correct but were not predicted by the system

$$Precision\ (P) = \frac{tp}{(tp+fp)} \qquad Recall\ (R) = \frac{tp}{(tp+fn)}$$

$$F1\ Score = \frac{2 \times P \times R}{P + R}$$

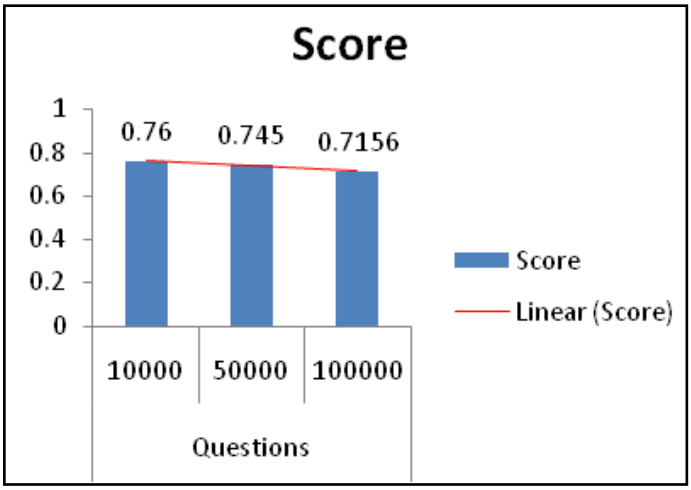
So in order to achieve higher F1 score both Precision and Recall

should be of higher order value. Macro-F1 Score takes the mean of the Precision values and Recall values to get the weighted harmonic mean across different data sets.

B. Sample Results

True Tags	Predicted Tags
android webview loaddata	android javascript
java android	android java javascript
sql sql-server tsql datetime	tsql sql php sql-server-2005
regex boost backslash	regex boost

With the current study the model predicts the Tags best with the Macro-F1-Score of 0.78, other methods are studied to improve this score and experiments are done to improvise the score.



IV. Conclusion

In this study the developed system could predict tags for Questions on Questions-and-Answers sites like StackOverflow, the classifier trained with co-occurrence distribution with TF-IDF model was found effective after experimenting with different classification algorithms. Data-set included some invalid records which affected the model learning.

V. Future Work

Model can be developed for parallel training with distributed computation like Hadoop framework combined with Machine learning techniques like Apache Mahout. This can not only increase the speed of the prediction but also combination of the several classification algorithms can be taken into consideration based on weighted scores.

References

[1] *System and method for Annotation and Ranking Reviews Personalized to Prior UserExperience, Authors: Yahia, Sihem Amer et. Al 2013.*
[2] *Context-Aware Computing: Opportunities, Authors: Edward Y. Chang , HTC, 2013.*
[3] *Sabine Schulte im Walde1 and Alissa Melinger, An in-depth look into the co-occurrence distribution of semantic associates, 2008.*
[4] *Bird, Steven, Edward Loper and Ewan Klein, Natural Language Processing with Python. O'Reilly Media Inc, 2014.*
[5] *Joseph O'Connor, NLP Workbook: A practical guide to achieving the results you want by, Thorsons Publications, 2014.*
[6] *Christopher Manning, Hinrich Schütze, Foundations of Statistical Natural Language Processing, The MIT Press Publications, 2014.*
[7] *Wang Jian, Davidson Brian, Explorations in tag suggestion and query expansion, ACM workshop on Search in social media, 2008.*
[8] *Eibe Frank, Gordon W Paynter, Ian H Witten, Carl Gutwin, and Craig G Nevill-Manning. Domain-specifc keyphrase extraction, 1999.*
[9] *Saha A, Saha R, Schineider K, A discriminative model aapproach for suggesting tags, 2008.*
[10] *StackOverflow datset is obtained from Creative Data Dumps http://data.stackexchange.com/help*
[11] *Xuchun Li, Lei Wang, and Eric Sung, Adaboost with svm-based component classifiers. Engineering Applications of Artificial Intelligence, 2008.*
[12] *Sabine Schulte im Walde and Alissa Melinger, An in-depth look into the co-occurrence distribution of semantic, 2008*