Open Data Profiling, Quality and Analysis

Chinmay Wyawahare
New York University Tandon School of Engineering
Brooklyn, New York
chinmay.wyawahare@nyu.edu

HemanthTeja Yanambakkam
New York University Tandon School of Engineering
Brooklyn, New York
hy1713@nyu.edu

Vineet Viswakumar
New York University Tandon School of Engineering
Brooklyn, New York
vv913@nyu.edu

ABSTRACT

Often, data scientists, machine learning engineers and big data specialists find it hard to identify sources of high quality data. Data available in public domains are often riddled with data quality issues such as missing values, missing metadata, non standard entries for column values, etc. In this project, we go through the data available at the NYC Open Data site and anticipate and fix the issues mentioned above in these datasets.

KEYWORDS

Data profiling, data mining, big data

ACM Reference Format:
Chinmay Wyawahare, HemanthTeja Yanambakkam, and Vineet Viswakumar. 2019. Open Data Profiling, Quality and Analysis . In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Data available in public domains are often riddled with data quality issues such as missing values, missing metadata, non standard entries for column values, etc. Whenever we start off with a public dataset, many-a-times it's the case when the data hasn't undergone data cleaning or data preprocessing. As a result, we have dataset containing NaNs of zeros or sometimes erroneous headers as well. In order to use the dataset and perform analysis over such dataset, we follow standard methods of data cleaning, data pre-processing and data profiling to make the data usable for production level usage.

The dataset which we have used for this project consisted of 1900 csv files comprising over 37 GBs of data. In order to process this vast amount of data, we used New York University's High Performance Computing 48-node DUMBO cluster and AWS t2.2xlarge instance to handle the heavy computations. Using bigdata technologies like Spark, Pandas and Modin (Distributed pandas) to work on dataframes comprising the dataset, we calculated metadata for each dataset and each column to identify misclassified entries. With the metadata information, we computed statistics to look at the data distribution and report the data types like integer, float, string, date. Along with this, we calculated precision and recall as metric for our algorithm to identify the labels from NYC Columns datasets using Fuzzy logic and regular expressions (Regex).

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

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

2 PROJECT TASKS

2.1 Task 1

In Task 1, we perform generic profiling as in real world open data rarely comes with well documented metadata. This makes it hard for developers and data scientists to identify the data type for column values and fix any issues there might be with the data. Lack of metadata also makes it hard to discern patterns that may be present in the dataset.

After importing the 1900 csv files, we processed each csv by creating a spark dataframe along with the datasets.csv file to find the mapping for each of the files. In order to find the various data types across each files, we defined user defined functions for each data type and computed the statistics per data types. Furthermore, we used *udf* functions on spark dataframe and *collected* the corresponding files to compute the statistics per each data type.

While performing this task, we encountered a major issue of typecasted data into the string column. We came across many entries which were of INTEGER, FLOAT and DATE data types but were typecasted into STRING datatype while data preparation. To make the analysis richer, we further took a deep drive into the STRING data type and computed the count for the datatypes - STRING/DATE, STRING/INTEGER, STRING/FLOAT to report the actual datatype count and breakdown the analysis for STRING datatype as well.

2.2 Task 2

In Task 2, we perform semantic profiling,i.e identify the data type for values in each column in the dataset and we count the number of occurences for eachsemantic type.

3 METHODOLOGY

To identify the semantic types mentioned in section 1.2, we use a combination of techniques including string matching, regular expressions and dictionary match.

3.1 Person Name

Person names come as either a combination of a First Name and Last Name or First Name, Middle Name and Last Name or the entire Full Name, eg. John Smith, Adam Elvis Presley, Vembu Srinivasan

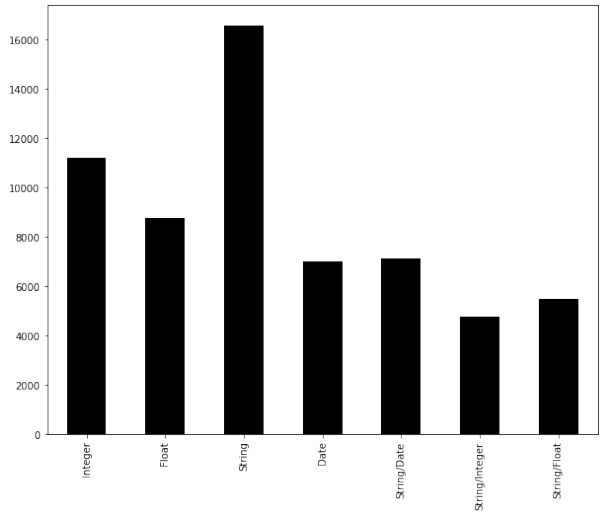

Figure 1: Distribution of data types across NYCOpenData

Raghav Hari Krishna.

Another issue when it comes to detecting names is that sometimes people have names of famous locations, deities, cardinal directions, colors, etc., eg.Kanye West.

A simple solution we implemented to solve this issue is run a regular expression and check if the value is a combination of alphabets and that the first letter is a capital letter as names don't contain numbers or special characters.

3.2 Business Name

Identifying names of businesses and companies is hard as they can easily be confused with names of people or places.

To identify if the column value is the name of a business, we run a regular expression to see if the column value has a registration value in its name, eg.LLC, Inc, lp., corp., co., co, ltd, capital, services, holdings.

3.3 Phone Number

In the US, phone numbers are a 10 digit combination. Each 3 digit area code has a capacity of 7,919,900 telephone numbers. Since phone numbers allow only numbers, we can easily identify if a given value is a phone number by checking if it is a 10 digit number.The disadvantage of using a regex to check if the value is a valid phone number is that in the US, the 5th and 6th digit cannot be 11 as these numbers are reserved for emergence services.

3.4 Address

An address represents the location of a building. As such, they include names of streets, avenues, zip codes, city and state names. To identify if the given value is an address, we compare the ratio of the occurrences of words like 'street','avenue', 'place' with the total number of words in that give value. If the ratio is above .85, we claim that the particular value is an address.One disadvantage of this method is that a full address can easily be confused for a street name.

3.5 Street Name

We use a similar technique compared to the technique we used in address identification to verify if the given value is a street name. Street name generally include words like 'street', 'avenue', 'road', 'boulevard', 'place' in their names.We compare the ratio of the occurrences of these words with the total number of words in the given entry. If the ratio is above 0.85, we claim that the value is a street name.

The advantages of this method is that we ignore the name and just search for words like 'street' or 'avenue'. This makes it less likely that the algorithm would confuse it for the name of a person as quite often, streets are named after famous personalities.

3.6 City

A city is defined as a large town with a local government. Cities, like places or locations are named after the people who discovered it or after some cultural significance and hence have no discernible pattern by which we can say that a given value is a city. To verify if a given value is a name of a city, we compare the value with a list of city names stored in a dictionary. If the name exists in the dictionary, we claim that the value is a city.

The advantage of this method is that the process is very quick. Searching in a dictionary is a fairly quick process and does not have any considerable memory overhead. The disadvantage of this method is that the dictionary might not contain the names of all the cities that exist. Hence, there is a probability that the value is not matched to an entry in the dictionary even though it is a city.

3.7 Neighborhood

A neighborhood is a geographically localised community. Neighborhoods can be combination of street names and addresses and can even include famous localities. To identify if a given value is a neighborhood, we create a list of known neighborhoods for the 5 boroughs in New York. We then check if the given value exists in the list we've built. If it exists, we claim that the value is the name of a neighborhood.

3.8 Coordinates

The geographical co-ordinate system can be used to pinpoint an exact location on a map. The range of values for coordinates vary from 90°N to 90°S for latitude and 180°E to 180°W for longitude. To identify if a given value is a coordinate, we run a regular expression to verify if the given value is a number and is lesser than 90 for latitude, or lesser than 180 for longitude.

3.9 Zip code

A ZIP code is a postal code used by the United States Postal Services. It is a 5-digit combination of numbers. The ZIP+4 code is a 9 digit combination of numbers which includes the 5 digit ZIP code and a 4 digit number representing a more specific location.To identify if a given value is a zip code, we run a regular expression to check if the given value contains 2 sets of numbers: a 5-digit number followed by a 4-digit number separated by a hyphen. If the value matches the regex pattern, we claim that the value is a ZIP code.

An advantage of using regex for ZIP codes is that it does not have the same restrictions as phone numbers in the US. This makes it quicker to identify if the given value is a ZIP code.

3.10 Borough

To verify if a given value is a borough, we create 2 lists. The 1st list contains values of counties and 2nd list contains the names of boroughs in New York City. For a value to be considered a borough, the value must be present in the counties list or the boroughs list. If the value is not present in either list, we claim that the value is not a borough.

3.11 School name

To verify if a given value is a school name, we compare the ratio of the similarity between the given value and a list of school names. If the ratio is above 0.85, we claim that the given value is a school name.

One disadvantage of this method is that the value might not be in the list even if it was the name of a school. This makes it impossible to detect which will leas to misclassification. Another potential drawback of this method is that it could be time consuming when implemented on large datasets.

3.12 Color

To verify if a given value is a color, we compare the ratio of the similarity between the given value and a list of colors. If the ratio is above 0.85, we claim that the given value is a color.

3.13 Car make

To verify if a given value is a car manufacturer, we compare the ratio of the similarity between the given value and a list of car manufacturers. If the ratio is above 0.85, we claim that the given value is the name of a car manufacturer.

3.14 City agency

City agencies help maintain different essential and non essential services to it's residents. To verify if a given value is a city agency, we compare the ratio of the similarity between the given value and a list of city agencies and their abbreviations. If the ratio is above 0.85, we claim that the given value is the name of a city agency.

3.15 Areas of study

To verify if a given value is an educational major, we compare the ratio of the similarity between the given value and a list of majors offered by universities and colleges. If the ratio is above 0.85, we claim that the given value is an area of study.
The advantage of this method is it's speed. Since we're searching for similar values in a dictionary(just similar, doesn't have to be exactly the same), the algorithm can match these patterns quickly. The limitation of this strategy is that the list may not have all possible majors offered by universities. This may lead the algorithm to believe that a certain value is not a university major because it doesn't match any value in the list, while in reality, it is a major offered by universities.

3.16 Subjects in school

To verify if a given value is a school subject, we follow the same process as we did for academic majors, i.e we compare the ratio of the similarity between the given value and a list of subjects thought in schools. If the ratio is above 0.85, we claim that the given value is a school subject.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets.

3.17 School Levels

To verify if a given value is a school level, we compare the ratio of the similarity between the given value and a list of school levels. If the ratio is above 0.85, we claim that the given value is the name of a school level.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets.

3.18 College/University names

To verify if a given value is the name of a university, we compare the ratio of the similarity between the given value and a list of universities and colleges. If the ratio is above 0.85, we claim that the given value is the name of a University or College.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets. Another problem could arise as many universities or colleges can be named after famous personalities or places. By employing a dictionary mapping method, there could arise a situation where we claimed a value is a university when in reality it is not. e.g New York could be claimed to be a university as it bears resemblance with New York University while in reality, it the name of a state.

3.19 Websites

A website is an address where a user can find and host information on the internet. Websites generally start with 'www' and end with 'com', 'in', 'org', 'info', etc. To verify if a given value is a website, we run a regular expression to split the value by '.'. After splitting,

if we find any of the values mentioned('www', 'http', 'https', 'com', 'org', 'gov', 'in', 'me', 'info', 'nyc', 'us') we claim that the value is a website.
An advantage of this method is that we save time by not comparing the entire string. Instead, we only search for patterns that must definitely exist in websites by using regular expressions and string matching.

3.20 Building Classification

To identify if the entry is a building classification, we first get a list of building codes and it's description from the NYC government website. We then compare the ratio of the similarity between the value and the list we've created. If the ration is greater than 0.85, we claim that it is a building classification.

3.21 Vehicle Type

Vehicle types can be broadly classified as vans, cars, buses, tractors, lorries, etc. To verify if a given value is a vehicle type, we compare the ratio of the similarity between the given value and a list of known vehicle types. If the ratio is above 0.85, we claim that the given value is the name of a vehicle type.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets.

3.22 Type of location

To verify if a given value is a location type, we compare the ratio of the similarity between the given value and a list of location types. If the ratio is above 0.85, we claim that the given value is the name of a type of location.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets.

3.23 Parks and Playgrounds

To verify if a given value is a park or a playground, we compare the ratio of the similarity between the given value and a list of known parks and playgrounds. If the ratio is above 0.85, we claim that the given value is the name of a park or playground.
The limitation of this strategy is that for every value, we need to check the entire list to find if there are similar entries. This can be a very inefficient method with regards to speed when it comes to large datasets.

4 PRECISION AND RECALL

4.1 Precision

Precision is measured over the total predictions of the model. It is the ratio between the correct predictions and the total predictions. In other words, precision indicates how good is the model at whatever it predicted.

4.2 Recall

Recall is the ratio of the correct predictions and the total number of correct items in the set. It is expressed as percentage of the total correct(positive) items correctly predicted by the model. In other words, recall indicates how good is the model at picking the correct items.

4.3 Task-2 Statistics

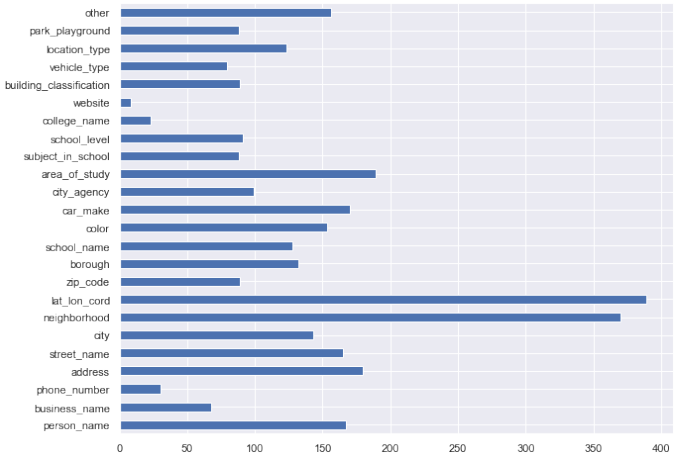

Figure 2: Distribution of labels across the NYCColumns data

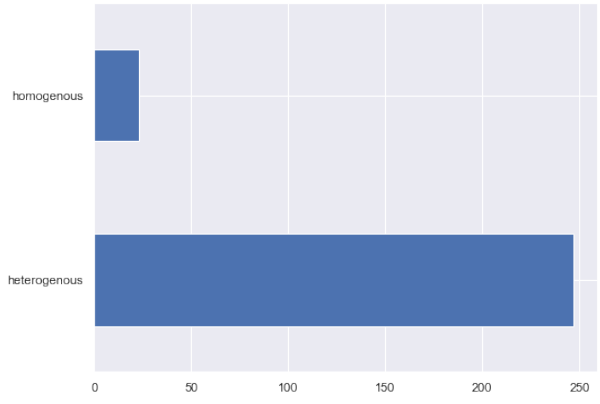

Figure 3: Distribution of nature of data

5 TASK 3

For task 3, we chose to work on NYC 311 dataset and identify the three most frequent 311 complaint types by borough. We developed different visualizations like area charts, treemap, stacked bar graphs and so on to address questions like:

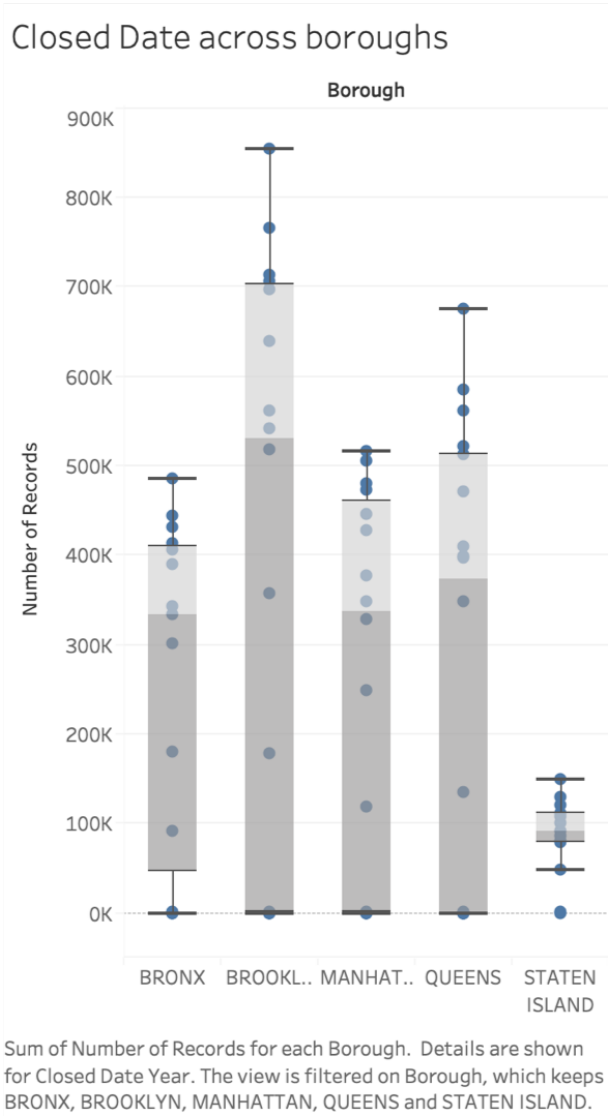

Figure 5: Closed Date across boroughs

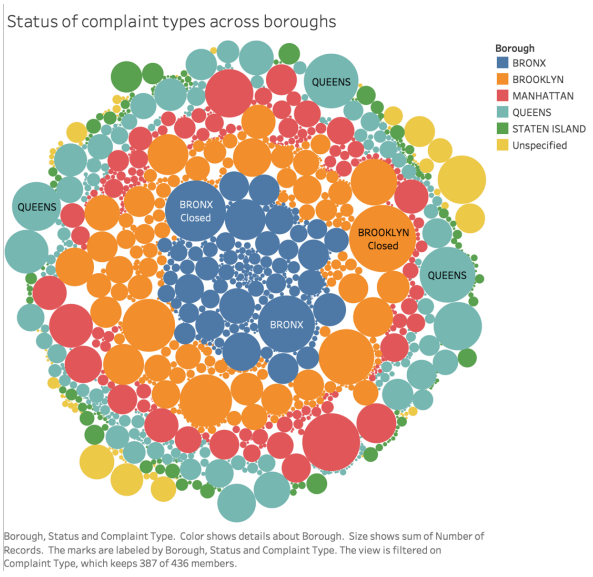

Figure 6: Complaint Type status across boroughs

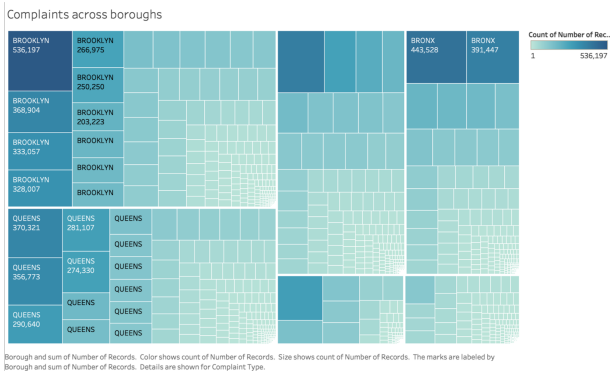

Figure 7: Complaints across boroughs

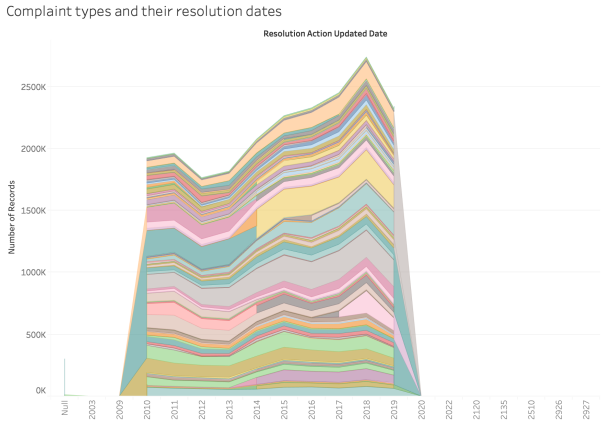

Figure 8: Complaint types over resolution dates

- Identify the three most frequent 311 complaint types by boroughs
- Are the same complaint types frequent in all five boroughs of the City?
- How might you explain the differences?
- How does the distribution of complaints change over time for certain neighborhoods and how could this be explained?

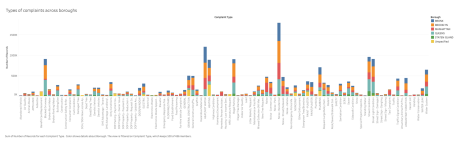

Figure 4: Types of complaints across boroughs

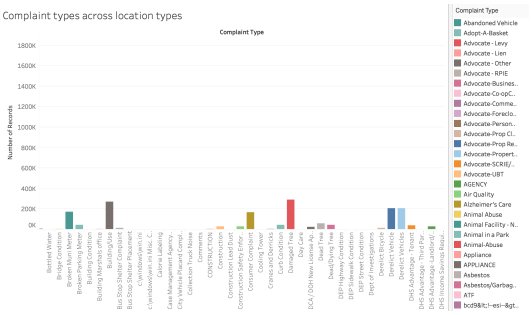

Figure 9: Types of complaints across various location types

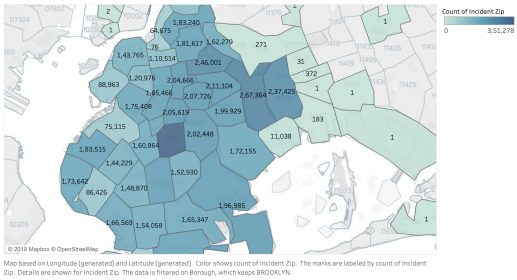

Figure 10: Distribution of incident zip across New York City

Figure 4, 5 and 6 depict the types of complaints across boroughs of New York City. These present the distribution of complaints along with various metrics across boroughs of New York City.

6 SUBMISSION

You can find the code on the GitHub repository:
https://github.com/gandalf1819/NYCOpenData-Profiling-Analysis

DUMBO HDFS directory:
/user/cnw282/2019BD-project-results

REFERENCES

[1] Yeye He and Dong Xin. SEISA: set expansion by iterative similarity aggregation. International conference on World Wide Web (WWW '11), 2011

[2] Fatemeh Nargesian, Erkang Zhu, Ken Q. Pu, and Renée J. Miller. Table union search on open data. Proc. VLDB Endow. 11, 7 (March 2018), 813-825

[3] Meihui Zhang, Marios Hadjieleftheriou, Beng Chin Ooi, Cecilia M. Procopiuc, and Divesh Srivastava. Automatic discovery of attributes in relational databases. ACM SIGMOD International Conference on Management of data (SIGMOD '11), 2011

[4] Madelon Hulsebos, Kevin Hu, Michiel Bakker, Emanuel Zgraggen, Arvind Satyanarayan, Tim Kraska, Çagatay Demiralp, and César Hidalgo. Sherlock: A Deep Learning Approach to Semantic Data Type Detection. ACM SIGKDD International Conference on Knowledge Discovery Data Mining (KDD '19), 2019

[5] Andrew Ilyas, Joana M. F. da Trindade, Raul Castro Fernandez, Samuel Madden. Extracting Syntactical Patterns from Databases. ICDE 2018

[6] Andrew Ilyas, Joana M. F. da Trindade, Raul Castro Fernandez, Samuel Madden. Extracting Syntactical Patterns from Databases. ICDE 2018

[7] Efficient Algorithms for Mining Outliers from Large Data Sets. Ramaswamy et al., SIGMOD 2000

[8] Ming Hua, Jian Pei: Cleaning disguised missing data: a heuristic approach. KDD 2007: 950-958

[9] Sumit Goswami, Mayank Singh Shishodia. A Fuzzy Based Approach To Text Mining And Document Clustering . 2013