Processing User Service Agreement using various Data Mining Techniques.

Nikhil Gola(MT18129) , Ridha Juneja(MT18009) , Saru Brar(MT18014) , Yogesh Pandey(MT18140)

Abstract

The Project Aims at generating extractive summarization of Document containing Terms Conditions.The Approaches applied are Two Step Clustering and Rule based Mining.We examined results obtained by both the methods in order to understand the performance of these Data Mining Strategies on the problem statement.

1 Problem Statement

In This Digital Era with tons of Softwares and Webapps comes Terms and Conditions which the consumer accepts knowingly or unknowingly. So in order to make them informed the Terms and conditions must be summarized.So we want to summarize the text Data of Terms and Conditions by using Data Mining Approaches.

2 Motivation

A new Deloitte survey found that over 90 percent of consumers accept legal terms and conditions without reading them.When faced with no choice, users are willing to accept potential consequences in exchange for access. For younger people, ages 18-34, the rate is even higher with 97 percent agreeing to conditions before reading.So it was High time to find some solution for the problem

3 Introduction

The increase in number of software and websites for every minute functioning of our daily lives, have increased the data and information tranfer as well. Whether we download any app , software or register for any website , we are supposed to click the checkbox by clicking Join to agree to accept and abide by our terms and conditions and we just go for it without reading it.

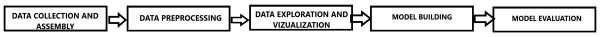

Figure 1: *The steps performed in preprocessing the data*

Keeping this problem in mind , we have created a system that will summarize the the text of terms and conditions by specifying the important contents of the file on the basis of privacy , data access , amendments and terminations and returning those set of lines from the document which are of importance from the above point specified by using techniques tf-idf, clustering , association pattern mining and visializing our results.

In this system , the summary we are trying to generate is an extractive summarization. This type of summarization method works by identifying important sections of the text and generating them verbatim ; thus, they depend only on extraction of sentences from the original text. These summaries contain the most important sentences of the input data, where the input can be a single file or multiple files.

4 Methodology

To build our system we have followed a framework. The high level steps for this framework are shown below.

To build this system , we have implemented two different approaches . Mainly the approaches used are different in terms of vizualization , and model building.

4.1 Data Collection and Assembly

To work on our project , we didnt have any predefined dataset. So we had to collect the data from every site and manually create the datatset for our use. After collecting that dataset we have divided it into train set and test set.

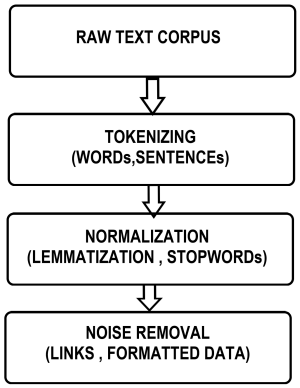

Figure 2: the steps performed in preprocessing the data *corpus*.

4.2 Data Preprocessing

Data preprocessing is a data mining technique that basically involves the transformation of raw data into an understandable format. So to preprocess the data of our system , we first decoded the corpus into binary format UTF-8 format. For every document we have divided it into sentences and and every sentence into words (tokens) .

From this set of words , we eliminated the stop words ,performed POS tagging,the process of marking words in the corpus corresponding to a particular part-of-speech , which is based on defination and the context .Then we removed the Noise , by eliminating punctuations , and words whose length is less than 3.

After doing this we have a set of unique words for the entire corpus relating to every file and count of all the unique words.

5 Data Exploration and Visualization

To implement this phase of our system we have tried two different data mining approaches to examine the diverse results we get by implementing them. The approaches are :

a) Two- step clustering b) Rule Based Mining

5.1 Two Step Clustering

After processing all the file content , system calculates feature values by computing *TF-IDF* score for the unique words over the entire corpus. TF-IDF for a word in a sentence is calculated as inversely proportional to the number of documents which contain that word. High value of a TF-IDF implies that word has a strong relation with in that sentence. We have used this approach to obtain the most important word that relates to the particular document. With this calculated TF scores we have , we have created a matrix , where the rows corresponds to the every document , and the coloumns corresponds to the unique words generated. After generating our feature vectors , we have performed K-Means clustering , to get the best relativity among the word of our corpus. To find the optimal value of k(number of clusters) we have implemented Elbow Method,the idea of this method is to run K-Means clustering on the data set for a range of values and then calculate the sum of sqaured errors(SSE). Then plot the line chart of SSE for each k . if the plot looks like a arm then the elbow of the arm is the value we get the optimal k. for our dataset we we analyse that our optimal k values lies inn the range (8,10). on observing we set of k = 9 and perfomed K-Means clustering.

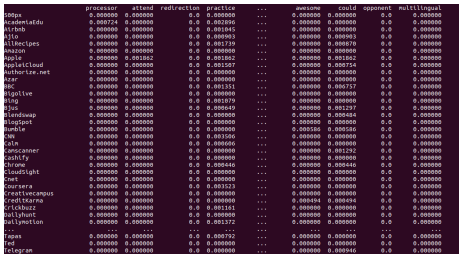

Figure 3: Feature vector matrix where rows corresponds to the document files and column corresponds to the unique words of our corpus.

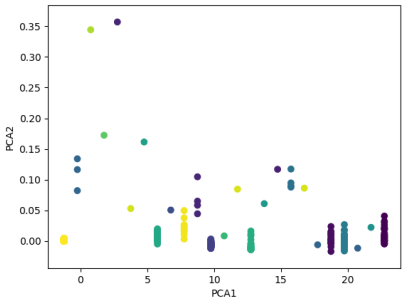

Figure 4: Cluster formed by taking k=50

After generating our feature vectors , we have performed K-Means clustering , to get the best relativity among the word of our corpus.

We first tried clustering by randomly choosing *k* (number of clusters), looking at the size of our word set (6600 words) , by taking k = 50 and 100 . on clustering we observed that at our important wordset gets distorts as seen in the figure.

To find the optimal value of *k* (number of clusters) we have implemented Elbow Method,the idea

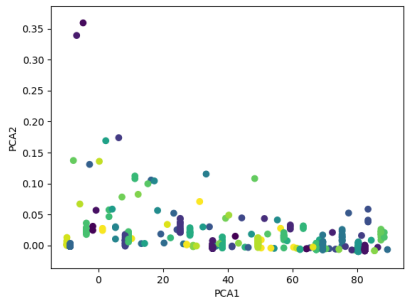

Figure 5: Cluster formed by taking k=100.

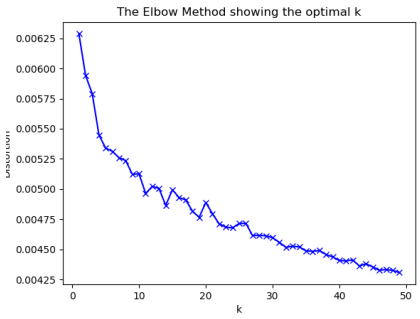

Figure 6: Elbow Method plot for our data gave k = 9 to be optimal

of this method is to run K-Means clustering on the data set for a range of values and then calculate the sum of squared errors(SSE). Then plot the line chart of SSE for each k . If the plot looks like an arm then the elbow of the arm is the value we get the optimal k. For our data set we we analyse that our optimal k values lies in the range (8,10).On observing we set of k = 9 and performed K-Means clustering.

After clustering to visualize our clusters we have performed dimensional reduction (from 173 feature vectors to 2 feature vectors) by doing feature projection, where it transforms the data in high dimension to fewer dimension space. for this we have used principal component analysis(PCA), that performs a linear mapping of the data to lower dimension in such a way that the variance of the data in the low dimension representation is maximized.

From here we start the second phase of our clustering, where for all the clusters, we have calculated a *cluster score*, that is the inverse of the number of words in that clusters. Then we analyzed them and founded two such clusters that have the most related words required to achieve our solution to the problem statement.One such

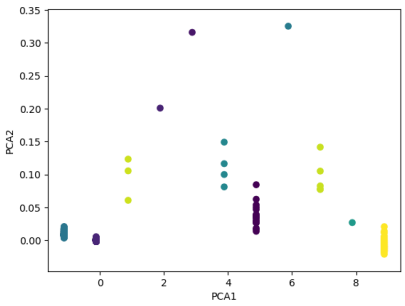

Figure 7: Clusters formed on taking k = 9.

For Files in Corpus:
  For Sentences in Files:
    For Words in Sentences:
      Check the cluster label for each word & Increment word count for that cluster
    Dictionary is formed in **'Z' format**
    assign sentences a label on the basis of **'Label Formula'**
  return Sentences with labels

Figure 8: the algorithm applied on the clusters

cluster which has all the words related to our problem statement is named as "Important Clusters".

Now we implement the following algorithm:

On running this algorithm, we get the sentences in a form of a dictionary in the "Z format", shown below.

We have mapped the sentences on the basis of a heuristic formula generated terms as the labelled as "Label Formula".And with these labelled sentences we extract those sentences with cover the "Important Label" and print them as the *final summary*.

5.2 Rule based Mining:

This technique basically Extract Association Rules from Text for the dictionary words and co-occurrence of other corpus-words.

Using Data insight - we had build the dictionary of privacy related terms. In this approach we have used the idea behind Apriori Algorithm in a different manner suiting our objective ,it is an algorithm for frequent item set mining and association rule learning over the different software license.The association rules are further found on the basis of Confidence : the confidence of co-occurrence, two words occur together and *Support* : this is basically the probability or IDF scores .

The algorithm used is as follows:

Step1: Filtering the corpus words only those

'Z' Format
Sentence : {'Cluster1' : word count, 'Cluster2': word count ..... 'ClusterN': word count }

Figure 9: The Z format look.

'Label Formula'
value = 0
if Cluster Label belongs to Important Cluster:
    value+= Word Count *Constant Number * Cluster Score
else:
    value+= Word Count * 1* Cluster Score
Sentence label = int(value)% Total Cluster

Figure 10: the label arkign heuristics

```
[[1.         0.33333333 0.33333333 ... 0.5        0.         0.5       ]
 [0.25       0.25       0.         ... 0.5        0.         0.25      ]
 [0.42105263 0.15789474 0.28947368 ... 0.28947368 0.02631579 0.26315789]
 ...
 [1.         0.875      0.375      ... 0.875      0.         0.25      ]
 [0.375      0.5        0.375      ... 0.25       0.         0.        ]
 [0.65277778 0.39583333 0.28472222 ... 0.36805556 0.00694444 0.24305556]]
```

Figure 11: Confidence matrix created

words are taken in consideration which are above the mean frequency.

*Step2:*Weighted each word in corpus with TF-IDF score.

Step3: Later we made confidence matrix for the pre-defined dictionary words corresponding to filtered corpus words this gives confidence for co-occurrence.

Then we calculate the below values:

$$support(word) = IDFscore(word)$$

$$P(A|B) = countoffilewordAandwordBappears$$

$$P(A) = countoffileswordAappears$$

$$confidence, C(A,B) = P(A|B)/P(A)$$

step4: Finding association rules : we had find the min-support and min-confidence by hit and trial initially using mean then later working with different parameters.

$$A(dictionaryword) -> B(wordcorpus)$$

The next step is the *Mining Phase:*

*step1:*In this we find the relevant sentences by mapping the various association rules to them , if any of the rule exist we are creating a map with the association rule and the associated sentences along the weight of each sentence.

step2: Later for each rule found and associated sentence , weight we have taken mean of those values then selecting only those sentence greater than the threshold(mean).Only these sentence will be in present in our *final summary.*

```
protection --> information
protection --> use
money --> law
money --> may
money --> account
money --> information
money --> make
money --> use
money --> service
privacy --> may
privacy --> use
nonexclusive --> policy
nonexclusive --> product
nonexclusive --> arise
nonexclusive --> connection
nonexclusive --> agree
nonexclusive --> claim
nonexclusive --> must
nonexclusive --> remain
```

Figure 12: The rules we get on the basis of confidence matrix

5.3 Model evaluation

To evaluate the model we have generated the ground truth by manually generating the summaries of the documents. And using this as our Ground truth to evaluate the model on the basis of similarity functions like cosine similarity , which we have implemented in 1-gram analysis method.

6 Observation

We Observed that both the techniques showed good amount of results.If we train our data with large data sets and also segregate them based on various types like social web apps, Browsers and various other formats we would get better results and can generate better summaries.

7 Results:

On implementing our two approaches we have generated the summary. The summary generated by *two fold clustering* for "DuckDuckGo.txt" is shown below:

the summary generated by *Rule based Mining* for "DuckDuckGo.txt" is shown below:

The summary generated by *two fold clustering* for "FaceBook.txt" is shown below:

the summary generated by *Rule based Mining* for "FaceBook.txt" is shown below:

another way to prevent search leakage is by using something called a post request, which has the effect of not showing your search in your browser, and, as a consequence, does not send it to other sites. you can turn on post requests on our settings page, but it has its own issues. post requests usually break browser back buttons, and they make it impossible for you to easily share your search by copying and pasting it out of your web browser's address bar.
other search engines save your search history. usually your searches are saved along with the date and time of the search, some information about your computer (e.g. your ip address, user agent and often a unique identifier stored in a browser cookie), and if you are logged in, your account information (e.g. name and email address).
in addition, when you visit any site, your computer automatically sends information about it to that site (including your user agent and ip address). this information can often be used to identify you directly.
similarly, we may add an affiliate code to some ecommerce sites (e.g. amazon & ebay) that results in small commissions being paid back to duckduckgo when you make purchases at those sites. we do not use any third parties to do the code insertion, and we do not work with any sites that share personally identifiable information (e.g. name, address, etc.) via their affiliate programs. this means that no information is shared from duckduckgo to the sites, and the only information that is collected from this process is product information, which is not tied to any particular user and which we do not save or store on our end. it is completely analogous to the search result case from the previous paragraph--we can see anonymous product info such that we cannot tie them to any particular person (or even tie multiple purchases together). this whole affiliate process is an attempt to keep advertising to a minimal level on duckduckgo.
duckduckgo does not collect or share personal information. that is our privacy policy in a nutshell. the rest of this page tries to explain why you should care. with only the timestamp and computer information, your searches can often be traced directly to you. with the additional account information, they are associated directly with you.

Figure 13: Summary generated for DuckDuckGo terms and condition file

DuckDuckGoSummary
DuckDuckGo does not collect or share personal information
Search engines could lose data, or get hacked, or accidentally expose data due to security holes or incompetence, all of which has happened with personal information on the Internet
However, in our case, we don't expect any because there is nothing useful to give them since we don't collect any personal information
your IP address, User agent and often a unique identifier stored in a browser cookie), and if you are logged in, your account information (e
Additionally, if you use our !bang syntax/dropdown, which bangs you use are stored in a cookie so that we can list your most frequently used ones on top of the !bang dropdown box
We do not use any third parties to do the code insertion, and we do not work with any sites that share personally identifiable information (e

Figure 14: Summary generated for DuckDuckGo terms and condition file

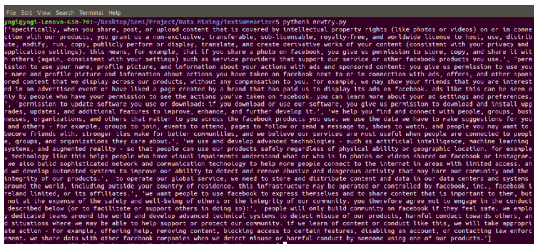

Figure 15: Summary generated for FaceBook terms and condition file

FacebookSummary
We also have developed, and continue to explore, new ways for people to use technology, such as augmented reality and 360 video to create and share more expressive and engaging content on Facebook
And we develop automated systems to improve our ability to detect and remove abusive and dangerous activity that may harm our community and the integrity of our Products
To provide these services, we must collect and use your personal data
Create only one account (your own) and use your timeline for personal purposes
Not share your password, give access to your Facebook account to others, or transfer your account to anyone else (without our permission)
We can remove content you share in violation of these provisions and, if applicable, we may take action against your account, for the reasons described below
If we determine that you have violated our terms or policies, we may take action against your account to protect our community and services, including by suspending access to your account or disabling it
We may also disable your account if you repeatedly infringe other people's intellectual property rights
To help support our community, we encourage you to report content or conduct that you believe violates your rights (including intellectual property rights) or our terms and policies
Specifically, when you share, post, or upload content that is covered by intellectual property rights (like photos or videos) on or in connection with our Products, you grant us a nonexclusive, transferable, sublicensable, royalty-free, and worldwide license to host, use, distribute, modify, run, copy, publicly perform or display, translate, and create derivative works of your content (consistent with your privacy and application settings)
This means, for example, that if you share a photo on Facebook, you give us permission to store, copy, and share it with others (consistent with your settings) such as service providers that support our service or other Facebook Products you use
In addition, content you delete may continue to appear if you have shared it with others and they have not deleted it
If you use content covered by intellectual property rights that we have and make available in our Products (for example, images, designs, videos, or sounds we provide that you add to content you create or share on Facebook), we retain all rights to that content (but not yours)
We may also suspend or disable your account if you create risk or legal exposure for us or when we are permitted or required to do so by law
We do not control or direct what people and others do or say, and we are not responsible for their actions or conduct (whether online or offline) or any content they share (including offensive, inappropriate, obscene, unlawful, and other objectionable content)

Figure 16: Summary generated for FaceBook terms and condition file

Filename	Accuracy with respect to ground truth
YouTube	87.5%
DuckDuckGo	60%
Facebook	88.23%
Paytm	68.42%
Loco	82.75%
Jabong	77.5%
GitHub	78.12%

Figure 17: Accuracy achieved by 2 step accuracy techniques

Filename	Accuracy with respect to ground truth
YouTube	75.5%
DuckDuckGo	64%
Facebook	83.45%
Paytm	60.89%
Loco	85.59%
Jabong	55.1%
GitHub	71.56%

Figure 18: Accuracy achieved by Rule based Mining techniques

The accuracy achieved by both the techniques is shown below for a particular test data set.

7.1 Refrences

[1] G. Sizov, Extraction-based automatic summarization: Theoretical and empirical investigation of summarization techniques, 2010. [2] M. A. Fattah and F. Ren, Automatic text summarization, World Academy of Science, Engineering and Technology, vol. 37, p. 2008, 2008. [3] About wordnet - wordnet - about wordnet, https://wordnet.princeton.edu/, accessed: 2015-05-15. [4] word2vec - tool for computing continuous distributed representations of words. - google project hosting, https://code.google.com/p/word2vec/, accessed: 2015-05-15. [5] J. Ramos, Using tf-idf to determine word relevance in document queries, in Proceedings of the first instructional conference on machine learning, 2003. [6] W. T. Chuang and J. Yang, Extracting sentence segments for text summarization: a machine learning approach, in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2000, pp. 152159 [7] H. Karanikas and B. Theodoulidis, Knowledge discovery in text and text mining software, Technical Report, UMIST Departement of Computation, January 2002. [8] H. Mannila, H. Toivonen and A. I. Verkamo, Discovery of frequent episodes in event sequences, Data Mining and Knowledge Discovery, 1(3), 1997b, pp. 259-289. [9] J. Paralic and P. Bednar, Text mining for documents annotation and ontology support (A book chapter in: "intelligent systems at service of Mankind, ISBN 3-935798-25-3, Ubooks, Germany, 2003). [10] M. Rajman and R. Besancon, Text mining:

natural language techniques and text mining applications. Proc. 7th working conf. on database semantics (DS-7), Chapan Hall IFIP Proc. Series. Leysin, Switzerland Oct. 1997, 7-10. [11] R. Agrawal and R. Srikant, Fast algorithms for mining association rules, In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, Proc. 20th Int. conf. of very Large Data Bases, VLDB, Santigo, Chile, 1994, 487-499. [12] R. Baeza-Yates and B. Ribeiro-Neto, Modern information retrieval (Addison-Wesley, Longman publishing company, 1999). [13] R. Feldman and I. Dagan, Knowledge discovery in textual databases (KDT), Proc. 1st nt. Conf. on Knowledge Discovery and Data Mining, 1995. [14] R. Feldman and H. Hirsh, Mining associations in text in the presence of background knowledge, Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, USA, 1996. [15] S. Brin, R. Motwani, and C. Silverstein, Beyond market baskets: generalizing association rules to dependence rules, KDD98, 1998, 39- 68