Analysis of Eluvio Data Science Challenge Data

In this project, we are given publication related data which is in csv format and contains eight features, ie date in UNIX format, date in DD/MM/YYYY format, the number of up_votes received by article, the number of down-votes received by article, title of article ,is the article only for readers above 18 ,author of article and category of article. This data-set contains approximately 500000 entries. Of the features, every article got 0 downvotes, and the category of every article is the same, i.e. world-news. As ths is a big data challenge. We used google colab, pyspark ,numpy and tensorflow for this challenge.

We began our analysis by first finding the authors with the highest upvotes count. The top 10 authors are maxwellhill, anutensil, Libertatea, Doremus Jessup, Wagamagam NinjaDiscoJesus, madazzahatter, madam1,kulkke and davidreiss666 Maxwellhill got 1935264 upvotes and ranked one of about 85090 writers, which are about 3.42% of total upvotes. We even created a dataframe of authors who had least upvotes i.e. 0. Of the total authors ,16.02% of authors got 0 upvotes. We then created a data-frame to find authors with highest upvotes to the number of articles available.We found that this data-frame mostly contains those writers who have just one article. At last, we found authors with the most number of publications . The top 10 authors of this data-frame are davidreiss666, DoremusJessup, anutensil, igeldard, maxwellhill, readerseven, twolf1, madam1, nimobom and madazzahatter.

Then, we found the ratio of articles that are only for readers above 18 and those which can be read by anyone. Of all the articles, only 315 articles are for the above 18 audience and they constitute about 0.0631 % of total articles. The average length of title of article for above 18 readers and for any reader is 88.76 and 87.62. This represents that the data is highly skewed with respect to this feature and length of titles don't provide much insight about the type of article.

We even found the most commonly used words (excluding stop-word) words in a corpus which contains all words of every title . Few of the tops words in this list are :china', 'us', 'new', 'u', 's', 'syria', 'russia', 'world', 'police', 'government', 'israel', 'iran', 'a', 'president', 'killed', 'people', 'state', 'attack', 'in', 'war', 'russian', 'military', 'uk', 'north', 'year', 'south', 'korea'

Next ,we tried to predict the type of article ,ie if its only for above 18 authors or for everyone by using various machine learning and deep learning models. As this is a natural language processing task, we applied a naive-bias classification method. Naive-bias, SVM and linear classification methods are considered suitable for natural language related tasks, Though naive-bias was fast, its area under curve was really low and the best we could achieve was about 35%. We then tried to use a random-forest classification method. Random-forest and other decision tree related methods are not preferred for NLP related tasks. Time constraint of random

forest was really high and thus , did not allow us to try much tuning of its hyper-parameters, but even with 25 decision trees, this method gave about 75% area under-curve. Finally, we developed a basic RNN model (Recurrent Neural Network model) and used it for classification tasks. Even with 10 epochs, the recall was about 99% (using micro average technique). The method was fastest among all three methods.
At last, we developed a model that could generate text by learning the syntax and writing style of someone. We took all article titles of davidreiss666 .We got about 8100 article titles and using created, created our word corpus for training the model . We developed a simple model which though is quick to train, but requires high epochs. We used about 700 epochs and the training took about 2 hours. After the model was trained, we fed it with some random words from the generated corpus . The words were :"Habyarimana commander led Ireland". The generated text, which we wanted of 20 words was : "Habyarimana commander lead Ireland s family s soldiers sentenced on anti mafia city appears prosecutor said should be held to his presidential election chatting"

References

1. https://www.coursera.org/learn/natural-language-processing-tensorflow/home/welcome
2. https://www.udemy.com/course/spark-and-python-for-big-data-with-pyspark/learn/lecture/7047204#questions

Future-works:

With this data, we can start building clusters of words, based on their occurrence which can help in grouping writers into various groups. We can even classify article titles into various clusters .