DSE 308 : Computational Linguistics Final Project
Report

Fiction and Non-Fiction News Text Generation Using Markov Chains

Submitted by

Roll No	Names of Students
17054	Arman Kazmi
17103	Himanshu Chaudhary

Under the guidance of
Dr. Rajakrishnan P. Rajkumar

IISER Bhopal
Bhauri, Bhopal – 462066

Dec 6, 2020

Contents

Chapter 1

Introduction

With major advancement in Natural Language Processing (NLP), new algorithms have been able to mimic human writing to a greater extent. One such algorithm is GPT-3 [1] which is an autoregressive language model that has been trained on 175 billion parameters. GPT-3 can generate samples of news articles which human evaluators have difficulty distinguishing from articles written by humans.

Despite, the success of deep learning algorithms, they often require large amounts of training data to achieve a better accuracy. Not only, large amount of data but huge computation power is also required. Large datasets are not readily available for specific domain and can be time consuming for building one.

1.1 Objective

Our objective in this project is experimenting with Markov Chains to generate News text having a writing style similiar to that of Fiction or Non-Fiction texts . In this project we have focused only on Fictional and Non-Fictional genre with the Top-Headlines News Corpus.

1.2 Theory

1.2.1 Markov Model

A Markov chain[2] is a model that tells us something about the probabilities of sequences of random variables, states, each of which can take on values from some set. These sets can be words, or tags, or symbols representing

anything. A Markov chain makes a very strong assumption that if we want to predict the future in the sequence, all that matters is the current state.

More formaly, consider consider a sequence of state variables

$$q_1, q_2, ..., q_i.$$

A Markov model embodies the Markov assumption on the probabilities of this sequence: that Markov assumption when predicting the future, the past doesn't matter, only the present.

$$\mathbf{MarkovAssumption} : \mathbf{P}(\mathbf{q_i} = \mathbf{a}|\mathbf{q_1}, ..., \mathbf{q_i}) = \mathbf{P}(\mathbf{q_i} = \mathbf{a}|\mathbf{q_i} - \mathbf{1})$$

Here's an example, modelling the weather as a Markov Chain.[1]

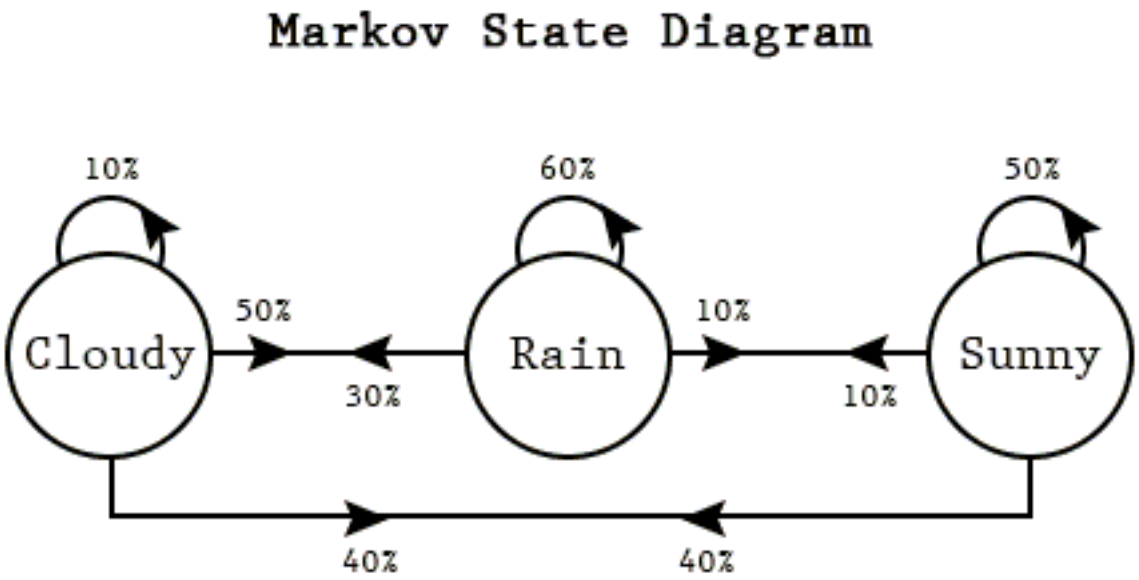

Figure 2

The transition matrix for the above state diagram in Figure 2 can be represented as below in Figure 3:

[1]https://techeffigytutorials.blogspot.com/2015/01/markov-chains-explained.html

Transition Matrix

	C	R	S
C	0.1	0.5	0.4
R	0.3	0.6	0.1
S	0.4	0.1	0.5

Figure 3

In the transition matrix rows and columns are the states that are shown in the above state diagram.
We can do calculations with the Matrix utilizing a State Vector(vector of our current conditions) to give us the probabilities of the next states. For example the probability of a sunny day after clouds is 0.4 which can be seen in the first rown and third column i.e. Row C and Column S - first value.

Chapter 2

Work Done

Markov chain can be applied for text generation where occurence of each word in a sentence can be predicted in terms of probability by using the current or previous words. In a paper[3]it was shown that the text generated using Markov chains was believed to be written by humans approximately 20-40 % of the time.

2.1 Implementation

2.1.1 Datasets

The datasets used in this work is scrapped from the news links provided by **Google-News-Api**[1] using **Newspaper-3k**[2] library from python. Since, Google-News-Api provides news links for different categories, for this work we chosed to work with Top-Headlines news data excluding Entertainment, Science, Technology and Sports categories, just to make sure the articles are in same domain. To produce genre specific text these articles were given a probability score of being Fiction and Non-Fiction using the algorithm of **Fictometer**[4] and then were used as corpus for training. Finally, the dataset consisted of three categories: *Non-Fictional top-headlines, Fictional top-headlines and top-headlines containing mixed Fictional and Non-Fictional articles.* These datasets were used seperately for training the Markov model.

[1]https://newsapi.org/s/google-news-api
[2]https://newspaper.readthedocs.io/en/latest/

2.1.2 Pre-Processing

The Non-fictional News articles corpus was created using 1900 articles and the rest of the two had 1400 articles. After Scrapping the main text of the articles, we removed sentence that were random and not related to the article such as many articles contained sentences such as *"Read More:", "Subscribe this news channel to get regular updates", "Click here to know your horoscope", "Share this article", "Advertisement", any links starting from "https://somelink", etc.* Also, symbols such as *"@", "#", "-", "(", "—", ")"* were removed from the corpus for training. However, symbols such as *".", "'", ",", "!", "?"* were not removed so as to make the generated corpus looks more human written.
After preprocessing the corpuses, we got 78,208 unique tokens for Non-Fictional news corpus, 34055 unique tokens for Fictional news corpus and 45728 unique tokens for mixed genre news corpus.

2.1.3 Method

To generate text using Markov chains we need to decide the states and the probabilities of jumping from each state to another.
Consider a sentence formed using a sequence of words

$$< start > w_1 w_2 w_3 ..... w_n < end >$$

where <start> and <end> denotes the starting and ending of a sentence. To predict a word w_i from the sequence, previous i-1 words can be considered as current states and model the probability of the word w_i.
In order to do this in our experiment, we created a vector for each distinct sequence of n words, having N components, where N is the total quantity of distinct words in the corpus. We then added 1 to the j_{t_h} component of the i_{t_h} vector, where i is the index of the i_{t_h} n-sequence of words, and j is the index of the next word. We normalized each word vector, to get a probability distribution for the next word, given the previous n tokens. This can be repeated k times to generate k length of sequence. This was implemented in the Python-3.7[3].

2.1.4 Toy Example

Consider the example sentence *"My name is Mister X"*, before training a Markov chain we need to decide the value of **n**: the number of tokens[4] our

[3]https://www.python.org/downloads/release/python-370/
[4]Anything between two spaces can be treated as token

chain will consider before predicting the next one.

To keep it simple, let us choose n = 1.
Next we need to find the total number of unique tokens in the training corpus. In our case, it has five unique tokens. We will initiate a 5×5 matrix filled with zeroes where rows and columns are the distinct tokens of the corpus. After that, we add 1 to the column corresponding to '*name*' on the row for '*My*'. Then another 1 on the row for '*name*', on the column for '*is*'. We continue this process until we have gone through the whole sentence.
The resulting matrix would be:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Since, our toy data has only single occurence tokens so the model would generate the same sentence again and again. Now, in our case the corpus has more than 35,000 unique tokens and by using different n values we created a transition matrix that gave significant results..

Chapter 3

Results and Discussions

To evaluate the performance of the model we generated text with different values of **n** with different length of sentences. But to run the Fictometer[4] algorithm for classification, the length of the sentences were kept above 50.

Generated sentences of length 50 with 1-word Markov chain i.e. $\mathbf{n = 1}$ and their score:-

1. **Model trained on Non-Fictional news corpus**:

   - ***The*** *labour and 13 disqualified legislators , after aide Eric Durr said . Cases of chewing sugar , respectively , 2019 and porters due to provide a major economies globally around the RdRp 14 hour , Nifty PSU will be held its spiritual head of cases , due to take oath* **Genre - Non-Fiction, Probability Score =0.003**

2. **Model trained on Fictional news corpus**:

   - ***The*** *cop pushed him not turn the sooner you can imagine selling . The dates as long argument over 100 in life . The Saturday hit China scrambles to find out during a nearby church ? Some e Business Summit held to you said . "It's almost paid for more recent* **Genre - Fiction, Probability Score =0.752**

3. **Model trained on Mixed genre news corpus**:

   - ***The*** *Government of his hopes . NCP leader Digvijaya Singh , Sushma Swaraj , 934 foreigners in India Ltd . " spreading lies , but to what happened on Monday as soon , artillery guns , respectively in the implementation of tennisfuture ceremonial . He will see... " he is* **Genre - Non-Fiction, Probability Score = 0.17**

The highlighted words in the above sentences are the seed words and the probability score is the probability of being Fiction, So a higher score means the text has more fictional writing.
Generated sentences of length 100 with 2-word Markov chain i.e. $\mathbf{n = 2}$ and their score:-

1. **Model trained on Non-Fictional news corpus**:

   - ***"He is*** *a high level pollution meeting earlier in the past 24 hours , the tally to 47 , 000 crore and Bank of India Ranjan Gogoi , who have officially tested positive for COVID 19 spread . The Prime Minister's Office AK Sharma , and a national address . After the riots . " The number of coronavirus positive cases are being keenly watched amid continued fears in a tweet , Prashant Kishor's tutoring is already helping solar and wind , which dates back to a record in two to three and further elevated following contact with the insecticide class which* **Genre - Non-Fiction, Probability Score = 0.001**

2. **Model trained on Fictional news corpus**:

   - ***He is*** *giving you directions , while maintaining social distance but decrease emotional or angry shared their grievances . He has the virusAustralia has over 130 cases of rioting and arson that took Delhi to avoid unnecessary travels and follow his daily routine which will not start now . Shefali Bagga named Vishal and Tamannaah . Akanksha Puri , assuring students protesting against the transfer must be identified and contact anyone from leaving to stop him unless the middle overs , it is more likely than not . "On the governmentś only concern was foreign intruders living in the situationánd there* **Genre - Fiction, Probability Score =0.752**

3. **Model trained on Mixed genre news corpus**:

   - ***He is*** *not just in case youd́ like to address , Modi said BJP had won a comfortable position to form the government . "Go back to domestic first class cricket , you can try to keep things under wraps , " Trump said on Sunday . "The issue erupted in Delhi's Shaheen Bagh , there had been closed for four categories of people in the Large Hadron Collider , the agitators put their hands on measuring the seasonal changes in the Lok Sabha member and an especially good place . We should tell you their stinger will fall in its decline* **Genre -Fiction, Probability Score = 0.86**

Using a 2-word chain in smaller sequence it does a pretty good job in compared to the 1-word markov chain model. In larger sequences the article loses its meaning and clearly the grammar and syntax is missing. While, in short sentences it sometimes generates better text.
However, when **n** was increased to 4 it gave better results:-

I have not slept *for two days and I was struggling to go to Mumbai , Ahmedabad , in Bangalore to check my back . I used to wake up and know I would fail . It was the need for her to see how she could chalk her path ahead and serve her people . She skipped the state BJP core committee meeting in Mumbai on Tuesday . Admitting that India were the better side on the day , agreed that the pitch was not the typical one found at the SCA Stadium , but felt the home team's total of 206 for* **Genre -Fiction, Probability Score = 0.99** - trained on Fictional News corpus
I have not slept *for two days and I was struggling to go to Mumbai , Ahmedabad , in Bangalore to check my back . I used to go through that and have different kinds of scans to actually know what is exactly happening with my back . "Unfortunately we didnt́ have those kind of machines at that time which would make clear that why I am getting back pain . It eventually turned out to be a black money recycling and political bribery scheme , the Congress on Wednesday called the bonds a " political extortion " and demanded an immediate statement from* **Genre -Fiction, Probability Score = 0.99** - trained on mix news articles.

Increasing **n** value above 5 produced more deterministic text as 5-word-chain or above was unique in the training corpus and hence produced deterministic results.

3.1 Genre specific generation

To test the performance of model generating a fiction based or non-fiction based article, we generated **100** samples of text, of length 100 and noted down the count of fiction and non-fiction articles. Score > 0.75 was treated as Fiction and Score < 0.50 as Non-Fiction and any score between them was rejected.
This process was repeated five times and hence a total of **500** samples of text were generated from each corpus.

The results on the three corpus are shown in the below three table.

Table 3.1: Model trained on Fictional corpus

Iterations.	No. of Non Fiction articles	No. of Fiction articles
Iteration 1	29	69
Iteration 2	29	67
Iteration 3	31	67
Iteration 4	23	73
Iteration 5	28	66

Table 3.2: Model trained on Non-Fictional corpus

Iterations.	No. of Non Fiction articles	No. of Fiction articles
Iteration 1	95	5
Iteration 2	93	6
Iteration 3	96	4
Iteration 4	97	3
Iteration 5	99	1

Table 3.3: Model trained on Mixed corpus

Iterations.	No. of Non Fiction articles	No. of Fiction articles
Iteration 1	57	38
Iteration 2	66	28
Iteration 3	52	40
Iteration 4	65	31
Iteration 5	65	32

On an average 69 fictional articles were generated using fictional corpus, and 96 non-fictional articles were generated using non-fictional articles. Using Mixed corpus the model generated on an average 61 non-fictional articles and 33 fictional articles as it can be seen in the below table.

Now based on this we calculated the probability of an article A generated is:

1. Fictional using:
    - **fictional corpus** is **69 %**,
    - **non-fictional corpus** is **4 %**,
    - **mixed corpus** is **33.8 %**
2. Non-Fictional using:
    - **fictional corpus** is **28 %**,
    - **non-fictional corpus** is **96 %**,
    - **mixed corpus** is **61 %**

Chapter 4

Conclusion and Future work

Using a Markov Chain as a model for text generation is a great introduction into the domain. This type of model provides a simple low complexity solution for generating text.

However, to generate a specific genre article, choice of corpus do matter. One can generate more fictional articles using fictional corpus and mixed corpus while there is a less probability of generating fictional articles using non-fictional corpus.
One interesting thing to note here is that one cannot produce always a fictional article using a fictional corpus, because the probability is only **69 %** but that is not the case with non-fictional corpus because the probability of being an article non-fiction is **96 %** and one should expect it too.

The text generated had no coherence as syntax and grammars were completely missing but the model performed well on short sentences which can be treated as a headline of a news because in general the news headlines are short.

In future, we would like to experiment the model by adding ***context-free-grammars*** to improve the syntax and the coherence of the sentences.
Also, it would be interesting to see if a simple Markov chain can produce specific category or domain related articles with improved grammars.

Acknowledgments

We thank Dr. Rajakrishnan P. Rajkumar for giving this opportunity to take this project and for the course on Computational Linguistics. We also thank Mr. Siddharth Ranjan for providing us the support we needed to complete this project.

References

[1] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, Dario Amodei "**Language Models are Few-Shot Learners**"

[2] Daniel Jurafsky James H. Martin **"Speech and Language Processing "**,

[3] Daniel R. M. Everett, J. R. Nurse, and A. Erola, **"The anatomy of online deception: What makes automated text convincing?,"** in *Proceedings of the 31st Annual ACM symposium on applied computing*, pp. 1115–1120, ACM, 2016.

[4] M. R. Qureshi, S. Ranjan, Rajakrishnan P. R. and K. Shah, *StoryNLP @ ACL 2019* **"A Simple Approach to Classify Fictional and Non-Fictional Genres"**,