

# **PREDICTION OF COVID MUTATION USING NEURAL NETWORK**

**A MAJOR PROJECT REPORT**

*Submitted by*

CH.EN.U4AIE20001

ANBAZHAGAN E

CH.EN.U4AIE20053

RAMYA POLAKI

CH.EN.U4AIE20069

TRINAYA KODAVATI

CH.EN.U4AIE20031

SMITHIN REDDY K

CH.EN.U4AIE20035

ABHIRAM KUNCHAPU

*in partial fulfilment for the award of the degree*

*Of*

**BACHELOR OF TECHNOLOGY**

**IN**

**INTELLIGENCE OF BIOLOGICAL SYSTEMS**



**AMRITA SCHOOL OF ENGINEERING, CHENNAI**

**AMRITA VISHWA VIDYAPEETHAM**

CHENNAI – 601103, TAMIL NADU

**AMRITA VISHWA VIDYAPEETHAM**

**AMRITA SCHOOL OF ENGINEERING, CHENNAI, 601103**



**BONAFIDE CERTIFICATE**

This is to certify that the major project report entitled “**PREDICTION OF COVID MUTATION USING RECURRENT NEURAL NETWORK**” submitted by

CH.EN.U4AIE20001

ANBAZHAGAN E

CH.EN.U4AIE20053

RAMYA POLAKI

CH.EN.U4AIE20069

TRINAYA KODAVATI

CH.EN.U4AIE20031

SMITHIN REDDY K

CH.EN.U4AIE20035

ABHIRAM KUNCHAPU

in partial fulfilment of the requirements for the award of the **bachelor of Master of Technology** in **COMPUTER SCIENCE ENGINEERING** is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Engineering, Chennai.

Signature

I.R.Oviya

CSE Dept Assistant Professor

This project report was evaluated by us on .....

INTERNAL EXAMINER

EXTERNAL EXAMINER

## **ACKNOWLEDGEMENT:**

I offer my sincere pranams at the lotus feet of Universal guru, MATA AMRITANANDAMAYI DEVI who blessed me with her grace to make this a successful project.

I express my deep sense of gratitude to Dr. P Shankar, Principal, Amrita School of Engineering, Chennai for his kind support. I would like to extend my gratitude and heartfelt thanks to Mrs. I.R.Oviya ,CSE Department , for his constant help, suggestions and inspiring guidance. I am grateful to my guide, Department of CSE , for his invaluable support and guidance during the course of the major project work. I am also indebted to Dr. Prasanna Kumar, Chairperson of Department of AI, ASE, Chennai for his guidance.

I specially thank our director Sri I B Manikandan who supported us in every possible manner.

## INDEX:

| Topic                         | Page. No |
|-------------------------------|----------|
| 1. Abstract-----              | 5        |
| 2. Introduction-----          | 6        |
| 2.1 DNA Sequencing-----       | 6        |
| 2.2 Neural Networks-----      | 7        |
| 2.3 Word Embedding-----       | 7        |
| 3. Literature Review-----     | 9        |
| 4. Methodology-----           | 11       |
| 4.1 Data Collection-----      | 11       |
| 4.2 Data Pre-processing-----  | 14       |
| 4.3 NN Embedding Model-----   | 14       |
| 5. Discussion and Result----- | 16       |
| 6. Conclusion-----            | 19       |
| 7. Future Scope-----          | 19       |
| 8. References-----            | 19       |

## **1. ABSTRACT:**

SARS-CoV-2, a new coronavirus also known as COVID-19, has caused a worldwide pandemic. This contagious RNA virus has now paralysed the entire earth. In the current global pandemic scenario, it is critical to anticipate SARS-CoV-2 as soon as possible because both the number of afflicted and fatality cases are increasing exponentially every day. This RNA virus is capable of causing mutations in the human body. Accurately determining mutation rates is critical for understanding the development of this virus and determining the risk of emergent infectious illness, the polymorphism character of SARS-CoV-2 allows it to adapt and survive in a variety of environments, making SARS-CoV-2 extremely difficult to anticipate. In such scenarios, this project can be very useful for predicting SARS-COV-2, because it employs an alignment-free technique, the study has the potential to be very beneficial for rapidly forecasting SARS-CoV-2 and other types of deadly viruses based on their genetic content.[8]

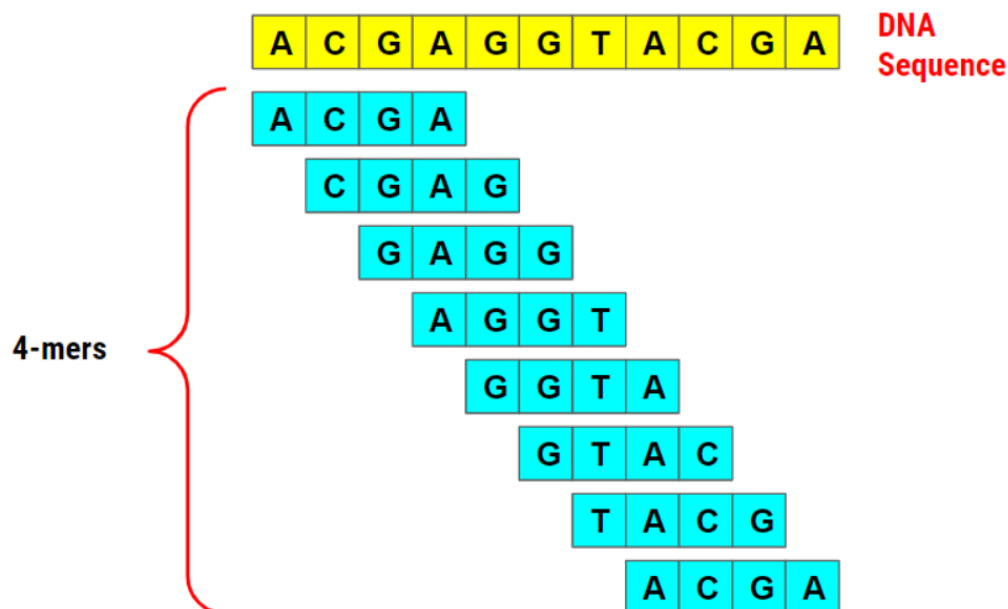
## 2. INTRODUCTION:

### 2.1 DNA Sequencing:

The method of establishing the order of nucleotides (the nucleic acid sequence) in DNA is known as DNA sequencing. It refers to any method or technology for determining the order of the four bases: adenine, guanine, cytosine, and thymine.

Sequencing enables physicians to detect whether a gene or the area that regulates a gene includes changes known as variations or mutations that are associated to an illness.

In bioinformatics, k-mers are long substrings inside a biological sequence, which are formed of nucleotides (A, T, G, and C). If a kmer of a DNA sequence data of length  $k$ , any sequencing read of length  $n$  can be broken up into a set of overlapping  $n-k+1$  kmers.



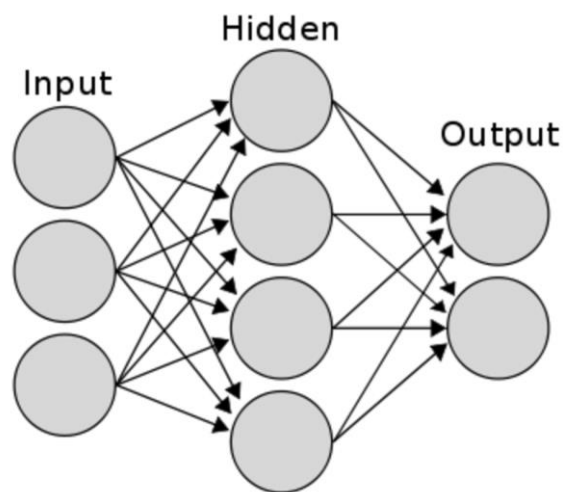
*Fig 2. 1 Kmer Over Lapping Technique*

Kmer analysis is used in bioinformatics as a tool to understand the composition of a sequencing library and identify reads containing errors. They are mostly employed in the area of computational genomics and sequence analysis to assemble DNA sequences, improve heterologous gene expression, identify species in metagenomic materials, and generate attenuated vaccines.

## 2.2 Neural Networks:

Predictive analytics, as we all know, utilises techniques like as predictive modelling and machine learning to examine past data in order to forecast future patterns. However, neural networks are not the same as traditional predictive tools. Because of the hidden layers that improve prediction accuracy, neural networks perform better in predictive analytics.

A neural network — of which recurrent neural networks are one type, among other types such as convolutional networks — is composed of three elementary components: the input layer, the hidden layers, and the output layer. Each layer consists of so-called nodes (neurons).



*Fig 2. 2 Simple Neural Network*

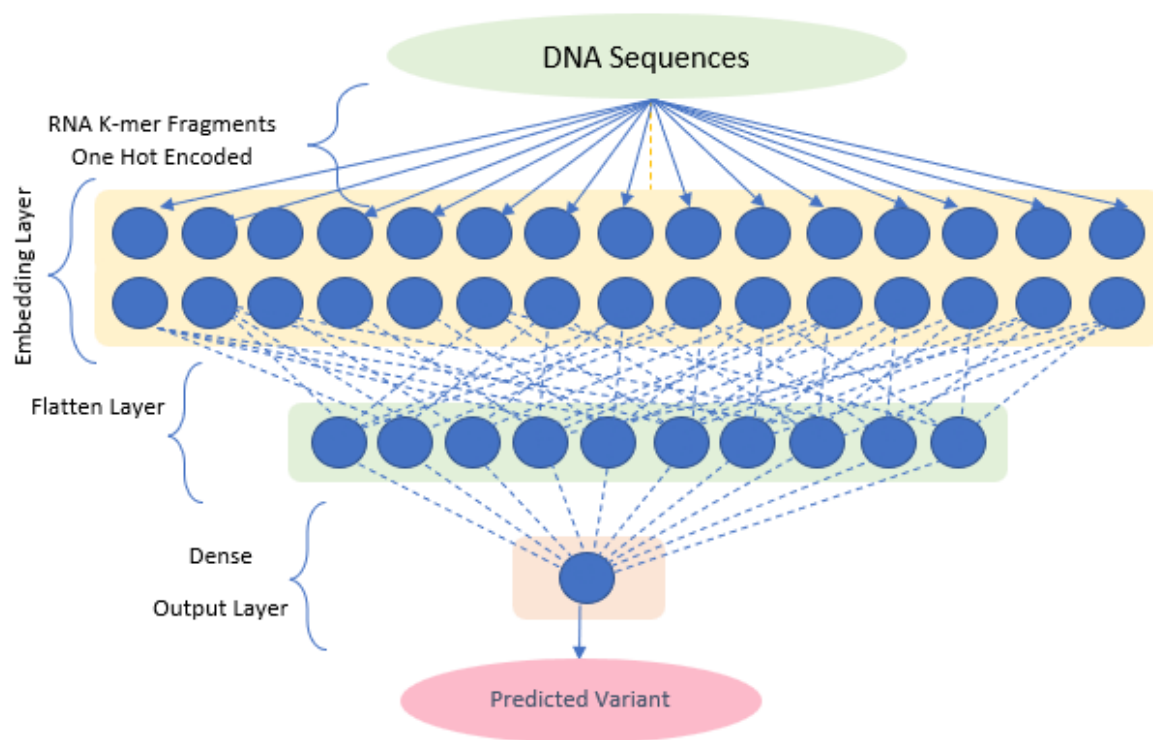
## 2.3 Word Embedding:

Since models do not interpret text or image data as directly as people do, the final level data must be in numerical form in order to develop any model in machine learning or deep learning. We need sophisticated ways to turn text input into numerical data, which is known as vectorization or word embeddings in the NLP area.

Vectorization, also known as word embedding, is the process of transforming text data to numerical vectors. The numerical vectors are then utilized to construct various machine learning models. In some ways, this is similar to extracting features from text in order to develop different natural language processing models.[11]

## Embedding Layer

An embedding layer is a word embedding learned in conjunction with a neural network model for a specific natural language processing task, such as language modelling or document categorization.[12] The embedding layer allows us to transform each word into a fixed length vector of defined size. Instead of merely 0's and 1's, the resultant vector is dense, with real values. Word vectors have a defined length, which allows us to better represent words while reducing their size.



*Fig 2. 3 Our Neural Network Model with Embedding Layer*

It necessitates the cleaning and preparation of document text so that each word can be encoded in a single pass. The model specifies the size of the vector space, which can be 50, 100, or 300 dimensions. Small random integers are used to start the vectors. The embedding layer is utilized on the front end of a neural network and is fitted with the Backpropagation method in a supervised manner.

The word vectors are mapped to the one-hot encoded words. When using a multilayer Neural model, the word vectors are concatenated before being sent into the model as input. Each word could be treated as one input in a sequence if a recurrent neural network is used.



This method of learning an embedding layer necessitates a large amount of training data and is time consuming, but it will produce an embedding that is both targeted to the specific text data and the NLP goal.[12]

### **3. LITERATURE REVIEW :**

The efficiency of antiviral medications is still hampered by viral evolution. The capacity to forecast this evolution might aid in the early diagnosis of drug-resistant viruses and the development of more effective antiviral medicines. Genome research have used a variety of approaches to attain this objective. Machine learning is one of these techniques, which makes it easier to explore structure-activity connections, anticipate secondary and tertiary structure evolution, and rectify sequence errors. One of the papers , provides us a unique machine learning approach for predicting potential point mutations in basic RNA sequence structure alignments. It shows that a nucleotide in an RNA sequence varies dependent on the other nucleotides in the sequence and predicts the genotype of each nucleotide in the sequence. [3]

To anticipate novel strains, neural networks are used, and then a rough set theory-based method is used to extract these point mutation patterns. On a number of aligned RNA isolates time-series species of the Newcastle virus, this approach is used. The validation of these methodologies is achieved by using two distinct data sets from two different sources. The results has been demonstrated that this approach is capable of predicting nucleotides in the new generation with a 75 percent accuracy. For the investigation of the correlation between distinct nucleotides in the same RNA sequence, the mutation rules are visualized. [3]

Some other researches says that the most essential reasons for protein design is the capacity to forecast how protein stability will vary when it is mutated. Several strategies for tackling this problem have been published, and their effectiveness has been evaluated using a global linear correlation between anticipated and experimental data. There is no direct statistical evaluation of their prediction performance, nor is there a direct comparison between various methodologies. A large collection of thermodynamic data on protein stability alterations caused by single point mutations was recently created (ProTherm). This permits machine learning techniques to be used to anticipate changes in free energy stability due to mutations starting with the protein sequence. [4]

They depicted a neural-network-based technique for predicting whether a particular mutation boosts or lowers protein thermodynamic stability in comparison to the native structure in this

research. And their model properly classifies >80% of the mutations in the database, based on a dataset of 1615 mutations. And it outperforms other approaches accessible on the web on the same job and with the same data. Furthermore, when this approach is combined with energy-based methods, the joint prediction accuracy rises to 90%, suggesting that it may be utilized to improve the performance of existing methods as well as protein design strategies in general. [4]

As we know, Viruses are capable of having rapid mutation in high rates[8], which allows them to generate drug-resistant variants faster. Understanding the viral adaptability process and designing medications that successfully fight possibly resistant mutants can both benefit from mining relevant rules from mutation data. So some papers were based on drug designing. In order to do so they had to try out with mutation prediction techniques. And one of them proposed a simple statistical relational learning strategy for mutant prediction, in which the input consists of mutation data with drug-resistance information, either as sets of mutations imparting drug resistance or as sets of mutants with drug susceptibility information. The algorithm generates a library of possibly resistant mutants by learning a set of relational principles that characterize drug resistance. Learning a weighted mixture of rules allows for the attachment of created mutants to a resistance score predicted by the statistical relational model and the selection of just the highest scoring mutants. [1]

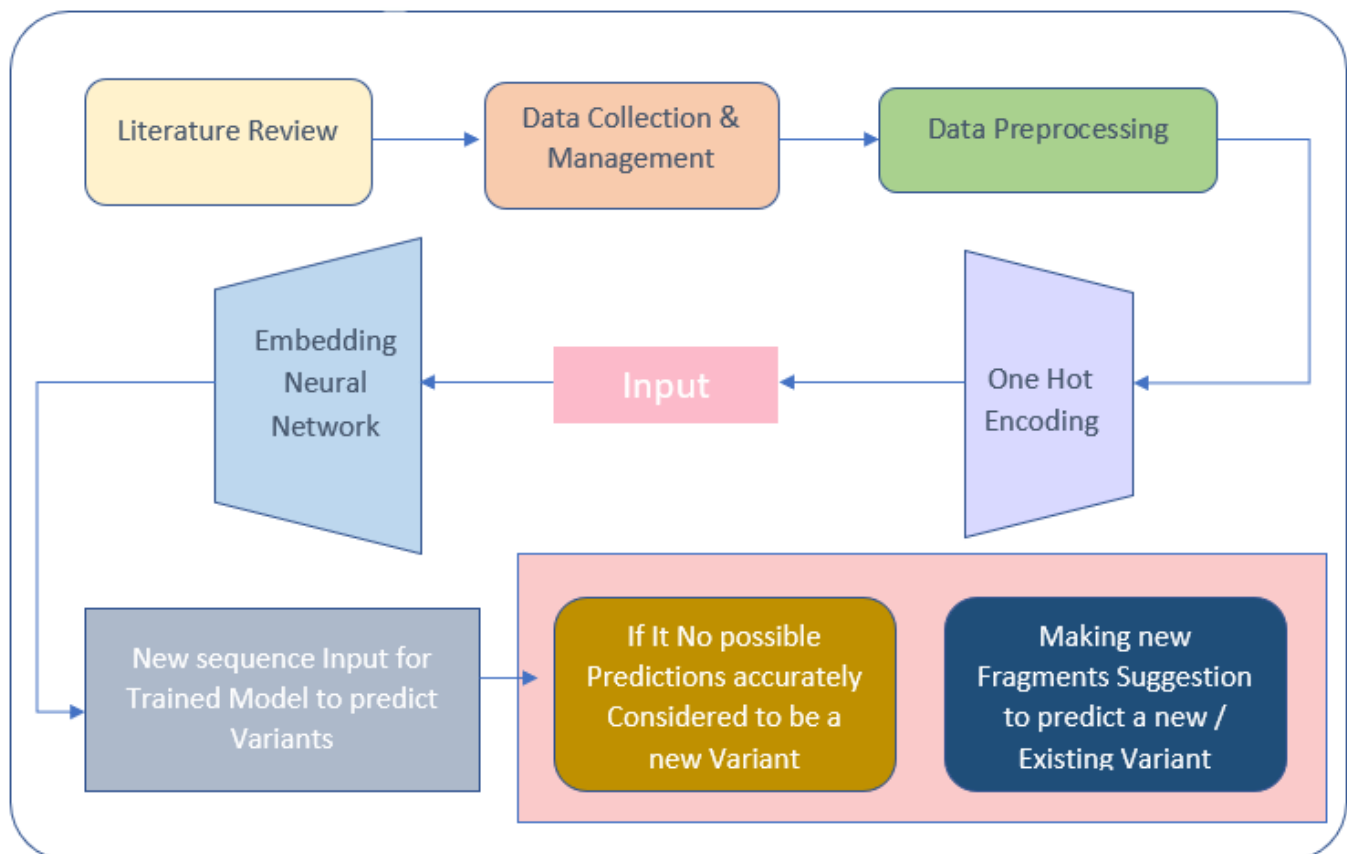
Yet , the primary objective in this research is to accurately forecast the effects of genetic variation.[8] Several machine learning algorithms have been created to learn information from evolutionary sequence data in order to achieve this aim. So some have involved with a deep generative model based on the variational autoencoder (VAE) that models distributions using a latent variable has proven to be the most effective way thus far. One of them , presents a mutationTCN, a deep autoregressive generative model for simulating inter-residue correlations in biological sequences that uses dilated causal convolutions and an attention mechanism. [2]

They have evaluated against a collection of 42 high-throughput mutation scan tests, we show that this model is competitive with the VAE model, with a mean improvement in Spearman rank correlation of 0.023. In comparison to the latent variable model, our model can more efficiently collect information from numerous sequence alignments with a reduced effective number of sequences, such as in viral sequence families. They also expand this architecture to a semi-supervised learning system with excellent prediction accuracy. They show that our

approach allows for direct data likelihood optimization as well as a simple and consistent training method. [2]

So, by looking up all these researches which has been proposed prior to our project depicts a good example for going further. So , in our project we will try to predict the SARS-CoV2 variants with Neural Network .

## 4. METHODOLOGY



*Fig. 4 Work Flow*

### 4.1.Data Collection

As an Initial Step of our model , we had to collect and create our own dataset as we didn't find any suitable dataset .So , NCBI Virus Database helped us in obtaining Covid Nucleotide Fasta Sequences in terms of Accession ID List as in Fig 4.1 to .csv data as in Fig .4.2. Now we had to extract the Sequences from NCBI directly using Accession-ID List [5] . BioPython module in python allowed us to manifest this extraction of fasta sequences using SeqIO API [6] .

**NCBI Virus**  
Sequences for discovery

About Us ▾ Find Data ▾ Help ▾ How to Participate ▾ S

**SARS-CoV-2 Data Hub** **Download ▾**

Quick Links: Betacoronavirus BLAST, SARS-CoV-2 Article, CDC Outbreak Information, PubMed, SRA Data

**Tabular View** Dashboard Visualizations Mutations in SRA Complete Tree

Selected Results: 0

**Refine Results** [Reset](#)

Virus +

Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), taxid:2697049 x

**Nucleotide (2,988,422)** **Protein (18,021,365)** **RefSeq Genomes**

**Accession** **Submitters** **Release Date** **Pangolin**

|                          |                          |                |            |   |
|--------------------------|--------------------------|----------------|------------|---|
| <input type="checkbox"/> | <a href="#">OM033239</a> | Feehan,A., ... | 2021-12-28 | L |
| <input type="checkbox"/> | <a href="#">OM033240</a> | Feehan,A., ... | 2021-12-28 | L |

*Fig 4. 1 NCBI DataHub*

|                        |                     |                        |              |
|------------------------|---------------------|------------------------|--------------|
| Alpha                  | 12/25/2021 11:42 AM | Microsoft Excel Com... | 3,891 KB     |
| Beta                   | 12/25/2021 11:42 AM | Microsoft Excel Com... | 448 KB       |
| CovidVariantDataset    | 12/25/2021 10:29 PM | Microsoft Excel Com... | 2,672,742 KB |
| Delta                  | 12/25/2021 11:42 AM | Microsoft Excel Com... | 1,977 KB     |
| Eta                    | 12/25/2021 11:42 AM | Microsoft Excel Com... | 200 KB       |
| FragmentedCovidDataset | 12/26/2021 9:36 PM  | Microsoft Excel Com... | 2,662,063 KB |
| Gamma                  | 12/25/2021 11:42 AM | Microsoft Excel Com... | 813 KB       |
| Kappa                  | 12/25/2021 11:42 AM | Microsoft Excel Com... | 70 KB        |
| Omicron                | 12/25/2021 11:42 AM | Microsoft Excel Com... | 1 KB         |








*Fig 4. 1 Accession ID List of Variants*

In order to proceed further we should know what is Pango Lineage(PangoLin).PANGOLIN (Phylogenetic Assignment of Named Global Outbreak Lineages) is a software tool developed by members of Andrew Rambaut's team, with an accompanying online application produced by the Centre for Genomic Pathogen Surveillance in South Cambridgeshire. [7]Its goal is to employ the PANGO nomenclature to categorise genetic lineages of SARS-CoV-2, the virus that causes COVID-19. A user can use the application to submit a whole genome sequence of a SARS-CoV-2 sample, which is then compared to other genome sequences and the most likely lineage assigned (PANGO lineage).[7]

| Nextstrain Clade | Pango Lineage   | WHO Label          |
|------------------|-----------------|--------------------|
| Alpha            | B.1.17          | Alpha $\alpha$     |
| Beta             | B.1.351         | Beta $\beta$       |
| Gamma            | P.1             | Gamma $\gamma$     |
| Delta            | B.1.617.2       | Delta $\delta$     |
| Kappa            | B.1.617.1       | Kappa $\kappa$     |
| Epsilon          | B.1.427,B.1.429 | Epsilon $\epsilon$ |
| Eta              | B.1.525         | Eta $\eta$         |
| Iota             | B.1.526         | Iota $\iota$       |
| Lambda           | C.37            | Lambda $\lambda$   |
| Mu               | B.1.621         | Mu $\mu$           |
| Omicron          | BA.1            | Omicron $o$        |

*Table. 4. 1 Variants and its Panglin [7]*

According to Pangolin in Table .4.1 [7] , We can notice that every distinct variant is provided with unique Pango Lineage. NCBI has a Filter to segregate the variants with Pangolin[5]. So then , we downloaded the sequences and stored it in .csv format as in Fig.4.3.

|   |                     |                        |              |
|---|---------------------|------------------------|--------------|
|  AlphaBP   | 12/25/2021 8:17 PM  | Microsoft Excel Com... | 1,442,954 KB |
|  BetaBP    | 12/25/2021 4:34 PM  | Microsoft Excel Com... | 152,397 KB   |
|  DeltaBP   | 12/25/2021 4:35 PM  | Microsoft Excel Com... | 665,792 KB   |
|  EtaBP     | 12/25/2021 1:06 PM  | Microsoft Excel Com... | 68,220 KB    |
|  GammaBP   | 12/25/2021 8:27 PM  | Microsoft Excel Com... | 318,733 KB   |
|  KappaBP   | 12/25/2021 4:34 PM  | Microsoft Excel Com... | 23,820 KB    |
|  OmicronBP | 12/25/2021 11:45 AM | Microsoft Excel Com... | 263 KB       |

*Fig 4. 2 Sequence Dataset of Variants*

## 4.2 Data Pre-Processing


















### Fragmentation :

Before training the model , we have to convert the sequences in the form sentences by fragmenting it into k-mers, preferably k is at least 5. K-mers is nothing but a sub-sequence with certain size of k in a sequence.

|       | Accession  | Sequence  | Variant |
|-------|------------|---|---------|
| 0     | OK234379.1 | CTTTC TTTCG TTCGA TCGAT CGATC GATCT ATCTC TCTC... | alpha   |
| 1     | OK511600.1 | CTTTC TTTCG TTCGA TCGAT CGATC GATCT ATCTC TCTC... | alpha   |
| 2     | OL532995.1 | TAAAG AAAGG AAGGT AGGTT GGTTT GTTTA TTTAT TTAT... | alpha   |
| 3     | OK218986.1 | AACCT ACTTT CTTTC TTTCG TTCGA TCGAT CGATC GATC... | alpha   |
| 4     | OK220836.1 | AACCT ACTTT CTTTC TTTCG TTCGA TCGAT CGATC GATC... | alpha   |
| ...   | ...        | ...   | ...     |
| 10173 | OL901854.1 | ACCAA CCAAC CAACC AACCA ACCAA CCAAC CAACT AACT... | omicron |
| 10174 | OL902308.1 | CAACT AACCT ACTTT CTTTC TTTCG TTCGA TCGAT CGAT... | omicron |
| 10175 | OV269084.1 | NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNN... | omicron |
| 10176 | OV269229.1 | NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNN... | omicron |
| 10177 | OV269330.1 | NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNNN NNNN... | omicron |

*Fig 4. 4 Fragmentation of Sequences*

So fragmenting them in order , will represent the sequence as a sentences with len(sequence)-(k-1) words as in Fig 4.4 and also, it's been done to all the variant datasets and stored as in Fig.4.5

|   |                    |                        |              |
|---|--------------------|------------------------|--------------|
|  AlphaFragment1  | 12/26/2021 8:43 PM | Microsoft Excel Com... | 1,046,613 KB |
|  AlphaFragment2  | 12/26/2021 8:42 PM | Microsoft Excel Com... | 1,046,839 KB |
|  AlphaFragment3  | 12/26/2021 8:43 PM | Microsoft Excel Com... | 1,046,905 KB |
|  BetaFragment1   | 12/26/2021 8:15 PM | Microsoft Excel Com... | 523,767 KB   |
|  BetaFragment2   | 12/26/2021 8:14 PM | Microsoft Excel Com... | 847,240 KB   |
|  DeltaFragment1  | 12/26/2021 8:35 PM | Microsoft Excel Com... | 1,047,449 KB |
|  DeltaFragment2  | 12/26/2021 8:34 PM | Microsoft Excel Com... | 1,048,188 KB |
|  DeltaFragment3  | 12/26/2021 8:35 PM | Microsoft Excel Com... | 1,047,984 KB |
|  DeltaFragment4  | 12/26/2021 8:33 PM | Microsoft Excel Com... | 1,048,528 KB |
|  DeltaFragment5  | 12/26/2021 8:35 PM | Microsoft Excel Com... | 1,049,744 KB |
|  DeltaFragment6  | 12/26/2021 8:34 PM | Microsoft Excel Com... | 747,832 KB   |
|  EtaFragment     | 12/26/2021 8:15 PM | Microsoft Excel Com... | 613,702 KB   |
|  GammaFragment1  | 12/26/2021 8:14 PM | Microsoft Excel Com... | 914,641 KB   |
|  GammaFragment2  | 12/26/2021 8:14 PM | Microsoft Excel Com... | 915,809 KB   |
|  GammaFragment3  | 12/26/2021 8:15 PM | Microsoft Excel Com... | 1,036,939 KB |
|  KappaFragment   | 12/26/2021 8:13 PM | Microsoft Excel Com... | 214,287 KB   |
|  OmicronFragment | 12/26/2021 8:13 PM | Microsoft Excel Com... | 2,358 KB     |

*Fig 4. 3 Fragmented Dataset of Variant*

## One Hot Encoding & Padding :

In one-hot encoding, every word (including symbols) in the provided text input is encoded as a vector of just 1 and 0 values. As a result, a one-hot vector is a vector with just one and zero entries. Each word is printed or encoded as a unique one-hot vector. This permits the one-hot vector to uniquely identify the word and vice versa, ensuring that no two words have the same one-hot vector representation.

In Keras , for Natural Language Processing , there sufficient tools to work with. In preprocessing module , one of them for text manipulation is ” one\_hot“ function. This function receives as input a string of text and returns a list of encoded integers each corresponding to a word (or token) in the given input string as in Fig 4.6

```
[[47 37 41 ... 0 0 0]
 [47 37 41 ... 0 0 0]
 [31 34 35 ... 0 0 0]
 ...
 [46 46 46 ... 0 0 0]
 [46 46 46 ... 0 0 0]
 [46 46 46 ... 0 0 0]]
```

*Fig 4. 4 One-Hot encoded and Padded vectors*

When we apply one hot to a sequence of words or text (as shown in Fig 4.4), we receive a list of list indices, each of which may be of different lengths. This is an issue for keras and numpy, which expect the list of lists to be in array-like shape - that is, each sub-list to be the same, fixed length.

So, the pad sequences function is used to make each of the sub-lists a fixed length by appending zeros to the ends of all lists as shown in Fig 4.6

### 4.3 NN Embedding Model

Keras[13] provides an Embedding layer for neural networks on text data. It requires integer encoding of the input data, with each word represented by a distinct value. The Tokenizer API, which comes with Keras, may be used to execute this data pre-processing step. The Embedding layer starts with random weights and learns an embedding for every word in the training dataset.[13]

It's a versatile layer that may be utilised in a number of ways like it can be used alone to learn a word embedding that can then be stored and reused in a subsequent model. And also it can be used as part of a deep learning model that learns the embedding as well as the model. And it is possible to use it, to load a word embedding model that has already been trained, which is a sort of transfer learning.

And in our project, we are using embedding layer as the input layer of the Neural Network and embedding the word vectors to give out the output to next flatten layer , which will ensure that it converts multi- dimensional data from previous layers to single Dimension. Finally we have a dense output layer as in Fig 4.7.

| Model: "sequential"         |                    |         |  |
|-----------------------------|--------------------|---------|--|
| Layer (type)                | Output Shape       | Param # |  |
| embedding (Embedding)       | (None, 30252, 100) | 10000   |  |
| flatten (Flatten)           | (None, 3025200)    | 0       |  |
| dense (Dense)               | (None, 1)          | 3025201 |  |
| Total params: 3,035,201     |                    |         |  |
| Trainable params: 3,035,201 |                    |         |  |
| Non-trainable params: 0     |                    |         |  |
| None                        |                    |         |  |

*Fig 4. 5 Summary of our Model*

## 5. DISCUSSION AND RESULT :

After training the model , we have achieved a low accuracy as in Fig 5.1. We faced several issues while training and pre-processing our model. The datasets which we created were so huge in order to preprocess and train. We had to reduce the data size in order to avoid Over use and Clashing of RAM in our personal computers.

1. Converting the Sequence to fragments has multiple the size of data enormously as we know about the length of a DNA Sequence.



2. And also when we do on hot encoding , the max words in dictionary have to be greater than no. of fragments in largest sequence of our dataset. We could not achieve this.
3. The padding of encoded word vector also will increase the size
4. From section 2.2 we know that ,embedding layers requires a large amount of training data and is time consuming , as reduction in data due to insufficient RAM has reduced accuracy.
5. While embedding the words in embedding layer, we have to mention the output vector size of the embedding layer to next layer .And length of this vector has to be equal or greater than the length of padded x data. We could not achieve this as this will increase the no. of parameters to be stored i.e., weights of neurons to be stored in large number. We can see no. of parameters in Fig 4.7 of section 4.3
6. In mainly , we could not afford the RAM size to train our model

So , collectively , It all has resulted in low accuracy. As we didn't achieve what we expected stopped our research here and we would like to discuss about this further in Future Scope.

If we are succeeding in getting desired accuracy , we can use our model to predict the output by giving a sequence and verify for accuracy. When we it is a new mutation , it will result in new fragments, and our model prediction accuracy might be low. So that time it can be a new variant . Another is we can use algorithm to get next suggestion of fragments to the prior fragments and generate new sequences and comparing it with other input sequence might tell us about its variant type.

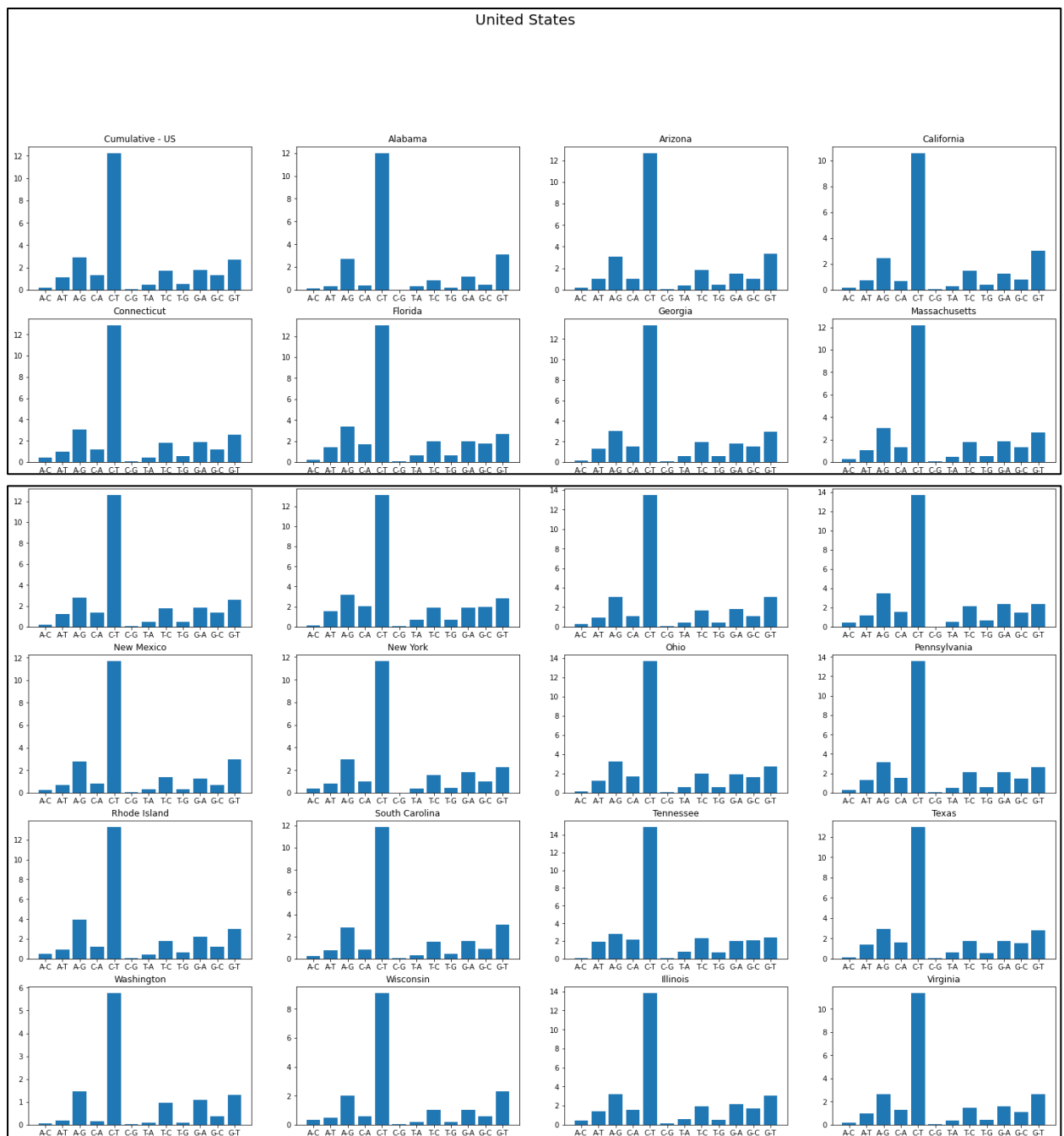
```
<keras.callbacks.History at 0x7effead89d90>

[24] # evaluate the model
      loss, accuracy = model.evaluate(x,y)
      accuracy

319/319 [=====] - 3s 8ms/step - loss: 0.0000e+00 - accuracy: 0.1474
0.14737670123577118
```

*Fig 5. 1 Accuracy of Model After Training*

In Fig.5.1 , we have found that most of the mutations in SARS-CoV2 genome was conversion of cytidine to Thymine . That data is from the United States [9] region depicting the frequencies of single mutation happened in SARS-CoV2 genome. And also, we could notice the 1<sup>st</sup> Subplot , a cumulative / General representation of Point Mutation.[10]



*Fig 5. 2 Frequency of Point Mutation in Samples in Different States of US*

## 6. CONCLUSION

In the verge of research , we have built our model using Embedding layers which can convert the DNA fragments to vectors and it predicts the variant. As Covid – 19 is being a threat to lives , it is necessary for us to do proper medication . Even though , scientists and researchers working on the drugs , it will be more fortunate to predict the next major variant if it can affect the humans as succeeding wave. In our project we tried to accomplish the prediction, but we could not get the desired outputs we will work on the algorithms and Mathematics to fight together against the global pandemic as in “ Precaution is better than Cure “.

## 7. FUTURE SCOPE

As we discussed in section 5 , due to the low accuracy , we found this algorithm requires high computation , time and storage. So , in future we would like to try with either with a good RAM space and a high computing devices or GPUS to train our model or we would like to work on algorithms like LSTM [14],GRU[14], BERT[11] and Autoencoders. We hope our research in future will yield good results to breakthrough the pandemic.

## 8. REFERENCES :

1. Cilia, E., Teso, S., Ammendola, S. *et al.* Predicting virus mutations through statistical relational learning. *BMC Bioinformatics* **15**, 309 (2014). <https://doi.org/10.1186/1471-2105-15-309>
2. Ha Young Kim, Dongsup Kim, Prediction of mutation effects using a deep temporal convolutional network, *Bioinformatics*, Volume 36, Issue 7, 1 April 2020, Pages 2047–2052,
3. Salama MA, Hassanien AE, Mostafa A. The prediction of virus mutation using neural networks and rough set techniques. *EURASIP J Bioinform Syst Biol.* 2016 May 13;2016(1):10. doi: 10.1186/s13637-016-0042-0. PMID: 27257410; PMCID: PMC4867776.
4. Emidio Capriotti, Piero Fariselli, Rita Casadio, A neural-network-based method for predicting protein stability changes upon single point mutations, *Bioinformatics*, Volume 20, Issue suppl\_1, 4 August 2004, Pages i63–i68, <https://doi.org/10.1093/bioinformatics/bth928>
5. Hatcher EL, Zhdanov SA, Bao Y, Blinkova O, Nawrocki EP, Ostapchuck Y, Schäffer AA, Brister JR. Virus Variation Resource - improved response to emergent viral outbreaks. National Library of Medicine, SARS-CoV-2 Data Hub , [NCBI Virus](https://www.ncbi.nlm.nih.gov/virus/) Nucleic Acids Res. 2017 Jan 4;45(D1):D482-D490. doi: 10.1093/nar/gkw1065. Epub 2016 Nov 28. PMID: 27899678; PMCID: PMC5210549.,

6. Biopython, , Introduction to SeqIO API Documentation, Introduction to SeqIO , [Introduction to SeqIO · Biopython](#)
7. Emma Hodcroft, CoVariants,-GISAID data,(28 December,2021), [CoVariants](#)
8. Raskin S. Genetics of COVID-19. J Pediatr (Rio J). 2021 Jul-Aug;97(4):378-386. doi: 10.1016/j.jped.2020.09.002. Epub 2020 Oct 7. PMID: 33058776; PMCID: PMC7539923.
9. Wang, R., Chen, J., Gao, K. *et al.* Analysis of SARS-CoV-2 mutations in the United States suggests presence of four substrains and novel variants. *Commun Biol* **4**, 228 (2021). <https://doi.org/10.1038/s42003-021-01754-6>
10. Mercatelli Daniele, Giorgi Federico M. Geographic and Genomic Distribution of SARS-CoV-2 Mutations *Frontiers in Microbiology* ,volume 11, 2020,Pages 1800 ISSN : 1664-302X , DOI: 10.3389/fmicb.2020.01800 , <https://www.frontiersin.org/article/10.3389/fmicb.2020.01800>
11. KDnuggets , The Ultimate Guide To Different Word Embedding Techniques In NLP,2021 Nov, [The Ultimate Guide To Different Word Embedding Techniques In NLP - KDnuggets](#)
12. Machin Library Mastery, What Are Word Embeddings for Text?, [What Are Word Embeddings for Text? \(machinelearningmastery.com\)](#)
13. François Chollet , keras (2015 ) , GitHub , GitHub repository , commit [5bcac37](#) , <https://github.com/fchollet/keras%7D%7D>
14. Analytics Vidhya , Explained Deep Sequence Modeling with RNN and LSTM ,(31 Dec 2020) ,Medium, [Explained Deep Sequence Modeling with RNN and LSTM | by Shachi Kaul | Analytics Vidhya | Medium](#)