Using the Long Short Term Memory Model for Amino Acid Gap Prediction

Jessica Mitchell[1]

North Carolina AT State University, Greensboro NC 27404, USA

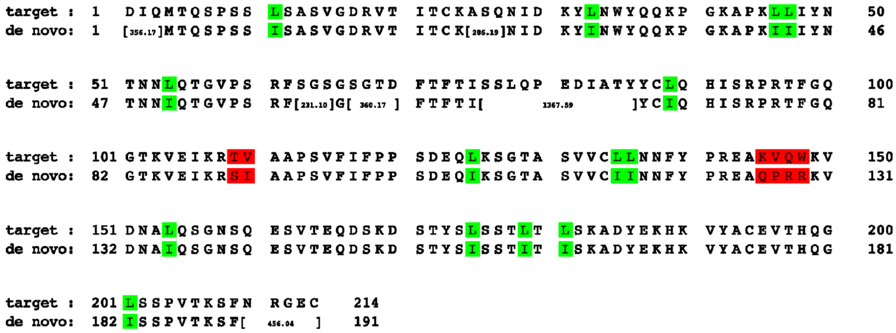
Abstract. The Long Short Term Memory Model (LSTM) was used to predict gaps in de novo scaffold protein sequences that were sampled from paper using amino acid sequences. This was conducted using python, google colab, research and information from many sources online as well.

Keywords: Long Short Term Memory · De Novo Scaffold · .

1 Background

1.1 Amino Acid

Within healthcare fields, scientific laboratories, and other forms of science amino acid sequences are analyzed and used. An amino acid sequence is the arrangement of an amino acids in a protein[1] . Within this project an amino acid sequence was used as an input to train a deep learning model. This model was then able to proceed with predicting gaps within a given amino acid sequence when portions of the sequence were provided. The sequence with the gaps used within this project is the De Novo scaffold.



```
target :  1   DIQMTQSPSS            LSASVGDRVT  ITCKASQNID     KYLNWYQQKP   GKAPKLLIYN   50
de novo:  1   [356.17]MTQSPSS       ISASVGDRVT  ITCK[286.19]NID KYINWYQQKP  GKAPKIIIYN   46

target :  51  TNNLQTGVPS  RFSGSGSGTD          FTFTISSLQP  EDIATYYCLQ  HISRPRTFGQ   100
de novo:  47  TNNIQTGVPS  RF[231.10]G[ 360.17 ] FTFTI[      1367.59     ]YCIQ  HISRPRTFGQ   81

target :  101 GTKVEIKRTV  AAPSVFIFPP  SDEQLKSGTA  SVVCLLNNFY  PREAKVQWKV   150
de novo:  82  GTKVEIKRSI  AAPSVFIFPP  SDEQIKSGTA  SVVCIINNFY  PREAQPRRKV   131

target :  151 DNALQSGNSQ  ESVTEQDSKD  STYSLSSTLT  LSKADYEKHK  VYACEVTHQG   200
de novo:  132 DNAIQSGNSQ  ESVTEQDSKD  STYSISSTIT  ISKADYEKHK  VYACEVTHQG   181

target :  201 LSSPVTKSFN  RGEC        214
de novo:  182 ISSPVTKSF[  456.04 ]    191
```

Fig. 1. This is an image of what was used within this project for input

1.2 Deep Learning

Deep Learning is focused within this paper through the use of the Long Short Term Memory (LSTM) model. This is a variant of the Recurrent Neural Network (RNN) architecture. RNNs are a type of artificial neural network that contains loops, that allow for information input into it to be stored. But there are limitations of the RNN which is why the LSTM is necessary.

2 Long Short Term Memory Model

2.1 LSTM Network Architecture

LSTM stands for Long Short Term Memory. LSTM is a type recurrent neural network that is capable of learning long-term dependencies between time steps of the sequence. It consists of two main components:

1. Sequence Input Layer: This layer inputs sequence data into the network.
2. LSTM Layer: This layer learns the long term dependencies between time steps of sequence data.

A simple architecture of LSTM that is used for classification consist of sequence input, LSTM, fully connected, softmax, and classification as shown above.(see Fig. 2).

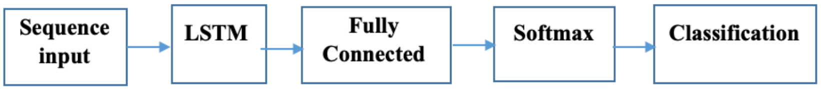


Fig. 2. This figure is of the Long Short Term Memory Architecture and is described below in following paragraphs

The program starts with sequence input layer and then moves to the LSTM layer. The LSTM network ends with a fully connected layer, softmax layer and classification layer.

In this project, a deep learning long short term memory model is trained to be able predict gaps in lettered sequence. The model is provided by target sequences that is used to train it and then it is supplied with de novo with gaps. The model is then tested to see if it can predict the gaps correctly. The layers of LSTM used in this project includes the following:

- Sequence Input
- LSTM
- Fully Connected
- Softmax
- Classification

2.2 Sequence Input

The first layer in LSTM is input layer. Sequence input in this case is the target sequence. The target sequence is loaded into the memory. Since the model only understand numbers, the target sequence is mapped into numbers. The numbers are also mapped backed to characters for feature use. The input and output data are then separated and placed in a format that is suitable for training by the model.

2.3 LSTM Layers

The second layer in LSTM is LSTM layer. The input to every LSTM layer must be three-dimensional. Therefore, the sequence input is first reshaped into 3D before being used in the LSTM layer. The three dimension consist of samples, time steps, and features. The LSTM input layer is specified on the first hidden layer of the network using the "input shape" argument. Within this project 1 layer was used which had

2.4 Fully Connected Layer

The third layer is fully connected layer. It is also the dense layer. In this layer, all the neurons connect to all the other layers. It takes input from the LSTM layer, performs certain operation on it and returns the output.

2.5 Activation Layer

In this layer, an activation function is used. An activation function is responsible for defining how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network. In this project, an activation function called Softmax is applied on the output layer.

2.6 Output Layer

The last layer in LSTM is the output layer. This is the layer in which predictions are made. For example, when De Novo with gaps is inputted into the model, this layer predicts the gaps.

3 Code

3.1 Python

For this semester long project python was the only language used. The code can be understood and broken up into different sections as can be seen within the comments. It began with importing the machine learning libraries necessary to accomplish task. Those that were imported and were the most beneficial was the numpy and keras libraries. Numpy is used for the the support of multidimensional arrays.

Listing 1.1. Python Code snippet of LSTM layer

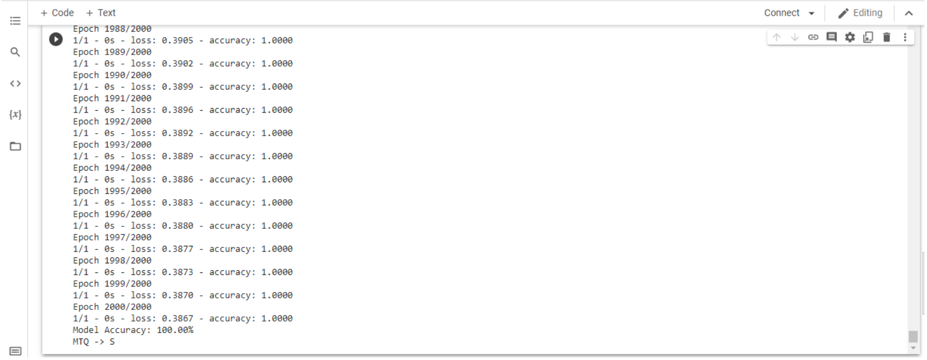
```
# create and fit the model
model = Sequential()
model.add(LSTM(200, input_shape=(X.shape[1], X.shape[2])))
model.add(Dense(y.shape[1], activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
model.fit(X, y, epochs=2000, batch_size=len(dataX), verbose=2, shuffle=False)
```

3.2 Challenges

Some of the challenges faced in this project was creating the function to predict the next occurring letter within the sequence. As well as training the model to have positive accuracy without overfitting occurring from constantly testing the model.

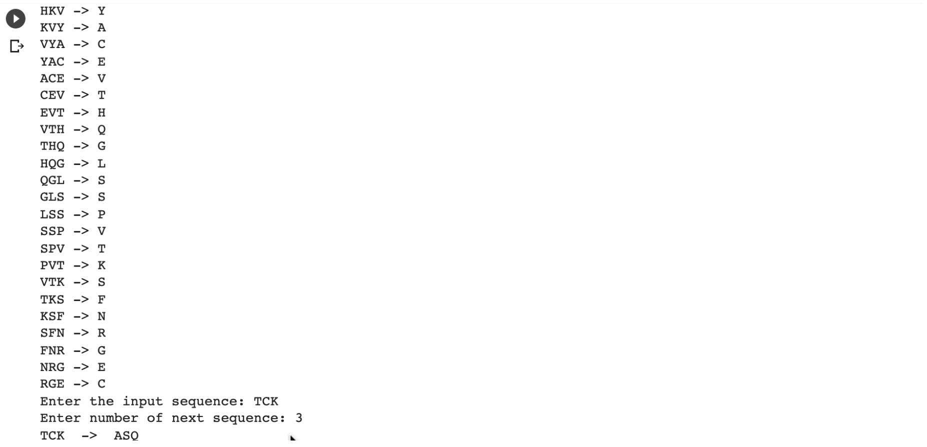
3.3 Results

The model accuracy was able to achieve at one hundred percent accuracy(see 4) while also being able to give the next occurring letter when a gap is present through a function when inputting part of the sequence and the number of output that the user would like to be predicted(see Fig. 4).



```
Epoch 1988/2000
1/1 - 0s - loss: 0.3905 - accuracy: 1.0000
Epoch 1989/2000
1/1 - 0s - loss: 0.3902 - accuracy: 1.0000
Epoch 1990/2000
1/1 - 0s - loss: 0.3899 - accuracy: 1.0000
Epoch 1991/2000
1/1 - 0s - loss: 0.3896 - accuracy: 1.0000
Epoch 1992/2000
1/1 - 0s - loss: 0.3892 - accuracy: 1.0000
Epoch 1993/2000
1/1 - 0s - loss: 0.3889 - accuracy: 1.0000
Epoch 1994/2000
1/1 - 0s - loss: 0.3886 - accuracy: 1.0000
Epoch 1995/2000
1/1 - 0s - loss: 0.3883 - accuracy: 1.0000
Epoch 1996/2000
1/1 - 0s - loss: 0.3880 - accuracy: 1.0000
Epoch 1997/2000
1/1 - 0s - loss: 0.3877 - accuracy: 1.0000
Epoch 1998/2000
1/1 - 0s - loss: 0.3873 - accuracy: 1.0000
Epoch 1999/2000
1/1 - 0s - loss: 0.3870 - accuracy: 1.0000
Epoch 2000/2000
1/1 - 0s - loss: 0.3867 - accuracy: 1.0000
Model Accuracy: 100.00%
MTQ -> S
```

Fig. 3. This figure is the model accuracy of the associated code



```
HKV -> Y
KVY -> A
VYA -> C
YAC -> E
ACE -> V
CEV -> T
EVT -> H
VTH -> Q
THQ -> G
HQG -> L
QGL -> S
GLS -> S
LSS -> P
SSP -> V
SPV -> T
PVT -> K
VTK -> S
TKS -> F
KSF -> N
SFN -> R
FNR -> G
NRG -> E
RGE -> C
Enter the input sequence: TCK
Enter number of next sequence: 3
TCK  ->  ASQ
```

Fig. 4. This figure is the final output when the de novo scaffold is inputted and areas where gaps are present are put, the length of the sequence is able to be adjusted

4 Conclusion

This project was able to demonstrate and produce an accurate prediction of the gaps within a de novo scaffold. For future works I believe that other sequences could be trained on the model to make even more predictions. This could be a great tool to assist future biologist and those who are interested in bioinformatics.

References

1. @miscnational cancer institute, title=NCI Dictionary of Cancer Terms, url=https://www.cancer.gov/publications/dictionaries/cancer-terms/def/amino-acid-sequence, journal=National Cancer Institute