Next Word Prediction Using LSTM

Sabrina Shawon, Abdus Sabur, D. M.Khalid Mahmud and Marshia Muntaka
1801002, 1801034, 1901004 and 1901021

Abstract

Next word prediction which is also called language modelling is one field of natural language processing that can help to predict the next word. It's one of the uses of machine learning. Some researchers have discussed it using different models such as Recurrent Neural Networks and Federated Text Models. Each researcher used their models to make the prediction and so did the researcher here. Researchers here chose to make the model using the Long Short Term Memory(LSTM) model with 200 epochs for the training. For the dataset, the researcher used web scraping. There are 180 Indonesian destinations in the dataset, spread across nine provinces. Tensorfow, Keras, Numpy, and Matplotlib were the libraries that the researchers used. The researcher utilized Tensorfowjs to obtain the model in JSON format. The researcher then used Google Colab to code the tool. With a final result of 8ms/step, 55%loss, and 75% accuracy, it is suffciently good to predict words that will come after.

Architecture Model

The model architecture is shown in the figure below.To plot the model, here we used tf.keras.utils.plotmodel. Then to show the layer name, just need to change the value of show layer names to be True. To download the model and name it, need to use the file then just name the file. Here we used this simple model so it won't take along time when being run.

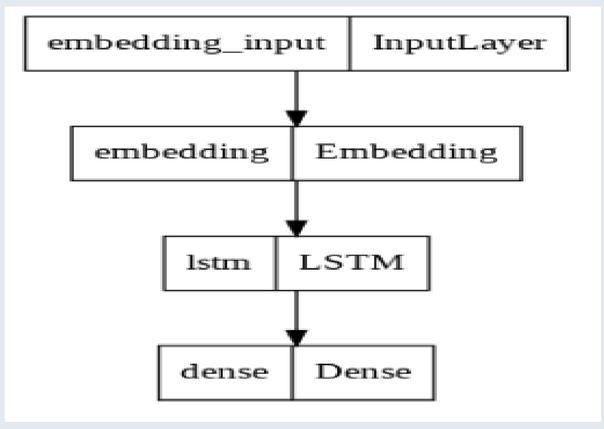

Fig: Architecture Model

Proposed Methodology

The next-word prediction model is based on LSTM (Long Short-Term Memory), a neural network architecture known for its effectiveness in understanding sequential data. We collected the dataset by employing web scraping techniques, which automate the extraction of data from websites, sparing us from the laborious task of manual data collection. This dataset consists of a variety of sentences from different sources, enhancing the model's ability to make accurate predictions.

The essential libraries used in this project are TensorFlow, Keras, NumPy, and Matplotlib. TensorFlow and Keras are pivotal for building and training the LSTM model. NumPy is used for efficient numerical operations, and Matplotlib assists in visualizing the model's performance. To make the model readily accessible, we exported it in JSON format using TensorFlow.js, enabling straightforward integration into various applications.

For coding and development, we opted for the Google Colab Notebook with the Python language, which provides a convenient and collaborative coding environment, eliminating the need for extensive local hardware resources.

Model Summery

To show the model summary, we just need to call model.summary then the result will be shown. Here's the model summary of the model which here it's using a sequential model.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 64, 100)           48500

 lstm (LSTM)                 (None, 150)               150600

 dense (Dense)               (None, 485)               73235

=================================================================
Total params: 272335 (1.04 MB)
Trainable params: 272335 (1.04 MB)
Non-trainable params: 0 (0.00 Byte)
```

Fig: Model Summery

Result

Accuracy and web view of words to get the maximum probability of predicting the word:

```
- accuracy: 0.9869
- accuracy: 0.9869
- accuracy: 0.9871
- accuracy: 0.9861
- accuracy: 0.9838
- accuracy: 0.9483
- accuracy: 0.9793
- accuracy: 0.9856
- accuracy: 0.9871
- accuracy: 0.9876
```

Fig: Accuracy

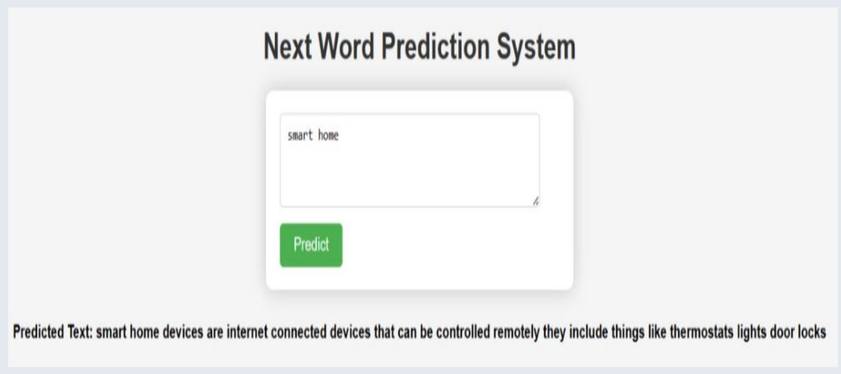

Fig: Web view of word prediction

Conclusion

Next word prediction is one of the NLP fields because it's about mining the text. The researcher here used the LSTM model to make the prediction with 200epochs. The result showed that it maintained get accuracy of 75 percent while the loss was 55 percent. Based on that result, it could be said the accuracy is good enough. It also showed that it's better than two of the two other research which used different models. The model could be used to predict the next word by giving the input of the destination.

References

[1] Jordan,MichaelI.,andTomM.MitchellI."Machinelearning:Trends,perspectives,andprospects."Science349,no.6245(2015):255-260.
[2] Sahoo,AbhayaKumar,ChittaranjanPradhan,andHimansuDas."Per-formanceevaluationofdifferentmachinelearningmethodsanddeep-learningbasedconvolutionalneuralnetworkforhealthdecisionmaking."InNature-inspiredcomputingfordatascience,pp.201-212.Springer,Cham,2020
[3] Prajapati,GendLal,andRekhaSaha."REEDS:RelevanceandenhancedentropybasedDempsterShaferapproachfornextwordpredictionusing alanguagemodel."JournalofComputationalScience35(2019):1-11.
[4] Ambulgekar,Sourabh,SanketMalewadikar,RajuGarande,andBhartiJoshi."NextWordsPredictionUsingRecurrentNeuralNetworks."In ITMWebofConferences,vol.40,p.03034.EDPSciences,2021.
[5]Stremmel,Joel,andArjunSingh."Pretrainingfederatedtextmodelsfornextwordprediction."InFutureofInformationandCommunicationConference,pp.4488.Springer,Cham,2021.