

Statistical Methods in Artificial Intelligence

Lyric Generation using Recurrent Neural Network

*Submitted By: Roll No.*

Praveen Balireddy 2018201052

Aman Joshi 2018201097

Arpit Gupta 2018201048

Shubham Pokhriyal 2018201080

ABSTRACT

This project demonstrates the effectiveness of a Long Short-Term Memory language model in our initial efforts to generate lyrics. The goal of this model is to generate lyrics that are to that of a given song.

Problem Statement

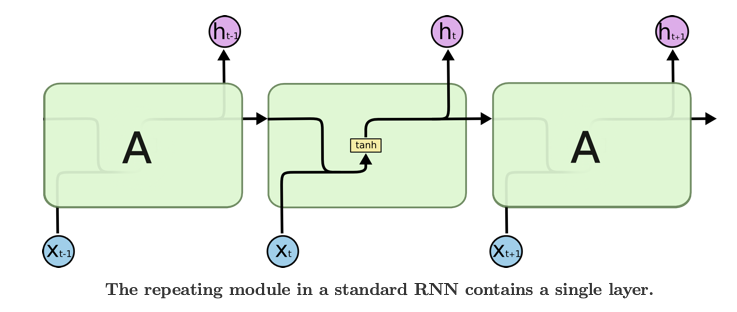
Our goal is to build a model that can generate lyric for both english and hindi songs. For hindi song generated lyrics will be in english translated.

OVERVIEW

RNN

A recurrent neural network (RNN) is a class of artificial neural network where connections between units form a directed graph along a sequence. This allows it to exhibit dynamic temporal behavior for a time sequence. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

Recurrent Neural Network comes into the picture when any model needs context to be able to provide the output based on the input. Sometimes the context is the single most important thing for the model to predict the most appropriate output.

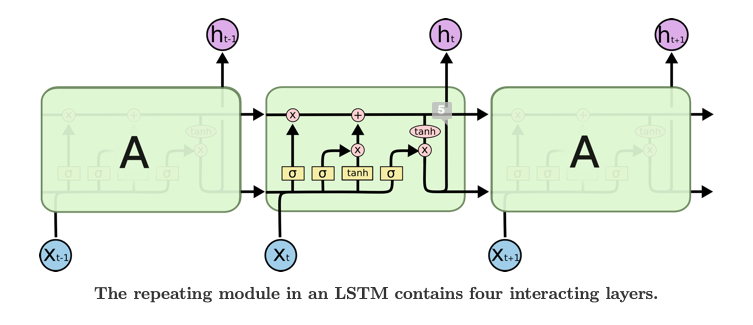


LSTM Networks

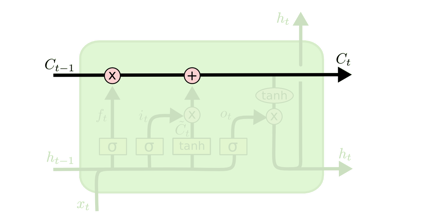
Long Short Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

LSTMs also have this chain like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.

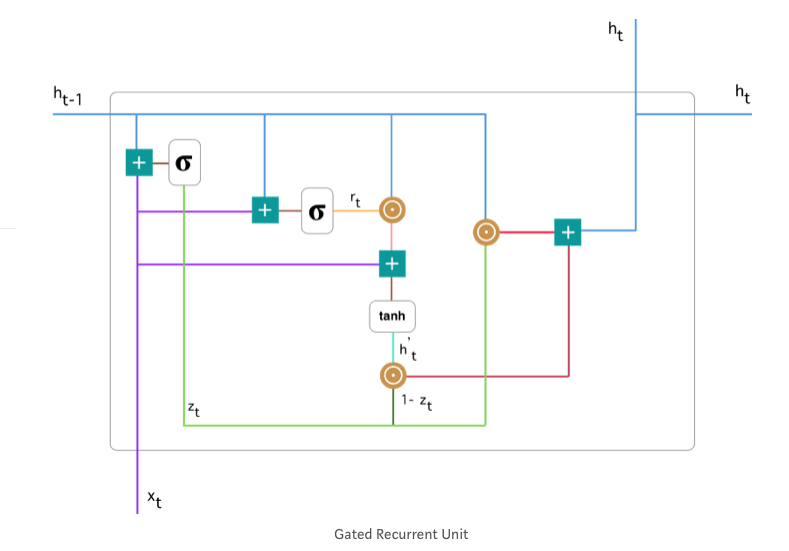


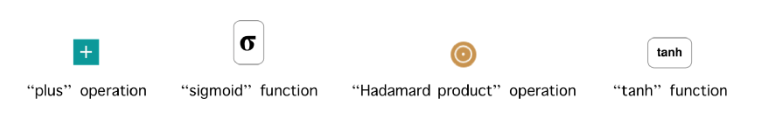
Core Idea behind LSTM



The key to LSTMs is the cell state. The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It’s very easy for information to just flow along it unchanged. The LSTM does have the ability to remove or add information to the cell state, carefully regulated by structures called gates. Gates are a way to optionally let information through. They are composed out of a sigmoid neural net layer and a pointwise multiplication operation. The sigmoid layer outputs numbers between zero and one, describing how much of each component should be let through. A value of zero means “let nothing through,” while a value of one means “let everything through!” An LSTM has three of these gates, to protect and control the cell state.

GRU





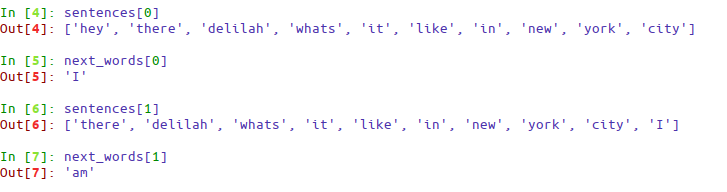
GRU (Gated Recurrent Unit) aims to solve the vanishing gradient problem which comes with a standard recurrent neural network. GRU can also be considered as a variation on the LSTM because both are designed similarly and, in some cases, produce equally excellent results.

To solve the vanishing gradient problem of a standard RNN, GRU uses, so called, update gate and reset gate. Basically, these are two vectors which decide what information should be passed to the output. The special thing about them is that they can be trained to keep information from long ago, without washing it through time or remove information which is irrelevant to the prediction.

APPROACHES

**Word Level Song generator:**

The idea here is to train the model with many sequences of words and the target *next\_word*. As a simplified example, if each sentence is a list of five words, then the target is a list of only one element, indicating which is the following word in the original text.



We don’t actually send the strings, but a vectorized representation of the word inside a dictionary of possible words (described in the later section). The idea is that after many epochs the model will learn “*the style*” of how the corpus is written, trying to adjust the weights of the network to predict the next word given a sequence of the N previous words.

### **The text corpus**

Corpus is created by concatenating all the songs. The data was noisy and unstructured. Some of the problems with the dataset:

* Whenever a line is going to be repeated twice, it is represented as ----2. This information is not very useful for our model. So we’ve removed such repetitions. It helps us in achieving our goal “To learn the style of how the corpus is written”.
* Preprocessing steps like removing extra spaces, smileys, invalid punctuations like ‘@’, ‘#’, e.tc.
* Cleaning words with one-sided parentheses only.
* Lastly changing each ‘\n’ to ‘ \n ’. Because of this step, we can treat newline character as a separate word. We are leaving the decision of starting new line to model itself.

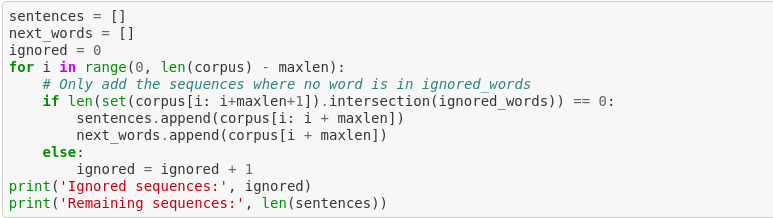
### **Getting word frequencies**

In the character level text generators, we have very fewer numbers of dimensions one for each character. However, in a word level generator, we have a dimension for each one of the different words, which can turn out to be in the tens of thousands (especially in an unstructured and noisy corpus as dirty as this one).

In order to avoid these large numbers of dimensions, we calculate the frequency of each one of the words, so we can use this information to filter out the uncommon words, thus reducing the dimensionality of the data. Hence the memory and time to train the network significantly reduced. We have filtered out the words which have a frequency less than MIN\_WORD\_FREQUENCY. This is done to ‘make the cut’ and select only the frequent words to form the final dictionary of the words. We also created the dictionaries to translate from word to index and from index to word.

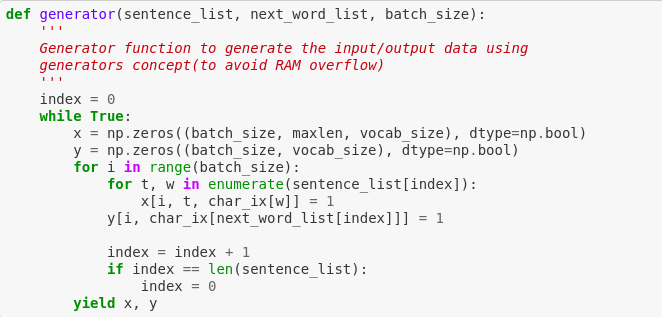
### **Creating and filtering the sequences**

We at this point we have *text\_in\_words* which is an array containing all the corpus word-by-word. We created sequences of size *SEQUENCE\_LEN* (another parameter that can be picked by hand) and store them in *sentences*, and in the same index, store the next word in *next\_words.*



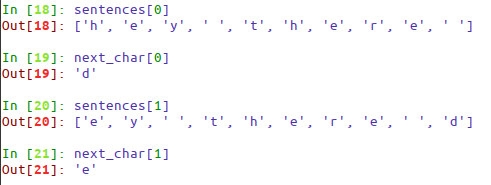
But there is a problem: in *text\_in\_words* we still have many of the words to be ignored. We cannot just go ahead and remove these words because we would be breaking the language and leaving incoherent sentences. That’s why we need to validate each possible sequence+next\_word, it should be ignored it if contains at least one of the ignored words. Without word filtering, we would’ve been ending up using a large number of parameters requiring 100s of GB of memory.

Hence we used data generators. Using data generators, you feed the model with *chunks* of the training set, one for each step, instead of feeding everything at once. The generator function gets a list of sentences and next\_words, and the size of the batch. Then it yields two numpy arrays of *batch\_size* consisting of input sentence and output words*.* We use the *index* variable to keep track of the examples we have already returned. It is reinitialized to 0 whenever we reach the end of the lists. This generator can be used for both training and evaluation (just passing a different *sentence\_list* and *next\_word\_list*).



Character level Song Generator

The idea here is to train the model with many sequences of characters and the target *next\_character*. A simplified example is shown below.



We directly send *Maxlen* characters to the model. The idea is that after many epochs the model will learn “the style” of how the corpus is written, trying to adjust weights of the network such that the model will be able to predict the next character given a sequence of *Maxlen* previouscharacters.

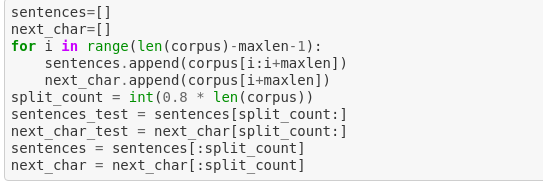
The text Corpus

The corpus creation was more or less the same as that of the word based model. However, there were few differences as follow:

* The space is treated as part of input here. While in word-based model space is used to split the sentence into words, here it is treated the same as that of any other character.
* The conversion of ‘\n’ to ‘ \n ’ is not performed here because we are treating space also as a part of input for model.

Creating Sequence

We created sequences of *Maxlen* (hyperparameter) and store them in *sentences* as in the same index store next character in the *next\_char.*



Then we used generator same as described in word-based approach to feed data to model.

WORK PERFORMED

RESULT

For english songs

For char-based approach -

|  |  |  |
| --- | --- | --- |
| **Architecture** | **Input (seed)** | **Generated lyrics** |
| LSTM\_40\_1\_64 | “ work this treacherous road for all the “ | work this treacherous road for all the side is that some could i don't know on was and but of your down and when i thought the light of i a |
| LSTM\_40\_1\_128 | “the chance on you chorus the first noel “ | the chance on you chorus the first noel the sayin' for a little boy and i can't follow baby i was some to my search that i say i was long an |
| LSTM\_40\_2\_128 | “ the mother mother touched, and dude ain” | the mother mother touched, and dude ain't down to hear a look is so bad with you so dead to help and we know that i ride my read or on our |
| LSTM\_80\_1\_64 | "yes i know so, there's other fish in the sea who would love to swim with me you'" | yes i know so, there's other fish in the sea who would love to swim with me you're so on the sound of your song i'll be don't little great i wish the sun you how the moon you last |
| LSTM\_80\_1\_128 | "early the books, the paintings and the furniture help me the signal's sounding o” | early the books, the paintings and the furniture help me the signal's sounding out a little alone gettin' everybody the sun like the sands then the mountain i've done when you so l |

For word-based approach -

|  |  |  |
| --- | --- | --- |
| **Architecture** | **Input (seed)** | **Generated lyrics** |
| GRU\_5\_1\_64 | "sand  walking with someone” | sand  walking with someone and holding his hand  i should walk with me    it seems i stood and forget all sometimes lovers don't do all they could  sometimes lovers hide in the cover  of innocence and blame  i know that we act just the same  your heart's |
| GRU\_5\_2\_128 | "stop the tears  how" | stop the tears  how i hate to spend our dust in the stars eyes  for the end of the road    so, look out for me, darling  on the power of love    a man don't crawl  that we need to walk around here tonight  gotta go down |
| GRU\_10\_1\_64 | "you made me come alive    [chorus]  i'll" | you made me come alive    [chorus]  i'll never come home  and i think i'll get my love  'cause i love you.  and i know i was never in    i can't take it from you  i know it's not worth trying  so long, so long, so long    so long, |
| GRU\_10\_2\_128 | "greatest times i've had when i was dancing with you" | greatest times i've had when i was dancing with you    i can't feel like your best man can  i've had way too much tonight    please don't tell me  no more i don't care maybe  i don't want to go outside out i couldn't drown it  but as much as tomorrow |
| LSTM\_5\_1\_64 | "'cause i'm in love with" | 'cause i'm in love with a beautiful gal  but she don't care about me  and i closed my eyes, will leave me girl    then i don't want to break these chains  until we want to be free    this is the part when i break free  cause i |
| LSTM\_5\_2\_128 | "i know that can never" | i know that can never end  the love will be a better way  may each day in the year be the one  i never wanna learn    gimme gimme gimme a man after midnight  won't you be alone and you remember  in the hills of mexico  well the |
| LSTM\_10\_1\_64 | "he loves you and life has just begun  it's" | he loves you and life has just begun  it's the time more than any words can say  and i am just a girl  not the kind of woman men would like to meet  just another girl  no one ever looks at in the street  but today i can't believe it's true  when you |
| LSTM\_10\_2\_128 | "it all right  you're all i ever need, my" | it all right  you're all i ever need, my darling  and i would just go no  to give you this heart of mine  you know what i need  when i try to explain it i be  that's as i sit here today  many starts is on  tonight  how i need a |

Hindi (English translated) Songs

For char-based approach -

|  |  |  |
| --- | --- | --- |
| **Architecture** | **Input (seed)** | **Generated lyrics** |
| LSTM\_40\_1\_8 | "hout you chorus i wont live i will die w" | hout you chorus i wont live i will die wanc in as in and beauth a my heart in beant like me the what no le the san and in sore and in and in |
| LSTM\_40\_1\_16 | "eloved didnt come beloved didnt come tha" | eloved didnt come beloved didnt come that beloved didnt come beaut the beade god on shall and shall god about you are and seare you are the |
| LSTM\_40\_1\_128 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |