AUTOMATIC TAG PREDICTION OF POEMS USING BI-DIRECTIONAL LSTM

BY

HARUN-UR-RASHID

ID: 153-15-6647

SABBIR HASAN
ID: 161-15-6919

AND

NAHIDA NAZNIN
ID: 161-15-6902

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Md. Tarek Habib
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised By

Sheikh Abujar
Senior Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2019

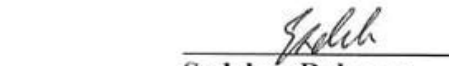
APPROVAL

This Thesis titled **"Automatic Tag Prediction of Poems using Bi-directional LSTM"**, submitted by Harun-Ur-Rashid, ID No: 153-15-6647, Nahida Naznin, ID No: 161-15-6902, and Sabbir Hasan, ID No: 161-15-6919 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 07 December 2019.

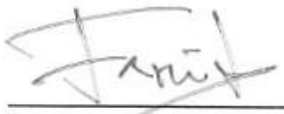
BOARD OF EXAMINERS

Dr. Syed Akhter Hossain — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Zahid Hasan — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Sadekur Rahman — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

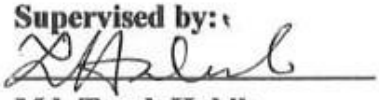


Dr. Dewan Md. Farid — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
United International University

DECLARATION

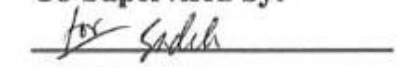
We hereby declare that this project has been done by us under the supervision of **Md. Tarek Habib, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:



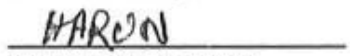
Md. Tarek Habib
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:



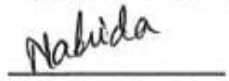
Sheikh Abujar
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:



Harun-Ur-Rashid
ID: -153-15-6647
Department of CSE
Daffodil International University

Sabbir Hasan
ID: -161-15-6919
Department of CSE
Daffodil International University



Nahida Naznin
ID: -161-15-6902
Department of CSE
Daffodil International University

ACKNOWLEDGMENT

First, we express our heartiest thanks and gratefulness to Almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Md. Tarek Habib**, **Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of machine learning to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to Prof. Dr. Syed Akhter Hossain, Head, Department of CSE, for his kind help to finish our project and to other faculty members and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate at Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

The assembly of poems is increasing day by day on internet. A prodigious amount of data sets available on the Internet. However, labeling poems is a very important task. The work in this paper is aimed to find a tagging solution using Bidirectional Long Short-Term Memory Recurrent Neural Network (BLSTM-RNN) that appeared to be very effective for modeling sequential data. To improve the specific functions cautiously optimal for each task, our solution only uses single set of task-independent features. Utilizing task specific information and advanced feature engineering, our proposal delivers almost state-of-the-art performance in predicting tagging tasks.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Poem is a type of composition or artistic writing that attempts to stir a reader's imagination or emotions [1]. There are thousands of poems from different poets, the writer in the online platform. It is very essential to sort them based on their genre and keywords. We need to extract the data from online; however, most of the tools or methods are created for gaining information from the article, and poems that are not very effective at all. For poems, our focus is to ensure a better tagging so that people all over the world can easily find any poem regarding their choice. The noisy nature of user tags makes tag prediction a truly challenging task.

As an introduction to this thesis, this chapter aims to provide some context to explain some of the motivations and challenges regarding the tag prediction of the poem. Section 1.2 describes some of the circumstances behind the appearance of the problem, and why it should be solved. Section 1.3 follows with a description of the problem, and what has been done to solve it. Section 1.4 presents the main objectives our thesis seeking for accomplished. Lastly, Section 1.5 provides an outline of the rest of the paper.

1.2 Motivation

There are thousands of poems from different poets, the writer in the online platform. It is very essential to sort them based on their genre and keywords. We need to extract the data from online; however, most of the tools or methods are created for fetching information from the article, and poems that are not very effective at all. For poems, our focus is to ensure a better tagging so that people all over the world can easily find any poem regarding their choice.

1.3 Context

1.3.1 Tag prediction

Tag prediction is the task of predicting a set of tags from the given content. It is closely related to that of automatic text annotation, especially in the research field, where the line between them seems to be largely obscured. One difference worth pointing out is that automatic text annotation's main goal of attaining categorization opts for tags at a higher semantic level, rather than accounting for the users' individual differences in tag conceptuality. This noisy nature of user tags makes tag prediction a truly challenging task and remains an unsolved problem.

1.3.2 Definition of Tag

Most definitions of the word tag share a similarity; that it is a label designed to provide information about someone or something. In Social Networking and Media, these are called hashtags because of the hash character '#' used in front of them. These tags allow for content categorization and marking which makes it available for later retrieval. However, since the problem of tag prediction is looked at from the angle of using social media and some other archive like the poem, the fact that what is attempted to predict is something user-generated to be more specific. This creates a random variable in the process of predicting tags of content because one has to deal with the individual's interpretation of what an object is or means. For example, a necklace to some might be only "necklace", while to others the label "heirloom" may apply as well. Hung et al. (2008) [2] define tags as the semantic concepts that an object activates in a cognitive sense, which is the definition stuck to throughout the thesis.

1.4 Research Questions

The goal of being able to extract usable knowledge and predict tag accurately from user-generated poems is formalized into the following main research question:

RQ: How to predict poem tags from the given dataset?

To help answer this question and accomplish the research goals, three sub-questions are defined:

RQ1: How to predict tags using BLSTM-RNN?
RQ2: How to use word-embedding models to help predict tags?
RQ3: What source of the text is better for training the word embedding models?

1.4 Expected Outcome

- Being able to predict the tag of poems more accurately using deep learning models.
- Finding out the most used tag of poems in terms of different genres and categories of poems.

1.5 Layout of Report

Our thesis report is organized as follows:

- Chapter One includes introduction to our project, motivation, research questions, and expected outcome.
- Chapter Two includes “Background”, related works, research summary, and challenges.
- Chapter Three includes Research Methodology.
- Chapter Four includes Experimental Results and Discussion.
- Chapter five includes Summary and Conclusion.

CHAPTER 2

BACKGROUND STUDY

2.1 Introduction

To gain some perspective on the research field and its main challenges, a literature survey is performed on tag prediction based on poem classification. First, the works considered the most similar to the work in this thesis are discussed in section 2.2. In Section 2.3, we will give a summary of our related works. In the challenges section, we will discuss how we can increase our accuracy.

2.2 Related Works

There are several works have done relating to our research. Kumar et al. (2014) [3], categorized poems by tag using machine-learning algorithm NB, KNN, SVM accuracies and got 80.00, 87.50, and 93.25% respectively. Ertugrul et al. (2018) [4] state about movie genre classification using BLSTM. Pylyp, A., & Shakhovska (2019) [5] mentioned their paper about the Auto-Tagging system for articles. However, the matter of fact that there is no specific work on predicting tag from poems using Deep learning approaches. Different types of statistical algorithms and machine learning are useful for classifying text documents such as k-Nearest Neighbor, Naïve Bayesian, and Support Vector Machine [6]. T. Joachim (1998) [7] showed about text categorization using Support Vector Machine and got an accuracy of 86.4%. The difficulty of classifying texts such as poems is less. Malaysian poem [8] is ranked using the conventional machine-learning algorithm e.g. support vector machine.

2.3 Research Summary

Table 2.1: Summary of the related work

SL	Author	Methodology	Description	Outcome
1	Kumar et al.	Using machine-learning algorithm NB, KNN, SVM	Classifying Tag of poems	NB 80.00, KNN 87.50, and SVM 93.25%
2	Ertugrul et al.	BLSTM, Logistic Regression	Classifying Genre of movies	Micro f1 score 67.61
3	Pylyp, A., & Shakhovska	LSTM	Auto-Tagging system for articles	Accuracy 98.06%
4	T. Joachim	SVM	Text Categorization	Combined Accuracy of 86.4%
5	Logan et al.	Radial Basic Function (RBF)	To classify theme of poem and differentiate between text and poem.	Discussed lyrics and poem.

2.4 Challenges

The main challenge for our thesis was not only a huge number of data collection but also make sure that the data is in its purest form. We have collected a thousand data of poems and intended to run a deep learning model. This is quite challenging for us to fit our model to such a small number of a dataset.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

In this section, we will discuss the research subject and instrumentation, data collection procedure, data processing, proposed methodology, statistical analysis, and implementation requirements. Firstly, in the research subject and instrumentation, we will discuss our topic. In the data collection procedure, we have discussed how we collected our data. Next, in the data processing part, we have discussed how we pre-processed it for our model. Then in the proposed methodology, we briefly addressed the algorithms and methodology that were used for this classification. Consequently, in the statistical analysis, we highlighted a few statistical methods and flow charts of the project. Finally, the chapter is closed by a clear concept about what we used for the project.

3.2 Research Subject and Instrumentation

Our research subject is to find appropriate tagging solution for poems. There are many poems stored in websites and archives. Sorting those poems by their genre, keywords, and tags could be very helpful. We have collected & preprocessed data from poetry foundation [9] website. We have created scraper tools, which automatically collect data of text documents from the websites.

3.3 Data Collection Procedure

An assembly of 1000 poems from a famous website, poetry foundation [9] taken for the experiment. More than 225 labels of the poem are attainable on the Internet. It is troublesome to consider all labels of poems for this research. Therefore, our data set has the highest 28 labels. On average, we have analyzed Eight tags per poem.

3.4 Data Format and Statistical Analysis

Data Format

Table 3.1: Format of the data set

Poems	Tags
I can see the sunscreen on your face not rubbed in, rivulets wet the under-chin. Let's get this next pitch right, guys, decades left of percolation.	Nature, Animals, Arts, Sciences, Painting, Sculpture, Gender, Sexuality
I find the heavens beautiful, I find the earth so too, the seas and the ground, the furling of water and gas, the bright distant points of our isolation. I take comfort in the swinging pendant traffic lights, the slurry of wet raw flour. I am programmed to this language, and can only voice my rejection of it in the same language. This is the power of diaspora, the difficulty in finding alternative. Let us send messages to the half-existent. To excuse oneself, to claim not knowing the future, is inhuman. I am so worthless that my body serves as brick, conscripted to build up my prison until it is time to lay my own body down for the walls. It is mechanical, snipping into the loop of every lace, separating from every link the cold wrapped bud. At first the skin is thick and bright, then darkly collapses. Nothing keeps its shape, nothing stands itself upright, we keep sliding apart into smaller and smaller components, and it is in the air above us now, we do not mingle with the outcome	Relationships, Friends, Arts, Sciences, Language, Ethnicity

Data Statistics

Table 3.2: Data Statistics

Number of Instance	Class
911	Poems
174	Number of Unique Tags

3.4.1 Data Preprocessing

After data collection, we need to preprocess it again. We have removed the punctuation, brackets, and stopwords so that while we train the model so that we could find maximum accuracy. Finally, data preprocessing was done in two-part; denoising and normalization.

Denoising

Denoising is a process by which we can remove any kind of Html tags and brackets that could have gathered with the dataset. It generally happens when we scrap data from different websites.

The pseudo code for denoising are:

1. Import Regular expression Library
2. Start a function cleaned.split()
3. cleaned=re.sub('[^A-Za-z]+', '', i)
4. appending cleaned by fil.append(cleaned)

Normalization

Data normalization is a process by which data attributes are organized in a data model or dataset. Data normalization increases data consistency and reduces or eliminates data redundancy. Data normalization also helps to object-to-data mapping. For our dataset, we used two functions, one for removing punctuations and other for stop words.

Stop words are those words that are need to be filtered before or after NLP (natural language processing) data. Stop-words are normally known as the most common words. For our poem tag classification, we have created a list of these stop words we have to eliminate. The list is given below.

Table 3.3: Stopwords of Poems

['im',	'an',	'match',	'me',
'interested',	'alcoholic',	'for',	'does',
'in',	'lets',	'the',	'that',
'feminist',	'take',	'always',	'so',
'oratory',	'a',	'sand',	'what',
'we',	'break',	'and',	'no',
'think',	'after',	'always',	'see',
'jess',	'the',	'air',	'the',
'should',	'great',	'i',	'sunscreen',
'say',	'san',	'make',	'on',
'specifically',	'bernardino',	'me',	'your',
'that',	'sculpture',	'find',	'face',
'yellowbreasted',	'party',	'a',	'not',
'engine',	'sparkling',	'pair',	'rubbed',
'sounds',	'toilet',	'of',	'in',
'on',	'pieces',	'leather',	'rivulets',
'the',	'lay',	'pants',	'wet',
'joshua',	'tiled',	'hanging',	'underchin',
'tree',	'into',	'in',	'lets',
'joshua',	'the',	'a',	'get',
'tree',	'pavilion',	'hut',	'this',
'midshimmy',	'silver',	'',	'next',
'i',	'flushers',	'touch',	'pitch',
'think',	'too',	'them',	'right',
'every',	'tv',	'definitely',	'guys',
'bird',	'piles',	'not',	'left',
'is',	'i',	'leather',	'of',
'mad',	'am',	'can',	'percolation']

Data Visualization

We have visualized data by showing frequency of top 20 tags. Some of the tags we have : ['activities', 'age', 'ancestor', 'ancestors', 'animals', 'architecture', 'arts', 'birth', 'birthdays', 'body']

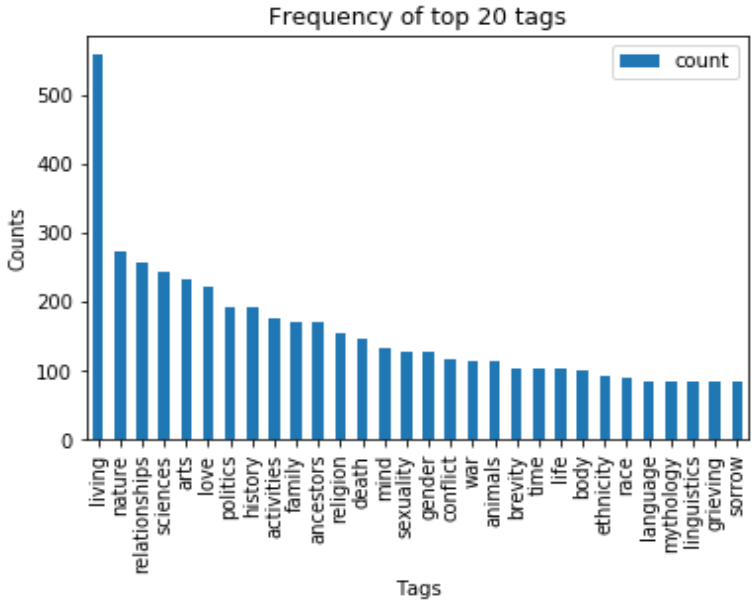


Figure 3.1: Frequency of top 20 tags of poem

Our data consist of a maximum of 28 tags per poem, a minimum of 1 tag per poem: 1, and an average number of tags per poem: 8.068057. The number of unique tags is 174.

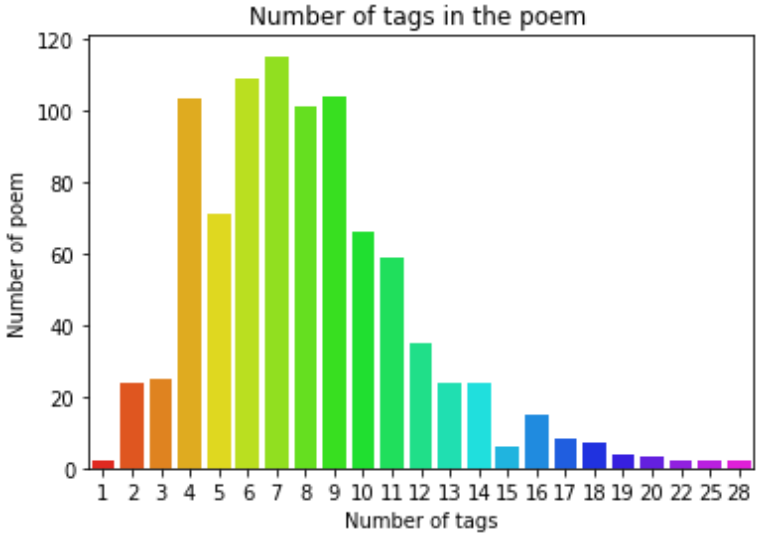


Figure 3.2: Number of tags in the poem

3.5 Proposed Methodology

3.5.1 Methodology

Predicting the tag of poems can be done in a few different ways. For our thesis, we have chosen a deep learning approach to predict tags. We first used IF-TDF for the numeric representation of the poem dataset. After that, we split our dataset into two part, training and testing part. For training and testing, we have used the BLSTM algorithm. Following figure 3.3 explain about our methodology used in our research.

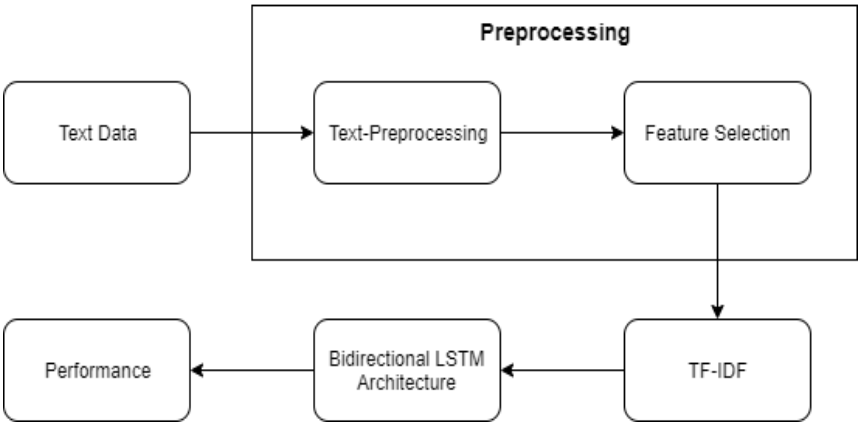


Figure 3.3: Methodology for tag prediction using BLSTM

TF-IDF

TF-IDF is known as a term frequency-inverse document frequency. TF-IDF weight is a statistical measurement normally used to describe how important a word is to a text document in a dataset. The importance increases proportionally to the number of times a word appears in the text document but is canceled out by the frequency of the word in the dataset. Which is It doesn't take into account the fact that the word might also be having a high frequency of occurrence in other text documents. TF-IDF handles this issue by multiplying the term frequency of a word by the inverse document frequency. The term frequency is calculated as follows:

$$\text{Term frequency} = \frac{(\text{number of Occurrence of a Tags})}{\text{Total Tags in a document}} \quad \text{..... (1)}$$

And the Inverse Document Frequency is calculated as follows:

$$\text{IDF (word)} = \frac{\text{Total number of poems}}{\textit{Number of poems containing the tags}} \quad \text{..... (2)}$$

3.5.2 Deep Learning Algorithm

Long Short Term Memory (LSTM):

LSTM [10] is one of the RNN architectures that are widely used. Unlike RNN, It has advantages that it does not have vanishing gradient problem and long-term dependency as LSTM uses multiple gates to carefully regulate the amount of information that will be allowed into each node state. It consists of three gates, an input gate, an output gate and a forget gate (ft). Forget gate (ft) controls the extent of forgetting memory from the last time. The input gate (it) is devised to control the extent of memory to keep flowing.
The figure 3.4 shows the basic cell of an LSTM model.

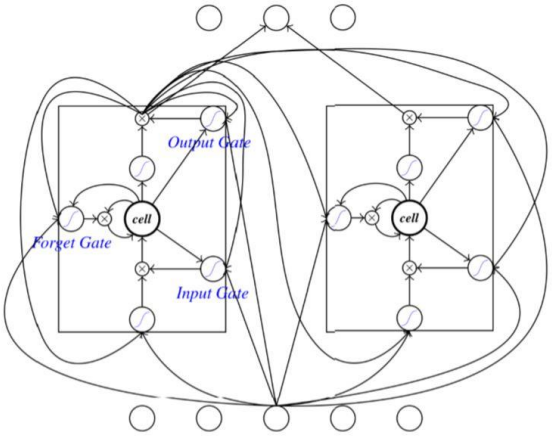


Figure 3.4: An LSTM network. The network consist five input units, a hidden layer composed of two LSTM memory and three output units. Each memory consist of four inputs but only one output. All connections of the left are drawn and other connections are skipped for keeping simplicity.

Bidirectional LSTM and Recurrent Neural Network (BLSTM-RNN)

A bidirectional LSTM (BLSTM) [11] propose around two independent layers to gather information from the past and future histories. It seems common to expect BLSTM to be an very efficient model for tagging tasks in NLP. To further enhancement of performance of our approach without disturbing the universality, we have introduced word embedding, that is a real-valued vector associated with each word. An inner presentation is considered containing syntactic and semantic information and has given a very pleasant feature for different NLP tasks [11]. Word embedding can be gained by training a neural network language model [12] or a recurrent neural network. Following Figure 3.5 explained about bidirectional LSTM network.

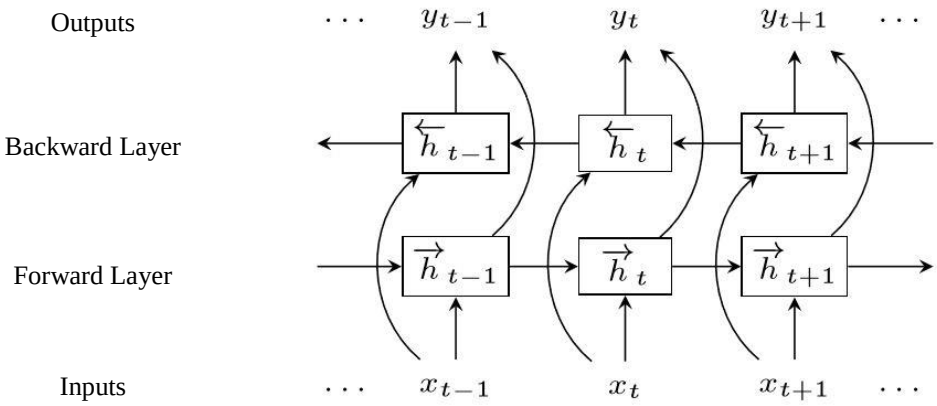


Figure 3.5: Bidirectional LSTM

For a given time step t, the minibatch input is $X_t \in R^{n\times d}$ number of examples: n, number of inputs: d) and the hidden layer activation function is ϕ. In the bidirectional architecture, we assume that the forward and backward hidden states for this time step are H_t (f) $\in R^{n\times h}$ and H_t (b) $\in R^{n\times h}$ respectively. Here h indicates the number of hidden units. We compute the forward hidden state and backward hidden state updates as:

Forward, $H_t(f) = \phi (X_t W(f) + H_{t-1} W(f) + b(f)$ ……(3)

Backward, $H_t(b) = \phi (X_t W(b) + H_{t+1} W(b) + b(b)$ ..……..…(4)

Here, the weight parameters $W(f) \in R^{d \times h}$, $W(f) \in R^{h \times h}$, $W(b) \in R^{d \times h}$, and $W(b) \in R^{h \times h}$, and bias parameters $b(f) \in R^{1 \times h}$ and $b(b) \in R^{1 \times h}$ are all model parameters.

Then we concatenate the forward and backward hidden states $H_t(f)$ and $H_t(b)$ to obtain the hidden state $H_t \in R^{n \times 2h}$ and feed it to the output layer. In deep bidirectional RNNs, the information goes on as input to the next bidirectional layer. Lastly, the output layer computes the output $O_t \in R^{n \times q}$ (number of outputs: q):

$$O_t = H_t W_{hq} + b_q \quad \text{..... (5)}$$

Here, the weight parameter $W_{hq} \in R^{2h \times q}$ and the bias parameter $b_q \in R_{1 \times q}$ are the model parameters of the output-hidden layer. The two directions can have different numbers of hidden units.

3.5.3 Tagging System

The schematic diagram of the BLSTM-RNN based tagging system is illustrated in Figure 3. Given a sentence w_1, w_2… w_n with tags y_1, y2... y_n, BLSTM-RNN is first used to predict the tag probability distribution o (w_i) of each word, then a decoding algorithm is proposed to generate the final predicted tags y_1', y_2'... y'_n. following figure 3.6 explained about BLSTM based tagging system.

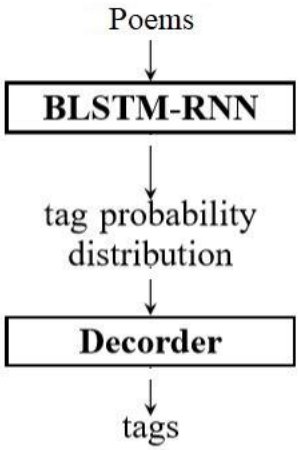


Figure 3.6: BLSTM based tagging system

BLSTM-RNN for tagging

The usage of BLSTM RNN has shown in Figure 3.6. Here w_i is the one hot representation of the current word which is a binary vector with dimension |v| where v is the vocabulary. To lessen |v|, each letter of the input word is transferred to its lowercase.

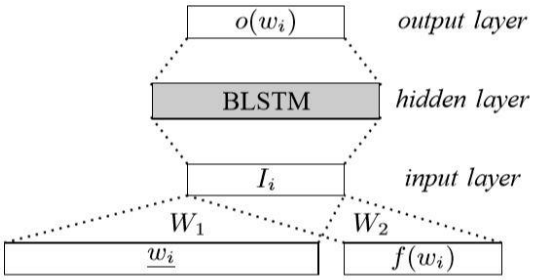


Figure 3.7: Usage of BLSTM for tagging

The upper case info is kept by introducing a three-dimensional binary vector f (w_i) to indicate if w_i is full lowercase, full uppercase or leading with a capital letter. The input vector I_i of the network is computed as follows:

$$I_i = W_1 w_i + W_2 f(w_i) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

Where W1 and W2 are weight matrixes connecting two of the layers. $W_1 w_i$ is known as the word embedding of w_i that is a real-valued vector with a much smaller dimension than w_i. In practice, to reduce the computational cost, W_1 is implemented as a lookup table, $W_1 w_i$ is returned by mentioning to w_i's word embedding stored in this table. It outputs the tag probabilities distribution of word w_i.

Embedding

To make a classifier using natural language, and embedding methods are needed. In this study, Word2Vec is used. However, because of the shortage of samples, a pre-trained Word2Vec model has to be used. The word-embedding dimension we use is 300. When we embedding words in the dataset, we skip the words that are not in the Word2Vec model. The following figure explains our model with embedding.

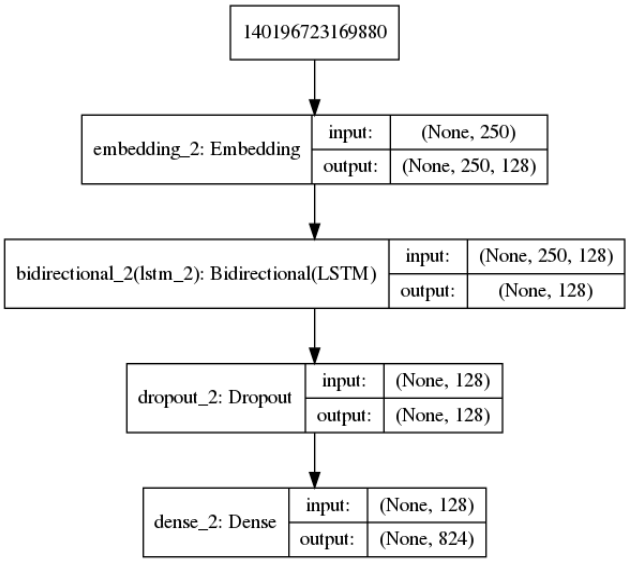


Figure 3.8: BLSTM working model

3.6 Implementation Requirements

After reviewing all the necessary statistical or theoretical concepts and methods, we created a list of Hardware, Software and developing tools we need for predicting Tag of poems. The probable necessary things are:

Hardware/Software Requirements

- Operating System (Windows 7 or above)
- Ram (more than 4 GB)
- Web Browser (preferably chrome)

Developing Tools

- python 3.7
- Anaconda
- Jupyter notebook
- NLTK
- Pandas
- Sklearn

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

In Chapter four, we will discuss the descriptive analysis of our project. We will state our experimental result and finally, we will close the chapter with a summarization of result.

4.2 Experimental Results

BLSTM-RNN:

To measure the effectiveness and accuracy of the algorithms we run the BLSTM model c onsisting of maximum length = 500 and batch size = 256. We divide the dataset into traini ng (60%), and test (40%) subsets.
We got the following result after 10 epochs,
Loss: 0.5955
Accuracy: 0.9397
Value loss: 0.5096
Value accuracy: 0.9976

The following figure explained about our result.

```
Poem: b'stori tell us sisyphus punish push boulder mountain god glare avalanch mood pill size star near quell cascad tum
ult still step graviti amplifi inclin hazard way boulder remind easier hot fudg sunda top long nap shade stori forgot te
ll us though sisyphus thrive learn guid wrist shoulder girdl safe protect later work safeguard everi insect crest consid
call even sun weight time bear strength king'
-----------------------
Actual Tags: Living Time Brevity Mythology Folklore Greek Roman Mythology


Predicted Tags: Living Time Brevity History Politics Mythology Folklore Fairytales Legends
```

Figure 4.1: Result of predicting tags with respect to actual tags

4.3 Descriptive Analysis

Word cloud Analysis

We could find the most used tag for poems by this process. We can easily determine those words that helped us to predict tags by a simple process. Now let us have a look at the most used tags in our dataset.

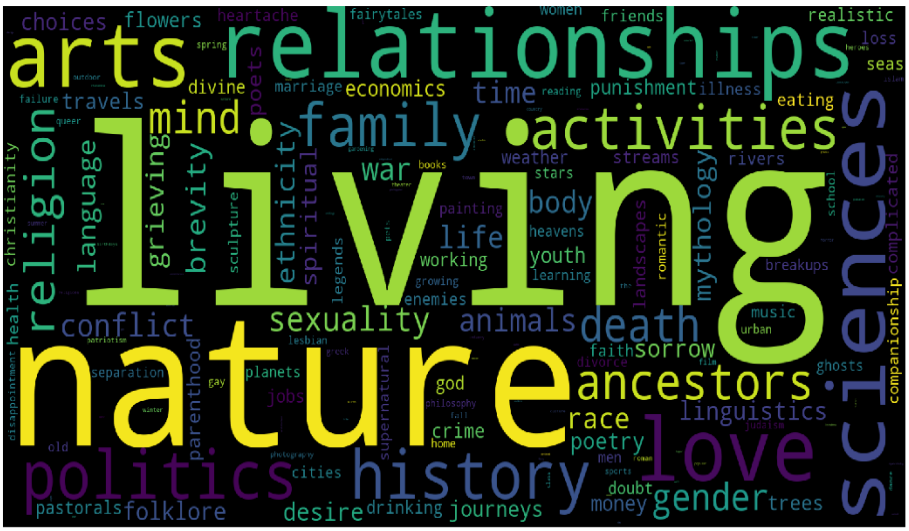


Fig 4.2: image plot of mostly used tag in our dataset

4.4 Summary

After getting the accuracy by using the Bag of Words and Term Frequency-Inverse Document Frequency for vector representation and from our deep learning algorithm, we trained our data and determined the prediction of tags, which is the more similar to the actual tag in our research dataset. We achieved 93.97% accuracy for BLSTM algorithms. However, the data seems overfitting to our model, so we could prepare the dataset more accurately. Also, there is a need to increase the size of the dataset for statistical significance. More preprocessed data will also help to achieve better accuracy for predicting tags.

CHAPTER 5
SUMMARY AND CONCLUSION

5.1 Summary of the Study

Our main target was to build a model that will help to predict the tag of poems. We took the deep learning approach based on BLSTM for predict tag. This system avoids involving task-specific features; instead, it utilizes word embedding learned automatically from the text. In this approach, we vectorize the poem then we split the data for training and testing. Following that, our BLSTM got 93.97% accuracy.

5.2 Conclusions

Our main goal was to build a model that helps predict the Tags of poems. We adopted a Bi-LSTM-based deep learning approach to predict the mark. This system avoids engaging important features; instead, it uses the foundational word automatically learned from the text. In this approach, we draw the poem and then divide the data for training and testing. After that, we achieved Bi-LSTM modulation with an accuracy of 93.97%. We tried to design a model that could be a solution for the label and this approach could be easily applied to different labeling tasks. This study attempted to identify an effective deep learning algorithm. Our results suggest that Bi-LSTM with word combinations is an effective marking solution and deserves further exploration.

5.3 Recommendations

Few recommendations for poetry tagging are:

1. Create a large dataset for high accuracy
2. Try to clean and better word-embedding model for preprocessing data and better accuracy
3. Find and list all the stop-words and this will help to increase the accuracy.

5.4 Implication for Further Study

Few implications that possible in further studies are:

1. Adding more categories of poems and the style of the writing can be added for more information and better result.
2. Using other deep learning algorithms like Gated Recurrent Unit (GRU), Hierarchical Attention Networks can apply on this dataset; can get a better understanding of which model give us the best and higher accuracy.

Reference:

1. Definition of Poem, available at <<https://kids.britannica.com/kids/article/poetry>>, last accessed on 01-06-2019 at 7:00am.
2. Hung, C.-C., Huang, Y.-C., Hsu, J. Y.-j., Wu, D. K.-C., 2008. Tag-based user profiling for social media recommendations. In: Workshop on Intelligent Techniques for Web Personalization & Recommender Systems at AAAI. pp. 49–55.
3. Kumar, Vipin & Minz, Sonajharia. (2014). Poem Classification Using Machine Learning Approach. 10.1007/978-81-322-1602-5_72.
4. Ertugrul, Ali Mert & Karagoz, Pinar. (2018). Movie Genre Classification from Plot Summaries Using Bidirectional LSTM. 10.1109/ICSC.2018.00043.
5. Mukalov, P., Zelinskyi, O., Levkovych, R., Tarnavskyi, P., Pylyp, A., & Shakhovska, N. 2019. Development of System for Auto-Tagging Articles, Based on Neural Network. In COLINS (pp. 106-115)
6. Noraini, J., Masnizah, M.: Shahrul Azman, N.: Poetry classification using support vector machines. J. Comput. sci. 8(9), 1441–1446 (2012)
7. T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features." Presented at the European Conference on Machine Learning, Chemnitz, Germany, 1998.
8. Logan, B., Kositsky, A., Moreno, P.: Semantic analysis of song lyrics. In: the Proceeding of IEEE Int. Conf. on Multimedia and Expo, 2, pp. 827–830 (2004).
9. Learn about Poetryfoundation, available at << https://www.poetryfoundation.org >>, last accessed on 01-06-2019 at 12:00pm.
10. Sepp Hochreiter and Ju¨rgen Schmidhuber. 1997. Long short-term memory. Neural computation, 9(8):1735–1780
11. Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. Signal Processing, IEEE Transactions on, 45(11):2673–2681.
12. Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th International Conference on Machine learning, pages 160–167, Helsinki, Finland.
13. Yoshua Bengio, Holger Schwenk, Jean-Se´bastien Sene´cal, Fre´deric Morin, and Jean-Luc Gauvain. 2006. Neural probabilistic language models. In Innovations in Machine Learning, pages 137–186. Springer.

Plagiarism Report

ORIGINALITY REPORT

28%	23%	12%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	arxiv.org Internet Source	7%
2	"Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012", Springer Science and Business Media LLC, 2014 Publication	3%
3	docplayer.net Internet Source	2%
4	en.d2l.ai Internet Source	2%
5	gluon.ai Internet Source	2%
6	stackabuse.com Internet Source	2%
7	Submitted to University College London Student Paper	1%