Classification of ailments based on medical text

Mai A. Shaaban[1*] **Mariam Kashkash**[1*] **Maryam Alghfeli**[1*] **Adham Ibrahim**[1*]
[1]Mohamed bin Zayed University of Artificial Intelligence
{mai.kassem,mariam.kashkash,maryam.alghfeli,adham.ibrahim}@mbzuai.ac.ae

1 Background

In the recent past, the expansion of the COVID-19 pandemic has reshaped the world radically. Hospitals and medical centers have become fertile ground for the spread of this virus, where patients are in close contact with someone with COVID-19. Social distancing plays a pivotal role in eliminating the spread of this virus (1). Hence, a new term appeared, which is telemedicine. Telemedicine is consulting patients by physicians remotely via vast communication technologies (2). However, the doctors' productivity may decrease due to the intense effort required to balance between in-patients and out-patients (3). Also, most people try to diagnose themselves by expressing their symptoms in the search engine. Then, they start reading from random unauthorized websites on the internet. On the contrary, this is not safe at all and may lead to the misclassification of the ailment.

A wide variety of deep learning paradigms can be applied to remedy this issue. This project aims to speed up the diagnosis process accurately using Natural Language Processing (NLP) models. The dataset that will be used contains more than 6000 text records of variant symptoms along with the type of ailment. The first step in the proposed work is to perform text preprocessing techniques such as lemmatization, stop words removal and generating word embeddings. Then, deep neural networks will take word embeddings as inputs to predict the output (i.e., the ailment). However, deep learning methods suffer from the risk of getting stuck in local optima. This is because the values of weights are initialized randomly. Not only the weights but also their parameters (4). The Bees Algorithm (BA) is one of the swarm intelligence algorithms. It is a population-based algorithm as well as it mimics the behavior of the bees in foraging in nature (5). In the proposed work, the Bees algorithm is used along with deep learning models to enhance the process of hyper-parameter tuning so that the overall performance of classifying unstructured medical text can be improved.

2 Related work

A dynamic deep ensemble model was proposed to classify the text into spam or legitimate by random forests and extremely randomized trees, whereas, features were extracted by convolutional and pooling layers by passing the word embeddings to prevent the manual feature extraction process (6). This model helped in adjusting the complexity of the model; moreover, the resulting accuracy rate was very high (98.38%). This review (7) gave a summary of the NLP methods which were used in the diagnosis of Bipolar disorder, as well as, suggested future prospects to apply the NLP in the medical field. An Arab medical dataset was generated with ten classes (Blood, Bone, Cardiovascular, Ear, Endocrine, Eye, Gastrointestinal, Immune, Liver, and Nephrological), in addition, this dataset was used to validate ABioNER and BERT models in the classification task; the result showed that the first model was better than the second one because it was already trained on Arabic medical corpus (8). This paper (9) proposed a supervised model to extract features from text (gathered from social media) and identified the information of each disaster; in addition, it trained a multi-label classifier to classify the disaster of the given text. The suggested model enhanced the semantic representation of the studied text and gave good results in the task of classification of two multi-label disasters. The model of Convolutional Neural Network (CNN) above of word embedding layer was used to classify Arabic text into different categories: art, music, environment, and finance; this model achieved high accuracy for the task of classification of Arabic sentences, as well as, the overfitting was resolved by using

Preprint. Under review.

dropout layers and l2 weight regularization (10). This article (11) tested and compared Short-Term Memory Networks (LSTM), Gated Recurrent Units (GRU), bidirectional GRU, bidirectional LSTM, LSTM with attention, and bidirectional LSTM with attention to analysis the dataset subjectively, however, LSTM was the best and its accuracy was 97.39%.

3 Methods

3.1 Text Augmentation

In light of the fact that the size of the data shrunk after dropping replicates to 706 data samples, We applied text augmentation in order to enhance the deep learning model performance and reduce the probability of overfitting. Text augmentation is a prevalent technique used to amplify data samples by generating different versions of the given textual data. Eventually, after applying the proposed method, the size of the data increased to 2829 data samples. For text augmentation, we used the nlpaug tool (12).

3.2 Exploratory Data Analysis

Balanced data with no missing values is an essential prerequisite for having a well-generalized model. Data analysis is crucial to identify patterns and extract practical information from the dataset. The objective of this study is to classify ailments; in our case, we have 25 categorical features (ailments). Hence, a balanced dataset should contain a relatively close percentage of occurrence for each class. Figure 1 shows that the given dataset is balanced and targets are equally distributed.

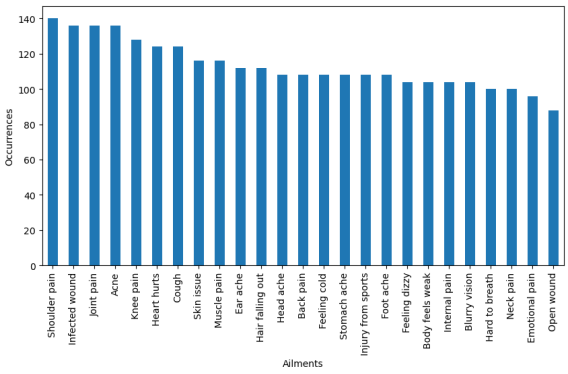

Figure 1: Class distribution of the ailments dataset (13)

3.3 Data Preprocessing

Converting textual data into digits is one of the main pillars of achieving natural language processing in various capacities. The words must be expressed numerically for machine learning or deep learning algorithms. A variety of methods are used to assess text corpus transformation into numerals, and each has its advantages and drawbacks. For instance, TF-IDF, one-hot encoding, and Word Embedding. Term Frequency – Inverse Document Frequency (TF-IDF) technique used in text mining to reflect how important a word is to a document in corpus. One-Hot encoding splits a phrase's words into a group and turns each word into a sequence of numbers regardless of meaning within the context (6).

Word embeddings differ from previously mentioned techniques in that this approach represents each word by a vector of numbers indicating the semantic similarity between the words. It creates a dense vector by transforming each word into a word vector that reflects its relative meaning within the document. Input is illustrated in a matrix $\mathbf{M} \in \mathbb{R}^{n \times d}$ which denotes a collection of phrases. Each phrase m has a sequence of words: w_1,w_2,w_3, ...,w_n; and every word is represented in a word vector of length d [MS].

3.4 Long Short-Term Memory (LSTM)

A Deep Learning approach for modelling sequential data is Recurrent Neural Networks (RNN). For sequential data such as text, RNNs help predict what word or phrase will occur after a particular text, which could be a beneficial asset. This approach produces cutting-edge outcomes on text classification problems.

Long Short-Term Memory Network (LSTM) is a type of RNN where LSTM cell blocks are in place of the standard neural network layers. LSTM models have been shown to be capable of achieving remarkable text classification performance. LSTM cell consists of three different cells, namely the input, the forget and the output gates, which are used to determine which signals can be forwarded to the next node. Figure 2 illustrates the structure of the LSTM cell. The hidden layer is connected to the input by a weight matrix U. W represents the recurrent connection between the previous hidden layer and the current hidden layer. The candidate hidden state $\tilde{C}$ is computed using the current input and the previous hidden state. C denotes the internal memory of the unit. It is a combination of the forget gate, multiplied by the previous memory, and the input gate, multiplied by the newly computed hidden state (14). Equations (1- 6) show the detailed workflow of LSTM cell.

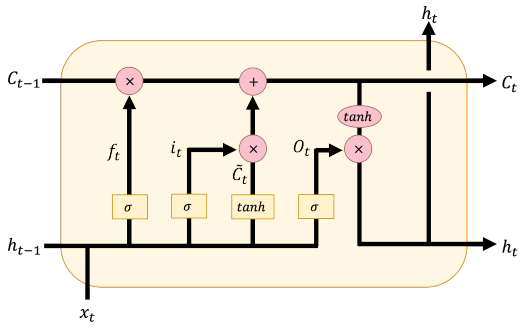

Figure 2: LSTM cell structure

$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right) \tag{1}$$

$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right) \tag{2}$$

$$o_t = \sigma\left(x_t U^o + h_{t-1} W^o\right) \tag{3}$$

$$\tilde{C}_t = tanh\left(x_t U^g + h_{t-1} W^g\right) \tag{4}$$

$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right) \tag{5}$$

$$h_t = tanh(C_t) * o_t \tag{6}$$

4 Experiments

In this section, we explore the ailment classification dataset (13) that contains 2829 unique text samples after performing data augmentation. First, data preprocessing techniques were applied including tokenization, stop words removal, and lemmatization. Then, we split data into training, validation and test sets to apply 10-fold cross validation along with LSTM to predict the patient's ailment. Finally, we conducted ablation studies and evaluated the model.

4.1 Model configuration

Table 1 summarizes the configuration of LSTM hyper-parameters: number of units, number of epochs, batch size, etc.

Table 1: Hyper-parameters setting of LSTM

Parameter	Value
Units	64
Epochs	20
Batch size	10
Dropout	0.2

4.2 Evaluation metrics

The performance of LSTM model was evaluated in Table 2 based on the following well-known metrics for multi-class classification:

- T_P: denotes true positives.
- T_N: denotes true negatives.
- F_P: denotes false positives.
- F_N: denotes false negatives.
- Precision: the proportion of the sum of true positive samples across all classes divided by the sum of true positive samples and false positive samples across all classes.

$$Precision = \frac{T_P}{T_P + F_P} \tag{7}$$

- Recall: the proportion of the sum of true positive samples across all classes divided by the sum of true positive samples and false negative samples across all classes.

$$Recall = \frac{T_P}{T_P + F_N} \tag{8}$$

- F1-score: is a weighted average of precision and recall.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{9}$$

- Accuracy: compares the set of predicted labels to the corresponding set of actual labels.

$$Accuracy = \frac{T_P + T_N}{T_P + F_P + T_N + F_N} \tag{10}$$

Table 2: Classification results of LSTM

Metric	Average weighted value
Precision	0.9837
Recall	0.9816
F1-score	0.9816
Accuracy	98.16%

As illustrated in Figure 3, the training and validation curves for around 20 epochs of training using the Adam optimizer (15) produced good fit model. However, results of training more epochs will be explored in Section 4.3.

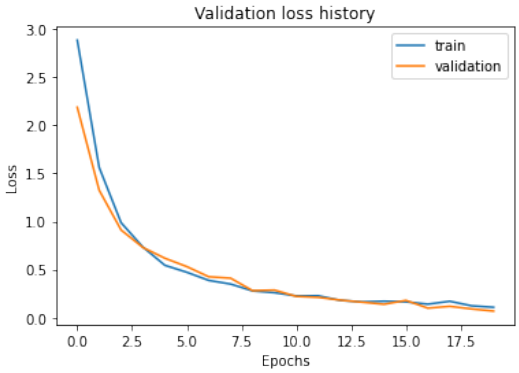

Figure 3: Training and validation curves of the LSTM model

Table 3: Performance of LSTM with different numbers of epochs

Epochs	Average accuracy%
10	96.07
20	98.16
30	98.69
40	98.40

Table 4: Performance of LSTM with different numbers of units

Units	Average accuracy%
32	98.19
64	98.16
128	98.09

4.3 Ablation studies

Table 3 shows the effect of increasing the number of epochs from 10 to 40. Nevertheless, there is no significant improvement in the performance of LSTM after 20 epochs. Furthermore, Table 4 shows the outcomes of applying different numbers of LSTM units starting from 32 to 128. However, the best accuracy achieved was 98.19% by setting the number of units to be 32 along with training 20 epochs.

References

[1] M. Lotfi, M. R. Hamblin, and N. Rezaei, "COVID-19: Transmission, prevention, and potential therapeutic opportunities," sep 2020.

[2] I. Khemapech, W. Sansrimahachai, and M. Toahchoodee, "Telemedicine - meaning, challenges and opportunities," *Siriraj Medical Journal*, vol. 71, no. 3, pp. 246–252, 2019.

[3] H. Wu and Z. Deng, "Do Physicians' Online Activities Impact Outpatient Visits? An Examination of Online Health Communities Completed Research Paper," tech. rep., 2019.

[4] D. Alsaleh and S. Larabi-Marie-Sainte, "Arabic Text Classification Using Convolutional Neural Network and Genetic Algorithms," *IEEE Access*, vol. 9, pp. 91670–91685, 2021.

[5] M. Kashkash, A. Haj Darwish, and A. Joukhdar, " new method to generate initial population of the Bees Algorithm for robot path planning in a static environment," in *Intelligent Production and Manufacturing Optimisation - The Bees Algorithm Approach* (D. Pham and N. Hartono, eds.), ch. 12, Birmingham: Springer 2022., 1st ed., 2022.

[6] M. A. Shaaban, Y. F. Hassan, and S. K. Guirguis, "Deep convolutional forest: a dynamic deep ensemble approach for spam detection in text," *Complex & Intelligent Systems*, pp. 1–13, 4 2022.

[7] D. Harvey, F. Lobban, P. Rayson, A. Warner, and S. Jones, "Natural language processing methods and bipolar disorder: Scoping review," *JMIR Mental Health*, vol. 9, p. e35928, 4 2022.

[8] J. Hammoud, A. Vatian, N. Dobrenko, N. Vedernikov, A. Shalyto, and N. Gusarova, "New arabic medical dataset for diseases classification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13113 LNCS, pp. 196–203, 2021.

[9] S. Xie, C. Hou, H. Yu, Z. Zhang, X. Luo, and N. Zhu, "Multi-label disaster text classification via supervised contrastive learning for social media data," *Computers and Electrical Engineering*, vol. 104, p. 108401, 2022.

[10] D. Sagheer and F. Sukkar, "Arabic sentences classification via deep learning," *International Journal of Computer Applications*, vol. 182, pp. 40–46, 07 2018.

[11] A. Al Hamoud, A. Hoenig, and K. Roy, "Sentence subjectivity analysis of a political and ideological debate dataset using lstm and bilstm with attention and gru models," *Journal of King Saud University - Computer and Information Sciences*, 2022.

[12] E. Ma, "NLP Augmentation." https://github.com/makcedward/nlpaug, 2019.

[13] P. Mooney, "Medical Speech, Transcription, and Intent," 2018. Accessed in 2022.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, p. 1735 – 1780, 1997.

[15] "Adam: A method for stochastic optimization," 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.