An Intelligent Multimodal Medical Diagnosis System based on Patients' Medical Questions and Structured Symptoms for Telemedicine

Hossam Faris[a,b], Maria Habib*[b], Mohammad Faris[b], Haya Elayan[b], Alaa Alomari[b]

[a] *King Abdullah II School for Information Technology, The University of Jordan, 11942, Jordan; hossam.fari@ju.edu.jo;*
[b]*Altibbi (https://altibbi.com), Amman, Jordan; maria.habib@altibbi.com, mohammad.faris@altibbi.com, haya.elayan@altibbi.com, alaa.alomari@altibbi.com*

Abstract

The massive increase in health-related digital data has revolutionized the power of machine learning algorithms to produce more salient information. Digital health data consists of various information, including diagnoses, treatments, and medications. Diagnosis is a fundamental service provided by healthcare agents for improving patient health. However, diagnosis errors result in treating the patient incorrectly or at an improper time causing harm to them. Computer-aided diagnosis systems are intelligent methods that help clinicians in making correct decisions by mitigating the potential of clinical cognitive errors. This paper proposes an intelligent diagnosis decision support system as part of a telemedicine [1] platform for serving the Middle East and North Africa (MENA) region. The proposed system utilizes a huge health-related dataset curated by the Altibbi company, which includes numerous unstructured patient questions written in different dialects of the Arabic language, and structured symptoms identified by specialized doctors. The system encompasses a fusion of machine learning models trained based on two modalities: the symptoms and the medical questions of the patients. Various feature representation techniques (i.e., statistical and word embeddings) and machine learning classifiers, including Logistic Regression (LR),

[1]Telemedicine is defined by the World Health Organization by "healing at a distance, which signifies the use of information and communication technologies to improve patient outcomes by increasing access to care and medical information."

Random Forest (RF), Stochastic Gradient Descent Classifier (SGDClassifier), and variants of the Multilayer Perceptron (MLP) classifier have been used for experiments. The output of the combination of the two modalities has shown promising predictive ability in terms of the classification accuracy, which is 84.9%. The obtained results indicate the potential of the model in predicting the diagnosis of possible patient conditions based on the given symptoms and patients' questions, which consequently can aid doctors in making the right decisions.

Keywords: Altibbi, Multimodal Diagnosis, Machine Learning, Natural Language Processing, Deep Learning, Document Embedding, TF-IDF, Feature Extraction, Digital Health, Telehealth, Telemedicine, Computer-Aided Diagnosis, Arabic Language, MENA

1. Introduction

Digital medical and health informatics have significantly transformed patients' primary care through better healthcare coordination, patient involvement, and improved diagnoses. Differential diagnosis is the process of deciding the etiology of a disease by their symptoms when multiple diseases intersect. It is known to be highly complicated when the case is to detect infrequent diseases. Meanwhile, the early detection of a disease can result in a dramatic impact on a patient's health. The World Health Organization (WHO) reported that approximately 5% per year of adults encounter diagnostic errors in high-income countries [1], while Mahumud et al. [2] proclaimed that nearly 850,000 diagnostic errors are reported annually from developed countries. Managing such clinical diagnosis uncertainty causes a problem, especially for inexperienced physicians or clinicians. Automating the process of diagnosis by computational techniques is a significant objective for online telehealth platforms. The benefits of automated computer-aided diagnosis systems are to make the clinical diagnosis available to all in real-time and save the doctors and patients effort and time. Diagnosis Decision Support Systems (DDSSs) provide clinicians with accurate information to address a condition. DDSSs have a considerable influence on promoting the accuracy of a targeted diagnosis and on improving therapeutic and patient-related decision-making. DDSSs can be classified as knowledge-based, non-knowledge-based, or a hybrid of them [3, 4]. Knowledge-based DDSS integrates a set of rules, which is known as the best practices addressing a condition in the literature. Whereas, the non-knowledge-based systems do not incorporate a predefined set of rules, but use machine learning algorithms to infer such rules from a large number of previously defined cases. On the contrary, the hybrid models integrate information from a predefined knowledge in medical sciences, as well as from learned knowledge of medical experiences.

The problem of misdiagnosis has been argued to be a consequence of cognitive errors made by clinicians, where the statistics show that three out of four diagnostic errors are attributed to a deficit in cognitive biases and clinical reasoning [5]. DDSSs powered by artificial intelligence techniques are known to be the best approaches that include cognitive experiences and medical knowledge to produce better patient health-related decisions [6]. Artificial intelligence is a branch of science that imitates the natural intelligence of humans by machines, where the machines can think and infer knowledge without human intervention by utilizing meta-learning techniques, such as the machine learning methods. Developing such intelligent diagnostic models is critical for mitigating clinical errors, and essential in helping clinicians taking the correct decisions at the right time. However, building efficient diagnostic systems requires the availability of a massive amount of relevant data to train and deploy them. Clinical and digital health platforms are rich resources of clinical raw data presented in various formats, including textual, auditory, or visual. Dealing with textual clinical data requires special methods capable of preprocessing and analyzing such data. Natural language processing techniques can handle and process textual data in order to generate representative features that capture hidden patterns of relationships. The learned features are deployed into learning algorithms to produce meaningful knowledge. Clinical natural language processing analyzes medical or clinical reports that consist of different information including the diagnosis and treatment, which is processed to infer such useful knowledge to aid clinicians in making decisions.

The aim of this article is to automate the process of diagnosis by proposing an intelligent model to help doctors and clinicians in making the correct decision during the diagnosis process. The plan for this model is to assist clinicians in the MENA region, who speak the Arabic language. Natural language processing in the Arabic context is not trivial since Arabic is one of the most complex languages morphologically and phonologically. Also, the Arabic language has different forms, including the dialectical Arabic and modern standard Arabic, where the dialectical Arabic differs among countries, and though in the spelling and writing styles. Furthermore, one of the main challenges when working in the Arabic context is the lack of clinical and medical datasets especially in the case of the multi-dialect. However, in this paper, Altibbi is utilized as a case study, where the data is collected. Altibbi [2] is a well-known digital health platform in the middle east and north Africa, which provides telemedicine services in the region. It has more than 2 million documented consultations, where all clinical notes are stored in its databases. One of Altibbi's primary objectives is to develop a computer-aided DDSS to assist their clinicians and doctors in the diagnosis process, reducing potential errors, and make the process available in real-time, which is also the main inspiration and objective of this paper. Relying on their telemedicine services, more than 10,000 structured symptoms and more than 4,000 diagnoses were curated in order to build such an intelligent diagnostic tool. Typically, the curated data is textual data that requires prepossessing and analysis, which is a fundamental step toward building deployable artificial intelligence models. Figure 1 illustrates the problem and the motivation behind it.

This paper tackles the problem of identifying possible diagnoses by implementing a multimodal classification approach, which is based on machine learning algorithms. This model is expected to provide different advantages; first, providing a reliable diagnosis in the early stages of a disease, which is challenging since the symptoms at the beginning stages are either ambiguous or overlapping [7]. Second, the ability to integrate important information as the medical history or the allergies of a patient, where missing such information makes the diagnosis process more complicated and results in a failure in differentiating the diseases correctly. Third, aids in mapping the clinical notes into their respective diagnosis based on the International Classification of Diseases (ICD), which is known to be cumbersome and error-prone [8].

The proposed classification model is a fusion of multiple modalities. Thus, it combines various information from multiple sources that act as a complementarity either at the data, feature, score, or decision levels. Integrating the data from multiple modalities can improve the efficiency of the learning algorithm. For example, to recognize the emotions of a person; a machine learning model can perform better when integrating data from facial expressions, speech, behavior, and the physiological or brain signals [9]. The

[2]https://www.altibbi.com/

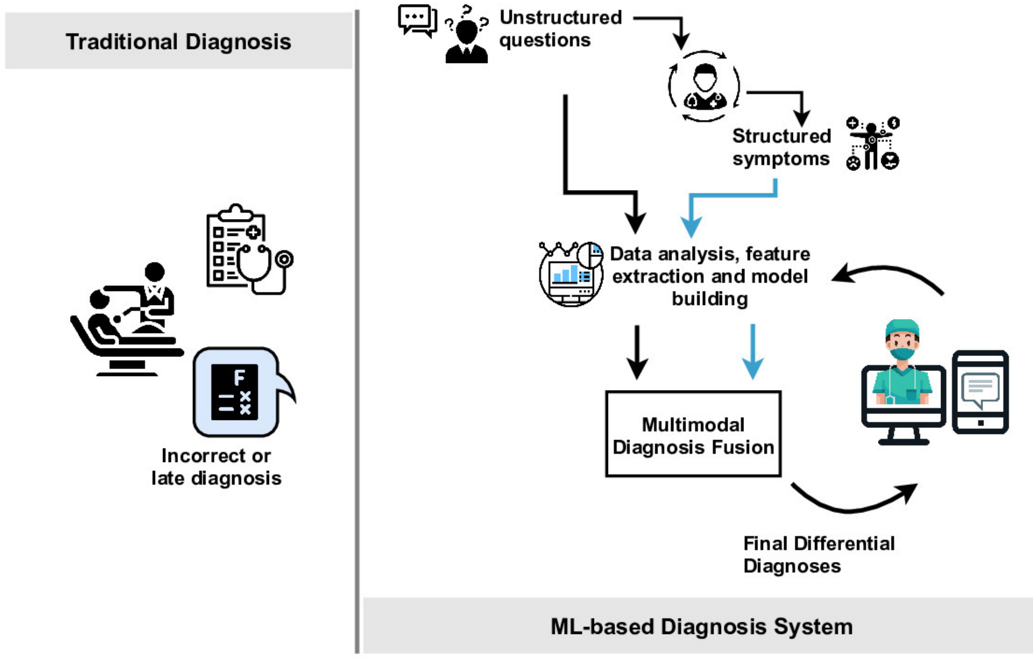

Figure 1: A description of the traditional and machine learning-based differential diagnosis system. On the left-side, the traditional process of diagnosis, where it is susceptible to behavioural or clinical errors or even late decisions. While on the right, the clinician decision is supported by a decision from a machine learning system that might be a multimodal system.

proposed multimodal-based machine learning system depends on two modalities: patients questions, and symptoms identified by General Practitioners (GPs). In this system, two independent machine learning models are developed for each modality then the results of the models are combined for the final predictions. The patients questions are handled by text vectorization techniques that represent the textual words by numerical values. These techniques include the Term Frequency-Inverse Document Frequency “TF-IDF” and hashing vectorizer, which are mainly syntactical features. As well as, the embedding models (e.g., Doc2vec embedding), which extracts the semantics of documents. Whereas, the data of the symptoms is structured data represented by ICD-10 codes that are marked by the GPs for each medical consultation. Mapping the consultations into their correct diagnosis is formulated as a multi-class classification; where the One-Versus-Rest (OVR) approach is utilized. OVR is a heuristic algorithm that makes binary-based machine learning algorithms capable of handling multi-class classification problems. Different machine learning classifiers have been used for experiments and compared independently based on each modality. The used classifiers are the LR, RF, SGDClassifier, and MLP classifier, which are discussed later in the paper. The final outputs of the two models are combined using different schemes; the ranking, summation, and multiplication. The proposed model is evaluated in terms of the accuracy, the inference and loading times, and the size of the classification model. The classification results of the proposed diagnosis model showed promising results that obtained an accuracy of 84.9%.

The main contributions of the proposed approach are:

- Developing a diagnosis decision support system that is based on a fusion of two modalities: structured clinical information and unstructured free-text consultations.
- Developing a system that can serve the context of the multi-dialect Arabic, which is very complex and challenging. Subsequently, deploying the proposed system into the digital health platform (Altibbi); in order to aid Altibbi's doctors in their diagnosis process efficiently and having correct decisions.

The rest of the paper is organized in sections as follows. Section 2: Recent related works in differential diagnostic systems based on machine and deep learning. Section 3: The methodology is presented, including the data collection, as well as the preprocessing, the features extraction, the architecture of the proposed QSDM, and the evaluation criteria of the proposed approach. Whereas, Section 4: The experimental settings, the conducted experiments, and a discussion of results were provided. Finally, Section 5: The findings and suggestions for additional future works.

2. Related works

Developing computational-based intelligent systems to aid in clinical decision-making is of great advantage, as they can avoid potential errors and produce more reliable results. However, there are no such studies on diagnosis prediction particularly (differential diagnosis) due to the lack of needed datasets, especially in non-English contexts. It is worth noting that there are several research studies that proposed computational-based differential diagnostic tools (i.e., visualDX [10], Uvemaster [11], INTEGRA [12], and MED-TMA [13]). However, they were not concerned with the application of natural language processing. The attention of this article is for natural language processing in the Arabic context, therefore, this section reviews recent studies related to single or multiple diagnosis prediction in Arabic and other languages.

2.1. Diagnosis of a single disease

Different studies have applied artificial intelligence techniques for the diagnosis of a specific disease, for example, in [14], a natural language processing approach is used for screening pregnant women for any suicidal behavior. The authors used an online platform for accomplishing the analysis. However, the results were not much satisfactory, but the authors recommended the use of artificial intelligence to aid in the prognosis of suicide. In [15], the authors proposed a machine learning approach for predicting the utilization of radiology resources for the surveillance of hepatocellular carcinoma based on features extracted from radiology reports. Several feature representations and machine learning classifiers experimented. Where the TF-IDF and SVM achieved the highest accuracy (92%). Moreover, Xue et al. [16] constructed a decision tree-based model for the diagnosis of heart disease using EHRs and medical knowledge. The authors utilized pre-trained clinical word embeddings for training the decision tree algorithm, which obtained good performance results (accuracy 89%).

Liu et al. [17] proposed an approach based on natural language processing and machine learning for the identification of liver cancer from textual radiology reports in the context of the Chinese language. The authors constructed a lexicon and utilized the extracted features into different machine learning algorithms (i.e, SVM, LR, and RF). Markedly, the proposed model achieved an f1-score of 90%. Searle et al. [18] proposed a machine learning-based model for the diagnosis of Alzheimer's disease based on features extracted from transcripts of spontaneous speech. The authors used a frequency-based (TF-IDF), and a distributed word representation (DistilBert) with SVM and LR. The (TF-IDF & SVM) as well as (DistilBert & LR) achieved very similar performance, but the (DistilBert & LR) obtained the best results (f1-score=88%). Moreover, Tong et al. [19] proposed an intelligent system for differentiating between the diagnosis of Ulcerative Colitis, Crohn's disease, and Intestinal Tuberculosis in the context of the Chinese language. The authors developed the model based on textual descriptive data of images of colonoscopy, where the extracted features were the TF-IDF and a trainable glove. Generally, CNN had a better performance when compared with RF. Küpper et al. [20] created a machine learning model for the detection of autism spectrum disorders based on the SVM algorithm, and data collected from 673 adolescents. Even the model achieved good results, but the model was not generalizing well. Also, Elaziz et al. [21] created a machine learning diagnostic tool for the diagnosis of Coronavirus disease (COVID-19) using chest x-ray images. Two evolutionary algorithms were utilized for feature selection of attributes extracted from the images, which then fed into KNN classifier. The sizes of the used datasets are approximately 1800 and 1500, which even their small size, they obtained an accuracy of 96% and 98%. Fathi et al. [22] proposed an intelligent approach based on a neuro-fuzzy method for the diagnosis of leukemia, including acute lymphoblastic leukemia and myeloid leukemia in children. However, the major concern was the lack of data which degrades the generalization power of the proposed model. Moreover, Chandra and Verma [23] designed a machine learning approach for the detection of Pneumonia using segmented lung chest X-ray images. The MLP and LR algorithms achieved the highest accuracy scores of nearly over 95%. However, the authors did not consider the scalability and model generalization problems. Yet, Aydin et al. [24] designed a machine learning methodology for the diagnosis of appendicitis in children. They used the decision tree algorithm on 7,244 patients, which achieved 94.69% of accuracy.

2.2. Diagnosis of multiple diseases

In the last few years, several research papers have studied the application of natural language processing and machine learning for the prediction of diagnoses based on the Electronic Health Records (EHRs), as well as the medical and clinical notes. For instance, considering the studies that concerned with the diagnosis of a different number of diseases, Jacobson and Dalianis [25] proposed a deep learning-based approach for the prediction of healthcare infections in the Swedish context. They applied different stacked autoencoders and Restricted Boltzmann Machines (RBM) with different feature representations, i.e., Word2Vec and TF-IDF. The best performance in terms of f1-score was 83% and was obtained by the (TF-IDF & RBM). In [26], the authors automated the classification of textual medical notes into the top 50 frequent diagnoses based on the ICD-9. They applied word and character-level feature representations into LSTM with an attention mechanism. The model did not perform very well, however, the authors provided a discussion of potential limitations. Moreover, Guo et al. [27] constructed an approach for the detection of diseases based on textual symptoms extracted from Electronic Medical Records (EMRs). The extracted features are represented using TF-IDF and fed into a Bidirectional LSTM (BiLSTM). The model achieved an Area Under the Curve (AUC) of 83% when applied to the Medical Information Mart for Intensive Care (MIMIC-III) database. In another paper in [28], in the context of the French language, a deep learning-based method was implemented for the detection of health-related infections based on clinical narratives. A Convolutional Neural Network (CNN) was compared with other machine learning algorithms (e.g., Support Vector Machine (SVM) and Naïve Bayes (NB)) at different word vectorizations (i.e., Word2Vec, Bag-of-Word (BOW), TF-IDF, and Glove). The CNN outperformed machine learning algorithms by obtaining 97% of the f1-score. Also, Atutxa et al. [8] proposed a deep learning-based model for classifying diagnostic reports into their respective ICD-10 codes. The study was implemented for different contexts, including the Italian, French, and Hungarian. Different models were employed (i.e., CNN, Recurrent Neural Networks (RNN), and transformers), where the features were represented using the Word2Vec embeddings. The study obtained very good results in terms of f1-score (Italian (95%), French (83%), Hungarian (96%)).

Furthermore, Nuthakki et al. [29] designed a neural network-based model for the identification of diagnoses from clinical notes using the MIMIC-III database. They classified the data into the top 10 and top 50 frequent classes of the ICD-9 standard, using pre-trained feature representations from the Wikitext103 dataset, and the LSTM classifier. The classification based on the top 10 classes obtained higher accuracy (80%) than the classification using the top 50. Similarly, in [30], the authors performed an automatic ICD-10 mapping of clinical documents. The BOW and TF-IDF were used and integrated into the SVM algorithm, while the Word2Vec was adopted with LSTM and CNN. The results demonstrated better performance for the deep learning classifier. Additionally, Kalra et al. [31] implemented an automatic classification approach for categorizing pathology reports into different diagnoses. The authors used TF-IDF, where the extracted features were fed into linear SVM, XGBoost, and LR. The findings revealed that the XGBoost classifier performed the best in terms of f1-score (92%). In another paper, Obeid et al. [32] implemented an automated detection method of the mental status using data reported from an emergency department provider. Different models were compared, including machine learning (e.g., SVM, NB, RF) and deep learning (e.g., CNN), as well as various features representations (e.g., TF-IDF, pre-trained Word2Vec, and non-trainable Word2Vec at different dimensions). The deep learning model achieved the best performance, where the accuracy was 94.5%. Moreover, Morillo et al. [33] developed a web-based framework based on machine learning for the diagnosis of mental disorders. The tool receives a set of symptoms and maps it into a suitable disorder based on ICD-10 codes. The authors trained the K-Nearest Neighbor (KNN) classifier using the TF-IDF feature vectorizer. However, the training dataset was relatively small.

Also, Castellazzi et al. [34] proposed a machine learning model for the diagnosis of Alzheimer's disease and vascular dementia, where the artificial neural network, SVM, and adaptive neuro-fuzzy inference system were used. The adaptive neuro-fuzzy inference system has achieved the highest accuracy of 84%. Furthermore, Poletti et al. [35] developed a machine learning model for the diagnosis and prediction of mood disorders of major depressive disorder, and bipolar disorder. The proposed model was based on hierarchical logistic regression. Even the used dataset was relatively small, but the model could achieve a score of the area under the curve of 97%. In addition, Fernandes et al. [36] trained a machine learning model for the detection of schizophrenia and bipolar disorder. The implemented model integrates multi-domain data of immune and inflammatory biomarkers of 416 conditions. The model achieved a sensitivity and specificity of 71% and 73%, respectively. Liu et al. [37] proposed a deep learning system (deep CNN) for differential diagnosis of skin diseases based on 16,114 cases. It showed the ability to recognize 26 skin conditions, yet, predict other 419 conditions. The model achieved 66% of top-one accuracy, while the accuracy of three certified dermatologists was 63%. Also, Oktay and Kocer [38] created a Convolutional Long Short-Term Memory (LSTM) for performing a differential diagnosis of Parkinson tremor and essential tremor. Combining the postural and resting positions achieved an accuracy of 90% when tested on 40 subjects. Born et al. [39] developed a deep learning approach for the differential diagnosis of COVID-19 based on ultrasound images. The aim of the model was to classify the images into COVID-19, Pneumonia, and healthy cases, which achieved an accuracy higher than 90%. Table 1 presents a summary of related papers.

Overall, the previous studies demonstrated potential efforts devoted to implementing differential diagnosis systems to promote clinicians' decision-making. Whilst they also disclosed the lack of such systems in the context of the Arabic language. This implies the need for additional research studies

Table 1: Summary of related works.

Reference	Language	Objectives	Techniques applied	Performance evaluation
[14]	English	Screening pregnant women to predict suicidal behaviors	The clinical Text Analysis and Knowledge Extraction System	486 pregnant women were diagnosed positive for suicidal behavior, among whom 146 had confirmed suicidal behavior.
[15]	English	The prediction of hepatocellular carcinoma	TF-IDF, SVM	Accuracy = 92%
[16]	English	The diagnosis of heart disease	DT algorithm	Accuracy = 89%
[17]	Chinese	The identification of liver cancer from textual radiology reports	SVM, LR, and RF	F1-score = 90%
[18]	English	The diagnosis of Alzheimer's disease	TF-IDF SVM, DistilBert LR	F1-score = 88%
[19]	Chinese	The diagnosis of Ulcerative Colitis and Crohn's disease	TF-IDF, Glove, CNN, RF	sensitivities = 99%, specificities = 97%
[20]	English	The detection of autism spectrum disorders	SVM algorithm	Adolescents ¡= 21 the AUC = 90% and Adolescents 21 the AUC = 84%
[21]	English	The diagnosis of Coronavirus disease (COVID-19)	Fractional Multichannel Exponent Moments (FrMEMs) and KNN	accuracy of 96% and 98% for two different datasets.
[22]	English	The diagnosis of leukemia	Neuro-fuzzy method (ANFIS), (GMDH) and the principal component analysis (PCA)	RMSE = 0.0865, MSE = 0.007
[23]	English	The detection of Pneumonia	MLP, LR	Accuracy of 95.63%
[24]	English	The diagnosis of appendicitis in children	DT algorithm	Accuracy of 94.69%
[25]	Swedish	The prediction of healthcare infections	Stacked autoencoders and RBM	F1-score = 83%
[26]	English	Automating the classification of textual medical notes into ICD-9	Word and character-level embeddings and LSTM	F1-score = 53%, AUC = 90%
[27]	English	The detection of diseases based on textual symptoms from EMR	TF-IDF and BiLSTM	AUC = 83%
[28]	English	The detection of health-related infections	CNN, SVM, NB, TF-IDF, BOW, Word2Vec, Glove	F1-score = 97%
[8]	Italian, French, and Hungarian	Classifying diagnostic reports into ICD-10	CNN, RNN and Transformers	F1-score = Italian (95%), French (83%), Hungarian (96%)
[29]	English	Classifying clinical notes into ICD-9	LSTM	Accuracy = 80%
[30]	English	Automatic ICD-10 mapping of clinical documents	BOW + TF-IDF and SVM, Word2Vec + CNN and LSTM	Accuracy = 72.02%
[31]	English	Automatic categorization of pathology reports into different diagnoses	TF-IDF, SVM, XGBoost, LR	F1-score = 92%
[32]	English	Automated detection method of the mental status	SVM, NB, RF, CNN, Word2Vec, TF-IDF	Accuracy = 94.5%
[33]	English	The diagnosis of mental disorders	KNN, TF-IDF	Accuracy = 95.7%
[34]	English	The diagnosis of Alzheimer's disease and vascular dementia	SVM, ANN, ANFIS	Accuracy = 84%
[35]	English	The prediction of mood disorders of major depressive disorder, and bipolar disorder	Hierarchical LR	AUC = 97%
[36]	English	The detection of schizophrenia and bipolar disorder	PCA, Traditional inferential statistics	sensitivity = 71%, specificity = 73%
[37]	English	The differential diagnosis of skin diseases	Deep CNN (Inception-v4)	Top-one accuracy = 66%
[38]	English	The differential diagnosis of Parkinson tremor and essential tremor	Deep convolutional LSTM	Accuracy = 90%
[39]	English	The differential diagnosis of COVID-19	VGG, VGG-CAM, NASNetMobile	Accuracy = 90%

to advance clinical diagnosis decision support systems in the MENA region.

3. Methodology

This section presents the stages of the conducted methodology, which consists of the data collection and preprocessing, features extraction in the case of the questions, the development of the classification model, and the evaluation of the model. Figure 2 shows an overview of the methodology.

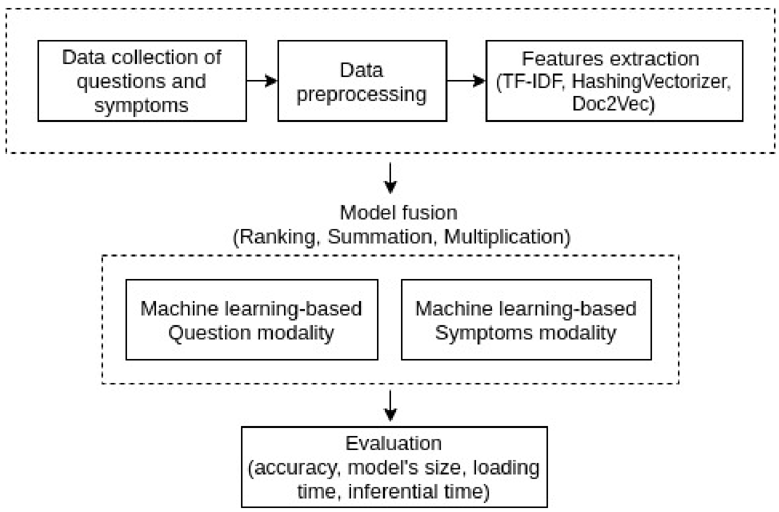

Figure 2: An overview of the proposed methodology.

3.1. Data collection and preprocessing

The total collected data from Altibbi is 263,867 questions (consultations) that are accompanied by symptoms and diagnoses. The total number of symptoms is 7,324, while the diagnoses are 7,410. Each consultation is accompanied by multiple symptoms and multiple diagnoses even that some of them infrequently occur. Primarily, the diagnoses that are repeated less than 20 times over the consultations were removed. Subsequently, the resultant consultations of no diagnosis were removed. Hence, the final number of questions is 246,814, and the number of diagnoses is 1206. Figure 3 shows the number of consultations in relation to the number of diagnoses. It is clear that most of the consultations are of one diagnosis. Meanwhile, several preprocessing steps are utilized to clean and prepare the data for the prediction model.

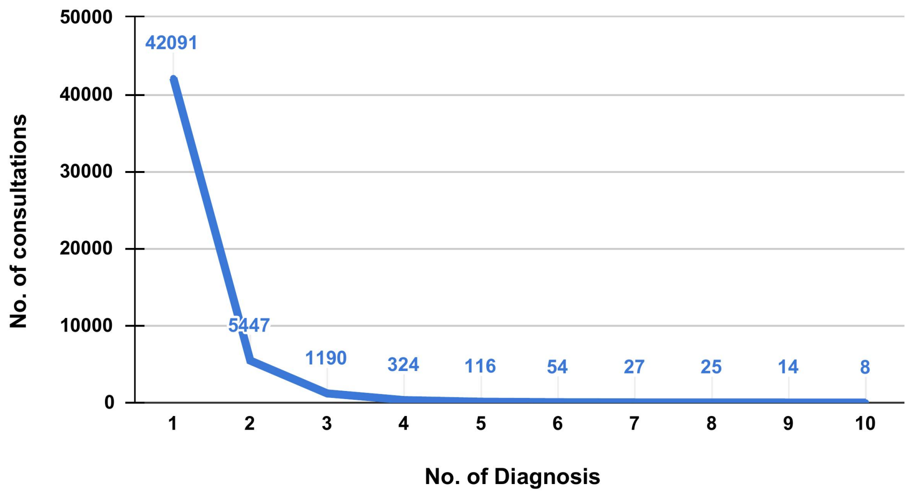

Figure 3: The relationship between the number of consultations and the number of diagnosis.

In the case of the symptom data, each symptom is a binary feature that reflects if it exists in the respective question or not. Similarly are the diagnoses, each diagnosis is a class label of a binary value, where 1 means exists, and 0 does not exist. The final records of data of the symptoms are multi-labeled of a various number of diagnoses. In the case of the questions, the questions were preprocessed by various natural language processing, including the elimination of non-Arabic phrases, numbers, special symbols, diacritics, hyperlinks, punctuation, and the removal of Arabic stop-words and negation words. In addition to the normalization of some Arabic characters. All questions were stemmed by using the light ISRI Arabic stemmer from the Natural Language Toolkit (NLTK) [40], and tokenized by the NLTK tokenizer.

3.2. Feature extraction

Primarily, extracting features from the textual data is done by vectorization. Vectorization is the process of transforming textual documents into numerical feature vectors. In the literature, several approaches have been proposed, such as TF-IDF, hashing vectorizer, and the word embeddings, as described in the subsequent subsections.

3.2.1. TF-IDF vectorizer

The TF-IDF is a textual vectorization technique that utilizes a weighting term to better represent the infrequent words in a corpus and decreases the influence of the frequent non-informative words. Since the existence of irrelevant features mislead the learning process and deteriorates the performance. The TF-IDF is defined by the cross-product of the Term Frequency (TF) and the Inverse Document Frequency (IDF) (TF-IDF = TF × IDF). TF is the proportion of the occurrences of term k over the number of unique words n in the dataset as in Equation 1. The IDF is the inverse document frequency that presents the frequency rate of a term across all documents (as in Equation 2), where d_n is the number of documents, and df_k is the number of documents that contain the term k. Hence, the frequent words will have a low TF-IDF scoring and vice versa.

$$TF = \frac{n_k}{n} \tag{1}$$

$$IDF = log_2(\frac{d_n}{df_k}) \tag{2}$$

3.2.2. Hashing vectorizer

The hashing vectorizer is a technique implemented by the scikit-learn library [41] to create a matrix of token occurrences. A key feature of it is that the generated unique textual tokens are not stored in the memory but mapped into special column indexes by hashing, where its value is the token count. The hashing is performed by using the MurmurHash, which is a non-cryptographic hash function [42]. Hashing the tokens has boosted the performance and reduced the used memory especially when dealing with large datasets. However, a limitation of the hashing vectorizer is that the method cannot retrieve the original words from the column indexes.

3.2.3. Document embeddings

Document embeddings are an extension of word embeddings, which in contrast represent each document as a vector. A document can be a short text (i.e., tweet, question), a paragraph, or an article. In this respect, word embeddings are distributed word representations that are created by predictive neural-based models. The main advantage of it is its ability to encode the semantic relationships of words in a corpus by denser vector representations. Hence, it is emerged based on the idea that similar words that appear in the same context, will have similar representations, and high similarity scores. A well-known model for creating word embeddings is Word2Vec that is developed by Google [43]. Word2Vec uses a shallow neural network to create the embeddings where the embedding length represents the number of the hidden layers, which is a hyperparameter to be optimized. Word2Vec has two training structures; the Continuous Bag-of-Words (CBOW), and the Skip-Gram (SG). The former takes a set of context words; in order to predict a target word, while the latter, uses the target word in order to predict the context words. CBOW is more efficient in representing frequent words, while the SG model is better in encoding the infrequent words.

On the other hand, Doc2Vec is a document embedding model that is also created by Google [44]. The Doc2Vec model encompasses the word vectors as well as a document vector. Each document has a unique randomly-initialized vector identifier, while the words' vectors might be shared among the documents. The document vector and the words vectors are concatenated or averaged in order to create the final document's embedding. Thereby, the embedding of a document can be learned by two different training models: the Distributed Memory Model of Paragraph Vectors (PV-DM), and the Distributed Bag-of-Words model of Paragraph Vectors (PV-DBOW). The former is similar to the CBOW, where it predicts and remember a target from the context via a stochastic gradient descent and back-propagation. Whereas, the latter is analogous to the SG model, where it uses the document's vector to learn and classify a set of words whether they belong to the current document or not.

3.3. Question-Symptom-Diagnosis Model (QSDM)

Primarily, this section describes the procedure of developing the QSDM approach. The QSDM is a fusion of two modalities: the first analyzes the symptoms and classifies them into four possible diagnoses. The number of suggested diagnosis is set to four as to match the doctors' preference, since suggesting more than four will be distracting. The second is the question classification modality that predicts maximally four potential diagnoses, where the final prediction depends on combining the results of the two modalities. The structure of the symptoms and question modalities relies on machine learning algorithms as will be discussed in the following subsections.

3.3.1. Logistic regression

LR is a statistical and linear machine learning algorithm for the classification [41], which is popular in the medical and natural language processing applications [45, 46]. It uses a logistic function to model the relationships between the independent variables and a dichotomous dependent variable. The logistic function is a Sigmoid (S-shaped) function that takes a value and transform it into a class label, (see Equation 3), where X is the input value to be transformed and e is the base of the natural logarithms. Mainly, it takes as input the feature vector $X = x_1, x_2, ..., x_n$, where n is the number of features (independent variables) and classifies them into a set of classes $C = c_1, c_2, ..., c_k$, where k is the number of classes.

$$f(x) = \frac{1}{1 + e^X} \tag{3}$$

The implementation of LR in scikit-learn library is regularized by default with various regularizers.

3.3.2. Random forest

RF is an ensemble learning method [47], which is a collection of decision tree classifiers that produce predictions. Each decision tree is constructed based on a different set of features that are drawn from the original feature set. Based on the predictions from all trees, the highly-voted class is considered as the final prediction. The Key advantages of the RF algorithm are its ability to avoid overfitting and to perform relative features importance.

3.3.3. Stochastic gradient descent

The SGDClassifier is a linear classifier implemented by the scikit-learn library that is regularized and trained by the Stochastic Gradient Descent (SGD). The SGD is an optimization algorithm that tunes the algorithm's parameters in order to minimize the cost function. The gradient of the loss function is computed for one random sample each time with a decreasing learning rate, which is faster than the gradient descent that considers the whole dataset while tuning the parameters.

The input of the model is sparse and dense arrays of features in the form of ($n\_samples$,$n\_features$), where the default model it fits is the linear SVM (by setting the loss to *hinge*). SGDClassifier supports various penalties, including the $L1$, $L2$, and the $ElasticNet$.

3.3.4. Multilayer perceptron

The MLP is a multilayer artificial neural network, which is constructed from a set of neurons distributed over a stack of layers. The perceptron is the simplest structure of the neural network that consists of two layers (hidden and output layers). The data flow through the input layer to the hidden layers and then to the output layer in one direction. The MLP is a well-known machine learning algorithm that performs a non-linear mapping of the input to the output via the non-linear activation part of a neuron. Each neuron has weights and bias parameters through which the network learns. The layered structure of neural networks empowers them to capture hierarchical hidden representations within the data when learning and back-propagating the information. During the training, each neuron performs a summation (of the weights w and input I with the bias β) as in Equation 4, where n is the number of input neurons. Whereas, the output (S) is activated by a non-linear function $f(x)$ (e.g. Sigmoid function). Thereby, the final output y_i is obtained by $f_j(S_j)$.

$$S_j = \sum_{i=1}^{n} \omega_{ij} I_i + \beta_j \tag{4}$$

$$f_j(x) = \frac{1}{1 + e^{-S_j}} \tag{5}$$

MLP has been applied successfully in various applications, such as object detection [48], financial forecasting [49], fraudulent detection [50], medical diagnosis [51], and other [52, 53].

3.3.5. Multi-class classification

Multi-class classification problems have naturally more than two classes to differentiate between. The problem is that the machine learning algorithms either originally developed to support binary classification (e.g., LR, SVM), or cannot handle the multi-class problem. However, various methods have been developed to handle the problem, which typically stands on transforming the problem into multiple binary classification problems. Such approaches are the One-Versus-One (OVO), and OVR. The OVO technique divides the problem into multiple binary classifications, where each pair of classes is considered a problem. Therefore, the total number of Classification Problems (CP) is given as in Equation 6, thus, the final output is a majority vote from all constructed classifiers. N_c is the total number of classes.

A major drawback of this technique is that the increasing complexity when having a large number of classes.

$$CP = \frac{N_c \times (N_c - 1)}{2} \tag{6}$$

Whereas, the OVR method divides the problem into a set of binary problems, where the number of constructed binary problems equals to the number of classes. Each problem classifies one class against the rest $(N_c - 1)$ classes, while the final prediction accounts for the one that has the best confident results. Figure 4 illustrates the OVR technique.

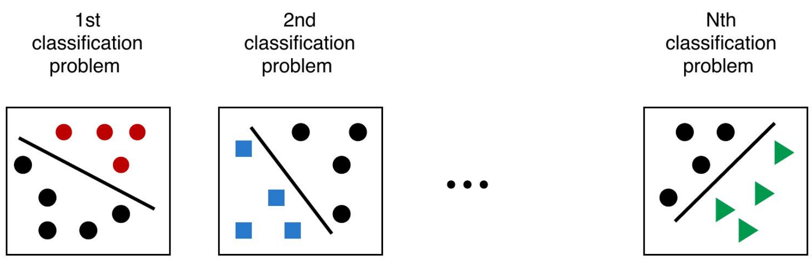

Figure 4: The OVR technique. Each box presents a binary classification problem, where the colored points represent the other classes.

3.3.6. System architecture

Mainly, the QSDM is a fusion of two parts: the symptom detection model and the questions model, as shown in Figure 5. The objective of combining the two modalities is to improve the results of the question model by aggregating informative features from the symptoms model. The symptom model includes all symptoms as binary features, hence, it involves the set of all unique symptoms from the questions (which are 7,324 features). The unique diagnoses are the set of labels (1206), which are represented by binary values. The symptom data is divided into 80% for training, and 20% for testing. The data is fed into various machine learning models, including LR, RF, SGDClassifier, and MLP classifiers. The training set is used to build the learning models, while the testing set is used for evaluating their performances. The developed models are based on the OVR method to deal with the multi-class classification. Each model is trained and tested individually. Though, the

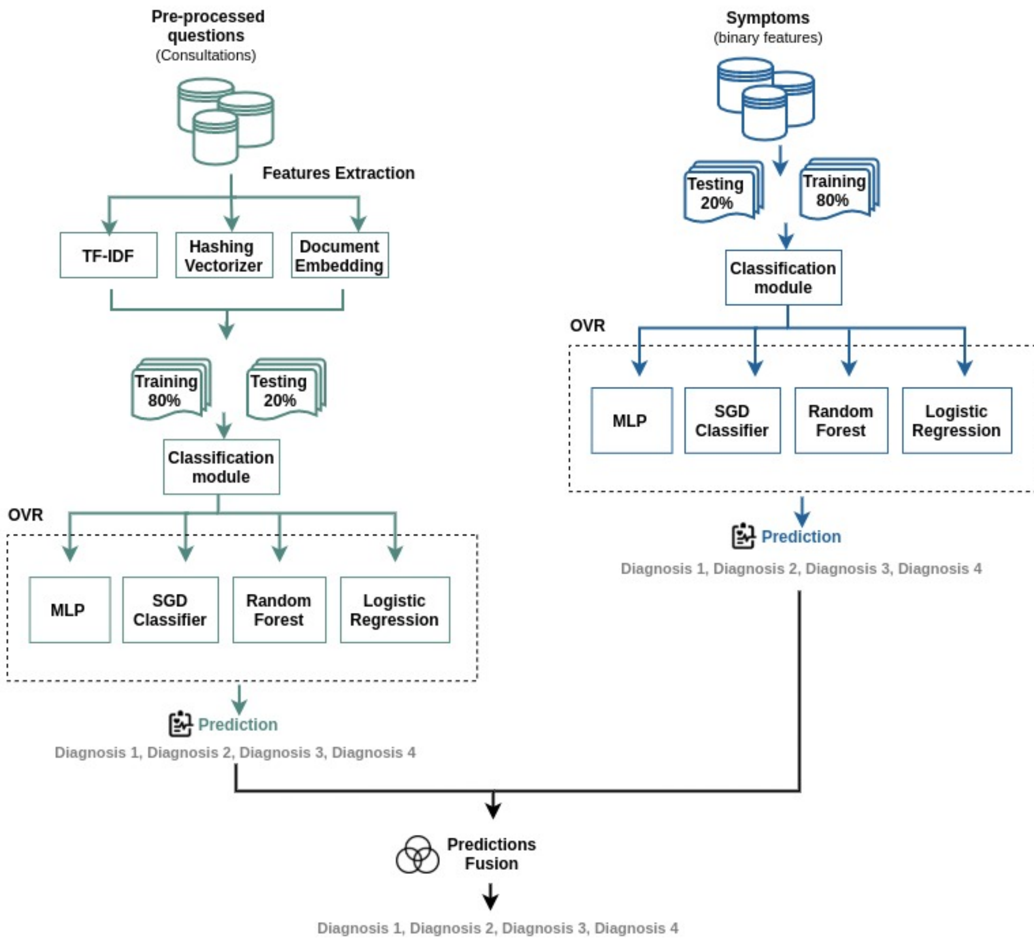

Figure 5: Representation of QSDM system architecture.

final predicted diagnoses are taken from the best performing classifier of this sub-model.

For the questions model, several feature extraction methods were utilized separately (TF-IDF, hashing vectorizer, and document embedding), where the document embedding is implemented via the Doc2Vec. The three generated datasets are divided into (80%, and 20%) for training and testing, respectively. Meanwhile, they are fed into the four classifiers through OVR. Next, the result of the best performing classifier is selected as the final predictions of the question model.

Combining the results of the two models can be performed by different fusion criteria, including the multiplication, the ranking, and the summation. In other words, for the multiplication, it takes the predicted probabilities of the two models (symptoms and questions) and performs an arithmetic multiplication between them, which then returns the highly-scored diagnoses. Similarly is for the summation, where the fusion is achieved by an arithmetic addition. Whereas, for the ranking, the highly-ranked diagnoses (based on the highest accuracy) are selected. In the ranking case, the results were reported in two cases; case one is when there is no repetition of diagnoses from the two models, and if existed (case two), the repeated diagnoses were removed and alternatives were taken from the model of the higher predictive power.

The results of the two modalities (symptoms and questions) were combined together to generate the final output.

3.4. Evaluation criteria

Four quantitative evaluation measures were considered for assessing the performance of the QSDM model; which are the accuracy at different precision levels, the model size, the model loading time, and the inferential time. The accuracy is the ratio of the correct diagnoses out of the total number of the respective diagnoses (m), which is defined by Equations 7 and 8. In Equation 7, P_V represents the probabilities of all diagnoses, where $V = [v_1, v_2, ..v_n]$ given that n equals the number of unique diagnoses. In Equation 8, y is the actual diagnosis of the consultations, and x is the predicted diagnosis. Where (m) is the number of considered diagnoses that is four. P is the probabilities of all diagnoses, and j is the diagnosis index.

$$\texttt{argmax}\ P_V = \{v|\ if\ v > z, \forall z \in V\ \wedge\ z \neq v\} \tag{7}$$

$$\texttt{Accuracy} = \frac{1}{m}\sum_{i}^{m}\ \{f(x) = 1|\ x = argmax(P_X)\ \wedge (x_j = y_j)\} \tag{8}$$

The accuracy is presented in terms of its precision. For example, the accuracy at precision one means that how much the algorithm is precise in retrieving at least one correct diagnosis out of the respective truth diagnoses. This is referred to as Precision_1. Precision_2 indicates the model ability to find at least two correct diagnoses, while Precision_3 refers to finding at least three diagnoses.

The model size is an important measure, especially, knowing that increasing the size of the model (e.g., increasing the number of hidden layers in a deep learning model) will in consequence improve the model's performance. However, it is critical since it might degrade the efficacy in situations where the infrastructure is limited. In addition, the loading time and the inferential time are two relevant metrics indicating the efficiency of the model in generating real-time predictions. The loading time corresponds to the needed time for deploying the model on the web, while the inferential time is the needed amount of time for performing a prediction.

4. Experiments and results

4.1. Experimental settings

The experiments were implemented by using Python version (3.7.3). The hosting machine is a cloud server that is running Ubuntu-1804-bionic-64, the memory capacity is of 64 GB, and the processor is Intel (R) Core (TM) i7-7700, the processor speed is 3.6 GHz, while the GPU is GeForce GTX 1080 of 8 GB.

All algorithms have been implemented based on the scikit-learn library. Regarding the LR algorithm, the penalty is the $L2-$regularizer, and the maximum number of iterations is 500. For the RF, the number of trees is 100, the $Gini - index$ was used for the evaluation of the split, and the maximum number of features for the split was determined by $\sqrt{f_n}$, where f_n is the number of features. In the case of the SGDClassifier, the loss function was set to log to provide probabilities for the output, the penalty is $l2-$regularizer, $\alpha = 0.0001$, the maximum iterations are 1000, and the learning rate is defined by $1.0/(\alpha * (t + t_0))$, where t_0 is a predefined constant, and t is the time step. The settings of the MLP classifier were the defaults based on the scikit-learn library. In which, the activation is based on the $Relu$ function, the optimizer is $Adam$, the learning rate is a constant (0.001), the maximum number of iterations is 200, and the hidden layer size has experimented at 10, 20, 30 and 40, where after this the performance no longer improving.

For the document embeddings, the experiments were implemented depending on Keras deep-learning framework [54], which built on the top of TensorFlow 2.0 [55]. The Doc2Vec model was utilized, for which the maximum number of epochs is 50, the embedding dimension is 500, the learning rate is 0.025, and the window size is three. The training structure of Doc2Vec is set based on the distributed memory model (PV-DM).

4.2. Questions modality-based results

Regarding the questions module, this subsection provides a comparison between the classifiers at different feature extraction methods, including the TF-IDF vectorizer, the hashing vectorizer, and the document embeddings. Table 2 presents the performance in terms of accuracy for the four algorithms based on the TF-IDF vectorizer. It is clear from the table that all algorithms achieved better results when predicted correctly at least one diagnosis (denoted by Precision_1). From the table, the LR algorithm was the best performing classifier that obtained (46.7%). The MLP (10) achieved a very good accuracy of 45.2%, even that it revealed a slight decline in comparison with LR. The MLP (20) and MLP (30), yet could achieve quite good results of (44.0%, 41.4%), respectively. However, the SGDClassifier performed the least (33.5%). Regarding the situation to predict at least two correct diagnoses (Precision_2), also the LR performed the best (40.4%), then the MLP (10) and MLP (20) by having (38.9%, 38%, respectively). Similarly is at predicting at least three correct diagnoses (Precision_3), the LR obtained the best accuracy (39%), then MLP (10), and MLP (20) which had an accuracy of 37.9%, and 37%, respectively.

Such important aspects to consider when developing a machine learning model is its size, the required time to deploy it on the web, and the inferential time to perform a prediction. In this regard, in terms of the M.S., the MLP (10) had a minimum size of 5.2 MB, while the RF was the highest of 17,300 MB. When considering the loading time, the MLP had the lowest time of 0.35 seconds. Whilst, at the prediction, the fastest algorithms were the LR and the SGDClassifier, which were needed 0.06 seconds to perform a prediction. Although the MLP classifiers had the least model sizes as well as the least loading times, the LR can achieve a higher accuracy score. However, this makes the MLP classifiers more preferable for a decision-maker who prioritizes the size and the time more than the accuracy.

Further, Table 3 presents the classifiers' performance when considering the hashing vectorizer, which also exhibits that the best performing classifier was the LR algorithm. The LR accomplished the best accuracy at various precision levels (Precision_1, Precision_2, and Precision_3) by having 45.6%, 39.4%, and 38.3%, respectively. Comparing the LR at the TF-IDF, and at the hashing vectorizer, it is noticeable that there is a slight decline of approximately 1%. For example, it dropped from 46.7% to 45.6% at Precision_1. Moreover, the RF, the SGDClassifier, and the MLP classifiers have experienced a small reduction in the accuracy as well, at Precision_1. This is in

Table 2: The accuracy measure, M.S. (MB), L.T. (seconds), and I.T. (seconds), for LR, RF, SGDClassifier, and MLP classifiers based on TF-IDF vectorizer.

Classifier	**Accuracy**			**M.S.**	**L.T.**	**I.T.**
	Precision_1	**Precision_2**	**Precision_3**			
LR$_{ovr}$	**0.467**	**0.404**	**0.391**	95	0.710	**0.060**
RF$_{ovr}$	0.392	0.331	0.327	17,300	45.89	128.3
SGDClassifier$_{ovr}$	0.335	0.279	0.274	187	0.420	**0.060**
MLP (10)	0.452	0.389	0.379	**5.2**	**0.350**	0.550
MLP (20)	0.440	0.380	0.370	7.9	**0.350**	0.550
MLP (30)	0.414	0.355	0.346	10.6	**0.350**	0.550
MLP (40)	0.386	0.328	0.320	13.3	**0.350**	0.550

contrast to the SGDClassifier that showed a slight increase in the accuracy of 34.7%. Also, the same is at Precision_3, which raised up to 28.1%. Overall, all classifiers gained a better accuracy at Precision_1 in comparison with Precision_2 and Precision_3.

Remarkably, in terms of the pickling size, the MLP classifiers had the least model sizes, where the MLP (10) had a minimum of 2.7 MB, while the RF had the largest size of 14,700 MB. Subsequently, regarding the RF, as it had the largest size, it also had the highest loading and inferential times of 27.45 and 128.1 seconds, respectively. On the contrary, the MLP (10) had a minimum loading time of 0.31 seconds, which also quite relative to the other MLP classifiers. In terms of the inferential time, the SGDClassifier had the lowest inferential time of 0.35 seconds, while the MLP classifiers had on average a 0.49 seconds. To this end, the LR accomplished the best in terms of accuracy, while the MLP classifiers can achieve better regarding the prediction and loading times. Yet, the SGDClassifier is the fastest at prediction.

Regarding the Doc2Vec embedding, it is clear from Table 4 that the MLP classifiers performed the best when predicted 25%, 50%, and 75% of the diagnoses. The MLP (40) obtained the best by having 30.3%, 25%, and 24.4%, respectively. It can be seen that MLP (20) and the LR achieved almost the same performance in terms of accuracy. However, the MLP (20) had a lower model size, and lower loading and inferential times, which give them a higher privilege over the LR. Additionally, even that the SGDClassifier performed as closely as the MLP (10), but the MLP also had a better performance in

Table 3: The accuracy measure, M.S. (MB), L.T. (seconds), and I.T. (seconds), for LR, RF, SGDClassifier, and MLP classifiers based on hashing vectorizer.

Classifier	**Accuracy**			**M.S.**	**L.T.**	**I.T.**
	Precision_1	**Precision_2**	**Precision_3**			
LR_{ovr}	**0.456**	**0.394**	**0.383**	92	0.380	0.550
RF_{ovr}	0.377	0.318	0.301	14,700	27.45	128.1
$\text{SGDClassifier}_{ovr}$	0.347	0.290	0.281	92.8	0.380	**0.350**
MLP (10)	0.427	0.366	0.356	**2.7**	**0.310**	0.480
MLP (20)	0.428	0.368	0.358	5.4	0.320	0.490
MLP (30)	0.402	0.343	0.335	8.1	0.320	0.490
MLP (40)	0.376	0.318	0.310	10.8	0.320	0.490

terms of the model size and the inferential time. Further, it is obvious that the RF failed to achieve any better results neither at the accuracy nor the model size nor the loading and inferential times.

Further, regarding either the model size, the loading, or inferential times, the MLP classifiers achieved the best results. For instance, the MLP (10) had the minimum pickling size (0.448 MB) and the minimum inferential time (0.020 seconds). Whereas the MLP (40), yet can have a relatively small model size (1.7 MB), and a fast prediction ability of (0.02 seconds). Even the document and word embeddings alongside the MLP classifiers can produce very good results, but it is expected that increasing the amount of training data will in consequence improve the results as well. This is considered by the authors as a next step to utilize a larger training dataset.

4.3. Symptoms modality-based results

Table 5 shows the results of the symptoms modality based on LR, RF, SGDClassifier, and four variants of MLP regarding the accuracy, model's size, inferential time, and loading time. It is clear from the table that the MLP (40) achieved the highest accuracy at the precision_1, precision_2, and precision_3, while the SGDClassifier achieved the worst. For instance, regarding the accuracy at precision_1, the MLP (40) obtained 85.2%, whereas, the SGDClassifier obtained 74.3%. Regarding the model's size, generally, the MLP has a smaller model size than the LR, RF, or the SGDClassifier. Similarly, in terms of the loading and inferential times, the MLP achieved the best performance by having 0.02, and 0.31 seconds, respectively. Even that

Table 4: The accuracy measure, M.S. (MB), L.T. (seconds), and I.T. (seconds), for LR, RF, SGDClassifier, and MLP classifiers based on Doc2Vec embeddings

Classifier	**Accuracy**			**M.S.**	**L.T.**	**I.T.**
	Precision_1	**Precision_2**	**Precision_3**			
LR_ovr	0.292	0.240	0.235	5.6	0.350	0.050
RF_ovr	0.105	0.078	0.061	502	1.820	0.810
SGDClassifier_ovr	0.267	0.217	0.212	5.8	0.340	0.050
MLP (10)	0.266	0.216	0.211	**0.448**	0.330	**0.020**
MLP (20)	0.294	0.242	0.236	0.848	**0.310**	**0.020**
MLP (30)	0.300	0.249	0.243	1.3	**0.310**	**0.020**
MLP (40)	**0.303**	**0.250**	**0.244**	1.7	0.320	**0.020**

the RF classifier, achieved the highest model size and loading and inferential times, which was expected since it has higher computational complexity than the other classifiers.

Table 5: The accuracy measure, M.S. (MB), L.T. (seconds), and I.T. (seconds), for LR, RF, SGDClassifier, and MLP classifiers based on the symptoms model.

Classifier	**Accuracy**			**M.S.**	**L.T.**	**I.T.**
	Precision_1	**Precision_2**	**Precision_3**			
LR_{ovr}	0.847	0.820	0.809	72	0.370	0.080
RF_{ovr}	0.848	0.818	0.806	1,254	3.700	1.100
$SGDClassifier_{ovr}$	0.743	0.704	0.691	72	0.470	0.100
MLP (10)	0.820	0.773	0.761	**2.1**	**0.310**	**0.020**
MLP (20)	0.848	0.812	0.801	4.1	**0.310**	**0.020**
MLP (30)	0.851	0.818	0.806	6.2	**0.310**	**0.020**
MLP (40)	**0.852**	**0.818**	**0.807**	8.2	**0.310**	**0.020**

4.4. Results of Fusion-based prediction

This subsection shows the results after combining the predictions of the questions with the predictions of the symptoms. The fusion of the two modules has shown powerful capability in improving the prediction results and providing more reliable differential diagnosis.

Table 6 shows the performance of the final combined models, where it describes the accuracy scores when predicting 25%, 50%, and 75% of the diagnoses (denoted by Precision_1, Precision_2, and Precision_3) across four fusion criteria (Ranking-I, Ranking-II, Summation, and Multiplication). It is clear that the best-obtained accuracy was at Precision_1. However, the fusion that is based on the multiplication, accomplished the best accuracy score of 84.9%, then the summation (84.6%), next is Ranking-I, and Ranking-II by having 82.8%, and 81.3%, respectively. Furthermore, even that Precision_2 and Precision_3 are relatively close in their performance, but there is a clear dramatic difference between Precision_3 and Precision_1.

Table 6: The accuracy score of the final prediction based on four fusion criteria: the ranking of case I (Ranking-I), and of case II (Ranking-II), the summation, and multiplication.

	Accuracy			
	Ranking-I	**Ranking-II**	**Summation**	**Multiplication**
Precision_1	**0.813**	**0.828**	**0.846**	**0.849**
Precision_2	0.761	0.784	0.809	0.811
Precision_3	0.741	0.769	0.796	0.798

4.5. Qualitative evaluation

For further assessment of the developed system, a qualitative analysis based on expert evaluation is conducted. The experts are specialized doctors who will use the clinical portal as a DDSS. Ninety expert doctors who collaborate with Altibbi for providing medical consultations have examined the results of the classification model using an online portal for doctors. The doctors' portal shows the consultation and its expected diagnoses, hence, the doctors label the accuracy of the diagnoses by four levels of precision. If the model produced 100% accurate diagnoses, or if it is accurate from 80% to 90%, from 70% to 80%, or from 50% to 60%.

Furthermore, the qualitative evaluation of the proposed module is presented by the pie chart in Figure 6. The chart shows that most of the predicted diagnoses are accurate by the precision of (80-90)% with a percentage of 44.9%, while 34.8% of the diagnoses are accurate by a level of (70-80)%. Moreover, 10% is accurate by a percentage of (50-60)%, and the last 10% is accurate 100%. Markedly, the results of the qualitative analysis presented by the experts match the results of the quantitative analysis from the proposed module, which indicates the robustness of the model and the trustworthiness of predicted diagnoses.

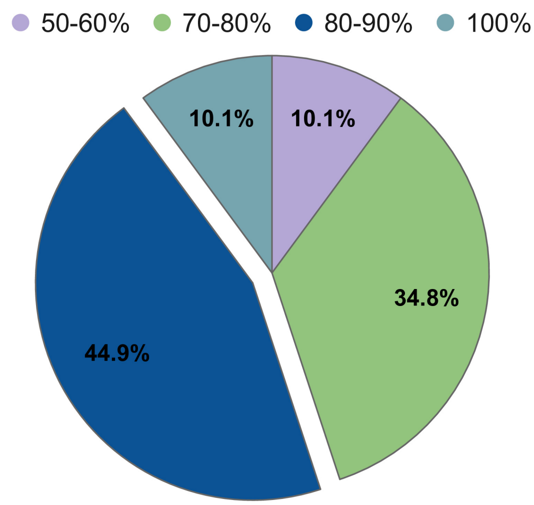

Figure 6: The qualitative analysis based on Altibbi expert doctors. The percentages inside the pie correspond to the proportion of consultations and their diagnoses, while the four colors represent the four levels of accuracy.

To illustrate more, one of Altibbi's doctors received a consultation that was a condition of a runny-nose. The possible diagnoses suggested by the developed model were two relevant and two irrelevant, the relevant diagnoses were the common cold, and allergic rhinitis, whereas, the irrelevant were tension headache, and fever. However, the doctor chose the common cold as the correct diagnosis. Based on the qualitative evaluation of Altibbi's doctors, the developed QSDM model is facing some limitations; first, sometimes the suggested diagnosis might have duplicates, for example, to suggest the common cold twice. Second, some symptoms might be related to a very common condition, but this condition might not be suggested by the model. As in the mentioned example, the common cold might not originally be suggested. These limitations might hinder the doctors from making the correct decision or make the diagnosis process slower. Therefore, tackling these limitations is essential to improve the developed QSDM model.

5. Conclusion and future work

Providing accurate differential diagnosis is hard, since, primarily, at an early stage of a disease the symptoms are unclear and overlapping. Developing a computer-aided diagnosis system to help clinicians in performing a trustworthy differential diagnosis is of significant importance. This article proposed a multimodal machine learning-based diagnostic system that helps Altibbi's doctors in making differential diagnosis decisions of clinical consultations. The proposed approach is a fusion of two modalities; the symptoms and the questions. Various machine learning algorithms have been utilized into the two modalities to make a differential diagnosis, this includes the LR, RF, SGDClassifier, and different variants of the MLP classifier. The questions module has utilized various feature extraction methods (i.e., TF-IDF, hashing vectorizer, and document embeddings). The final model represents a late fusion of two models, where the fusion is performed based on various approaches, such as ranking, summation, and multiplication. The fusion-based on multiplication achieved the highest performance in terms of accuracy (84.9%). In consequence, this can be a promising model for a decision support system that can perform a differential diagnosis process. However, improving the accuracy of the model is of serious importance. The increasing number of consultations in Altibbi provides a valuable asset to increase the performance of the proposed model. Furthermore, this consequently, increases the structural symptomatic features. Having large-scale data opens additional opportunities for applying advanced computational techniques in order to achieve higher accuracy, such as deep learning and transformers methods. Moreover, adding the results of diagnostic tests and labs could be a third modality that can improve the classification accuracy, and alleviate the model's limitations.

References

[1] W. H. O. WHO, Diagnostic errors, 1988.

[2] R. Mahumud, M. Sultana, N. Sheikh, M. Ali, D. Mitra, A. Sarker, Diagnostic errors in low and middle-income countries: Future health and economic burden for patient safety, Austin Emerg Med (2016).

[3] H. Jimison, P. Sher, J. Jimison, Clinical decision support systems: Theory and practice, Decision Support for Patients (2007) 249–261.

[4] S. Montani, M. Striani, Artificial intelligence in clinical decision support: a focused literature survey, Yearbook of medical informatics 28 (2019) 120.

[5] P. Cerrato, J. Halamka, Chapter five - how mobile technology and ehrs can personalize healthcare, in: P. Cerrato, J. Halamka (Eds.), Realizing the Promise of Precision Medicine, Academic Press, 2018, pp. 93 – 117.

[6] C. S. Royce, M. M. Hayes, R. M. Schwartzstein, Teaching critical thinking: a case for instruction in cognitive biases to reduce diagnostic errors and improve patient safety, Academic Medicine 94 (2019) 187–194.

[7] E. E. Bron, M. Smits, J. M. Papma, R. M. Steketee, R. Meijboom, M. De Groot, J. C. van Swieten, W. J. Niessen, S. Klein, Multiparametric computer-aided differential diagnosis of alzheimer's disease and frontotemporal dementia using structural and advanced mri, European radiology 27 (2017) 3372–3382.

[8] A. Atutxa, A. D. de Ilarraza, K. Gojenola, M. Oronoz, O. Perez-de Viñaspre, Interpretable deep learning to map diagnostic texts to icd-10 codes, International Journal of Medical Informatics 129 (2019) 49–59.

[9] J. Zhang, Z. Yin, P. Chen, S. Nichele, Emotion recognition using multimodal data and machine learning techniques: A tutorial and review, Information Fusion 59 (2020) 103–126.

[10] E. Vardell, C. Bou-Crick, Visualdx: a visual diagnostic decision support tool, Medical reference services quarterly 31 (2012) 414–424.

[11] J. A. Gegundez-Fernandez, J. I. Fernandez-Vigo, D. Diaz-Valle, R. Mendez-Fernandez, R. Cuina-Sardina, E. Santos-Bueso, J. M. Benitez-del Castillo, Uvemaster: a mobile app-based decision support system for the differential diagnosis of uveitis, Investigative Ophthalmology & Visual Science 58 (2017) 3931–3939.

[12] A. Papakonstantinou, H. Kondylakis, E. Marakakis, Integra: a web-based differential diagnosis system combining multiple knowledge bases, in: Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments, 2020, pp. 1–6.

[13] J. Yoon, S. Lee, C.-H. Sun, D. Kim, I. Kim, S.-S. Yoon, D. Oh, H. Yun, Y. Koh, Med-tma: A clinical decision support tool for differential diagnosis of tma with enhanced accuracy using an ensemble method, Thrombosis Research (2020).

[14] Q.-Y. Zhong, E. W. Karlson, B. Gelaye, S. Finan, P. Avillach, J. W. Smoller, T. Cai, M. A. Williams, Screening pregnant women for suicidal behavior in electronic medical records: diagnostic codes vs. clinical notes processed by natural language processing, BMC medical informatics and decision making 18 (2018) 30.

[15] A. Brown, J. Kachura, Natural language processing of radiology reports in patients with hepatocellular carcinoma to predict radiology resource utilization, Journal of the American College of Radiology 16 (2019) 840–844.

[16] D. Xue, A. Frisch, D. He, Differential diagnosis of heart disease in emergency departments using decision tree and medical knowledge, in: Heterogeneous Data Management, Polystores, and Analytics for Healthcare, Springer, 2019, pp. 225–236.

[17] H. Liu, Y. Xu, Z. Zhang, N. Wang, Y. Huang, Z. Yang, R. Jiang, H. Chen, A natural language processing pipeline of chinese free-text radiology reports for liver cancer diagnosis, arXiv preprint arXiv:2004.13848 (2020).

[18] T. Searle, Z. Ibrahim, R. Dobson, Comparing natural language processing techniques for alzheimer's dementia prediction in spontaneous speech, arXiv preprint arXiv:2006.07358 (2020).

[19] Y. Tong, K. Lu, Y. Yang, J. Li, Y. Lin, D. Wu, A. Yang, S. Yu, J. Qian, et al., Can natural language processing help differentiate inflammatory intestinal diseases in china? models applying random forest and convolutional neural network approaches, BMC medical informatics and decision making (2020).

[20] C. Küpper, S. Stroth, N. Wolff, F. Hauck, N. Kliewer, T. Schad-Hansjosten, I. Kamp-Becker, L. Poustka, V. Roessner, K. Schultebraucks, et al., identifying predictive features of autism spectrum disorders in a clinical sample of adolescents and adults using machine learning, Scientific reports 10 (2020) 1–11.

[21] M. A. Elaziz, K. M. Hosny, A. Salah, M. M. Darwish, S. Lu, A. T. Sahlol, New machine learning method for image-based diagnosis of covid-19, Plos one 15 (2020) e0235187.

[22] E. Fathi, M. J. Rezaee, R. Tavakkoli-Moghaddam, A. Alizadeh, A. Montazer, Design of an integrated model for diagnosis and classification of pediatric acute leukemia using machine learning, Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine 234 (2020) 1051–1069.

[23] T. B. Chandra, K. Verma, Pneumonia detection on chest x-ray using machine learning paradigm, in: Proceedings of 3rd International Conference on Computer Vision and Image Processing, Springer, 2020, pp. 21–33.

[24] E. Aydin, İ. U. Türkmen, G. Namli, Ç. Öztürk, A. B. Esen, Y. N. Eray, E. Eroğlu, F. Akova, A novel and simple machine learning algorithm for preoperative diagnosis of acute appendicitis in children, Hernia 353 (2020) 4–9.

[25] O. Jacobson, H. Dalianis, Applying deep learning on electronic health records in swedish to predict healthcare-associated infections, in: Proceedings of the 15th workshop on biomedical natural language processing, 2016, pp. 191–195.

[26] H. Shi, P. Xie, Z. Hu, M. Zhang, E. P. Xing, Towards automated icd coding using deep learning, arXiv preprint arXiv:1711.04075 (2017).

[27] D. Guo, M. Li, Y. Yu, Y. Li, G. Duan, F.-X. Wu, J. Wang, Disease inference with symptom extraction and bidirectional recurrent neural network, in: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2018, pp. 864–868.

[28] S. Rabhi, J. Jakubowicz, M.-H. Metzger, Deep learning versus conventional machine learning for detection of healthcare-associated infections in french clinical narratives, Methods of information in medicine 58 (2019) 031–041.

[29] S. Nuthakki, S. Neela, J. W. Gichoya, S. Purkayastha, Natural language processing of mimic-iii clinical notes for identifying diagnosis and procedures with neural networks, arXiv preprint arXiv:1912.12397 (2019).

[30] S. S. Azam, M. Raju, V. Pagidimarri, V. C. Kasivajjala, Cascadenet: An lstm based deep learning model for automated icd-10 coding, in:

Future of Information and Communication Conference, Springer, 2019, pp. 55–74.

[31] S. Kalra, L. Li, H. R. Tizhoosh, Automatic classification of pathology reports using tf-idf features, arXiv preprint arXiv:1903.07406 (2019).

[32] J. S. Obeid, E. R. Weeda, A. J. Matuskowitz, K. Gagnon, T. Crawford, C. M. Carr, L. J. Frey, Automated detection of altered mental status in emergency department clinical notes: a deep learning approach, BMC medical informatics and decision making 19 (2019) 164.

[33] P. Morillo, H. Ortega, D. Chauca, J. Proaño, D. Vallejo-Huanga, M. Cazares, Psycho web: a machine learning platform for the diagnosis and classification of mental disorders, in: International Conference on Applied Human Factors and Ergonomics, Springer, 2019, pp. 399–410.

[34] G. Castellazzi, M. G. Cuzzoni, M. Cotta Ramusino, D. Martinelli, F. Denaro, A. Ricciardi, P. Vitali, N. Anzalone, S. Bernini, F. Palesi, et al., A machine learning approach for the differential diagnosis of alzheimer and vascular dementia fed by mri selected features, Frontiers in neuroinformatics 14 (2020) 25.

[35] S. Poletti, B. Vai, M. G. Mazza, R. Zanardi, C. Lorenzi, F. Calesella, S. Cazzetta, I. Branchi, C. Colombo, R. Furlan, et al., A peripheral inflammatory signature discriminates bipolar from unipolar depression: A machine learning approach, Progress in Neuro-Psychopharmacology and Biological Psychiatry 105 (2020) 110136.

[36] B. S. Fernandes, C. Karmakar, R. Tamouza, T. Tran, J. Yearwood, N. Hamdani, H. Laouamri, J.-R. Richard, R. Yolken, M. Berk, et al., Precision psychiatry with immunological and cognitive biomarkers: a multi-domain prediction for the diagnosis of bipolar disorder or schizophrenia using machine learning, Translational Psychiatry 10 (2020) 1–13.

[37] Y. Liu, A. Jain, C. Eng, D. H. Way, K. Lee, P. Bui, K. Kanada, G. de Oliveira Marinho, J. Gallegos, S. Gabriele, et al., A deep learning system for differential diagnosis of skin diseases, Nature Medicine (2020) 1–9.

[38] A. B. Oktay, A. Kocer, Differential diagnosis of parkinson and essential tremor with convolutional lstm networks, Biomedical Signal Processing and Control 56 (2020) 101683.

[39] J. Born, N. Wiedemann, G. Brändle, C. Buhre, B. Rieck, K. Borgwardt, Accelerating covid-19 differential diagnosis with explainable ultrasound image analysis, arXiv preprint arXiv:2009.06116 (2020).

[40] E. Loper, S. Bird, Nltk: The natural language toolkit, in: In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics, 2002, pp. 63–70.

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, the Journal of machine Learning research 12 (2011) 2825–2830.

[42] A. Appleby, Murmurhash 2.0, 2008.

[43] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[44] Q. Le, T. Mikolov, Distributed representations of sentences and documents, in: International conference on machine learning, 2014, pp. 1188–1196.

[45] K. Selig, Bayesian information criterion approximations for model selection in multivariate logistic regression with application to electronic medical records, Ph.D. thesis, Technische Universität München, 2020.

[46] S. D. Swamy, S. Laddha, B. Abdussalam, D. Datta, A. Jamatia, Nitagartala-nlp-team at semeval-2020 task 8: Building multimodal classifiers to tackle internet humor, arXiv preprint arXiv:2005.06943 (2020).

[47] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.

[48] A. Dhillon, G. K. Verma, Convolutional neural network: a review of models, methodologies and applications to object detection, Progress in Artificial Intelligence 9 (2020) 85–112.

[49] Z. Alameer, M. Abd Elaziz, A. A. Ewees, H. Ye, Z. Jianhua, Forecasting gold price fluctuations using improved multilayer perceptron neural network and whale optimization algorithm, Resources Policy 61 (2019) 250–260.

[50] S. Kataria, M. T. Nafis, Internet banking fraud detection using deep learning based on decision tree and multilayer perceptron, in: 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 2019, pp. 1298–1302.

[51] M. Hosseinzadeh, O. H. Ahmed, M. Y. Ghafour, F. Safara, S. Ali, B. Vo, H.-S. Chiang, et al., A multiple multilayer perceptron neural network with an adaptive learning algorithm for thyroid disease diagnosis in the internet of medical things, The Journal of Supercomputing (2020) 1–22.

[52] A. H. Elsheikh, S. W. Sharshir, M. Abd Elaziz, A. Kabeel, W. Guilan, Z. Haiou, Modeling of solar energy systems using artificial neural network: A comprehensive review, Solar Energy 180 (2019) 622–639.

[53] H. Moayedi, M. Mosallanezhad, A. S. A. Rashid, W. A. W. Jusoh, M. A. Muazu, A systematic review and meta-analysis of artificial neural network application in geotechnical engineering: theory and applications, Neural Computing and Applications (2020) 1–24.

[54] F. Chollet, et al., Keras, www.keras.io, (access date: Sep, 2020), 2015.

[55] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al., Tensorflow: Large-scale machine learning on heterogeneous systems, arXiv preprint (2015).