Disease diagnosis system based on text samples using NLP and deep learning algorithm

[1]V.Manideep reddy, Information Technology, Tkr college of engineering and technology, RN Reddy Colony, Meerpet, Hyderabad, Telangana 500097, vootlamanideepreddy@gmail.com

[2]G.Akshay nayak, 20K91A1222, Information Technology, Tkr college of engineering and technology, RN Reddy Colony, Meerpet, Hyderabad, Telangana 500097 akshaynayak0703@gmail.com

[3]Abdul Abeed, 20k91a1201, Information Technology, Tkr college of engineering and technology, RN Reddy Colony, Meerpet, Hyderabad, Telangana 500097 , abdulabeed0001@gmail.com

[4]B Shiva Charan, 20K91A1259, Information Technology, Tkr college of engineering and technology, RN Reddy Colony, Meerpet, Hyderabad, Telangana 500097 sc606605@gmail.com

D Kavitha, Assistant professor, Information Technology, Tkr college of engineering and technology, RN Reddy Colony, Meerpet, Hyderabad, Telangana 500097 , davidikavitha2011@gmail.com

Abstract:

Now a days some rural people and blind getting sudden death due to lack of medical services and doctors. According to WHO yearly 10 million people lasting due to inadequacy of treatment and doctors. In this research work an advanced application is designed on cloud using deep learning technology. In this work patient diseases are taken as input in the form of text or speech samples. Already available medical dataset is trained by deep learning algorithms, for testing purpose patient data is using. The patient data and training data is analysed through NLP techniques and the chatbot is giving answers to patient. More over in any emergency cases alert messages can be send to minimum 5 family members. At final calculating test accuracy, F1 score, sensitivity and Recall using confusion matrix. This model is outperformance the technique and compete with present technologies.

Keywords: NLP, Chatbot, deep learning, Disease diagnosis

Introduction

Creating a disease diagnosis system based on a chatbot involves leveraging natural language processing (NLP) and machine learning techniques. Here's a general outline of the steps involved in developing such a system: Clearly define the purpose and scope of your disease diagnosis chatbot. Determine the types of diseases it will be able to diagnose and the level of complexity involved. Gather relevant medical data, including symptoms, diagnoses, and treatment information. You may need datasets with labeled examples for training your machine learning model. Clean and preprocess the collected data. This may involve handling missing values, normalizing data, and converting text data into a suitable format for analysis. Implement NLP techniques to understand and interpret user inputs. This involves tokenization, stemming, and other text processing methods. Create a knowledge base that contains information about various diseases, symptoms, and possible treatments. This can be used to provide informative responses to users. Train a machine learning model to predict diseases based on user inputs. Choose a suitable algorithm based on the complexity of the problem. Common approaches include decision trees, support vector machines, or more advanced methods like neural networks. Choose a chatbot framework (such as Dialogflow, Microsoft Bot Framework, or Rasa) and integrate your disease diagnosis model with it. This allows users to interact with the chatbot through natural language. Design an intuitive and user-friendly interface for the chatbot. Ensure that users can easily input their symptoms and receive relevant information. Thoroughly test your chatbot for different scenarios and user inputs. Validate its accuracy in diagnosing diseases against known cases. Implement strong security measures to protect user data, especially when dealing with sensitive health information. Ensure compliance with healthcare data protection regulations. Collect feedback from users and continuously improve the chatbot's performance. This may involve updating the knowledge base, retraining the machine learning model, or enhancing NLP capabilities. Consider legal and ethical aspects, including compliance with healthcare regulations, informed consent, and privacy policies.within computer science, particularly a subset of artificial intelligence (AI), dedicated to equipping computers with the capability to comprehend both written and spoken language, akin to human understanding. NLP integrates computational linguistics, which involves rule-based modeling of human language, with statistical, machine learning, and deep learning models. These combined technologies empower computers to analyze human language, whether presented as text or voice data, and to grasp its complete meaning, encompassing the speaker or writer's intentions and sentiments. NLP underpins various computer programs, facilitating tasks such as language translation, response to spoken commands, and swift summarization of extensive text, often in real time. Chances are high that you've encountered NLP in everyday applications like voice-operated GPS systems, digital assistants, speech-to-text dictation software, customer service chatbots, and other user-friendly conveniences. Beyond consumer applications, NLP is increasingly integral to enterprise solutions, contributing to the optimization of business operations, enhancement of employee productivity, and streamlining of critical business processes. Human language is replete with complexities that pose significant challenges for developing software capable of accurately discerning the intended meaning from text or voice data. Homonyms, homophones, sarcasm, idioms, metaphors, grammar and usage exceptions, and variations in sentence structure are just a few of the intricacies inherent in human language. While humans naturally learn to navigate these nuances over years, programmers must impart this understanding to natural language-driven applications from the outset for them to be genuinely useful. Various tasks in Natural Language Processing (NLP) are designed to deconstruct human text and voice data, aiding computers in making sense of the information they ingest. Some of these tasks include: Speech Recognition (Speech-to-Text): This task involves reliably converting voice data into text data, essential for applications that respond to voice commands or answer spoken questions. The challenge lies in the diverse ways people speak—quickly, with slurred words, varying emphasis and intonation, different accents, and sometimes using incorrect grammar. Part of Speech Tagging (Grammatical Tagging): This process determines the part of speech of a word or text based on its use and context. For instance, it identifies 'make' as a verb in "I can make a paper plane" and as a noun in "What make of car do you own?" Word Sense Disambiguation: This task involves selecting the correct meaning of a word with multiple meanings through semantic analysis, determining the sense that fits best in the given context. For example, it helps distinguish the meaning of the verb 'make' in "make the grade" (achieve) vs. "make a bet" (place). Named Entity Recognition (NEM): NEM identifies words or phrases as meaningful entities, recognizing, for instance, 'Kentucky' as a location or 'Fred' as a man's name. Co-reference Resolution: This task identifies when two words refer to the same entity, such as determining that 'she' refers to 'Mary.' It can also involve identifying metaphors or idioms in the text. Sentiment Analysis: Sentiment analysis attempts to extract subjective qualities like attitudes, emotions, sarcasm, confusion, or suspicion from text. Natural Language Generation: Described as the opposite of speech recognition, this task involves putting structured information into human language, completing the cycle by generating text from non-linguistic data.

Literature survey

Support Vector Machines (SVMs) have been extensively studied and applied in various fields. Below are some seminal papers and key literature on SVM: "A Tutorial on Support Vector Machines for Pattern Recognition" Christopher J.C. Burges Published in Data Mining and Knowledge Discovery, 1998. This foundational tutorial provides a comprehensive introduction to SVMs, explaining the principles, optimization, and practical considerations. "Support-Vector Networks" Corinna Cortes and Vladimir Vapnik Published in Machine Learning, 1995. The original paper introducing Support Vector Machines, providing insights into the mathematical foundations and the formulation of the SVM algorithm. "Learning to Classify Text Using Support Vector Machines" Thorsten Joachims Published in KDD '98 Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, 1998. Focuses on the application of SVMs in text classification, highlighting their effectiveness in natural language processing tasks. "A Practical Guide to Support Vector Classification" Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin Published in Technical Report, 2003. This guide provides practical insights into tuning parameters and applying SVMs to classification problems, particularly using the popular software LIBSVM. "Support Vector Machines: A Survey" I. Guyon and A. Elisseeff Published in Machine Learning, 2002. A comprehensive survey of SVMs, covering various aspects such as formulations, kernel functions, and applications in different domains. "Introduction to Support Vector Machines" Nello Cristianini and John Shawe-Taylor Published in Cambridge University Press, 2000. Part of the book "An Introduction to Support Vector Machines and Other Kernel-BasedLearning Methods," this work provides a thorough introduction to SVMs and related concepts. "SVM Tutorial"Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik Published in Statistics and Computing, 2002. A tutorial that covers the basics of SVM, kernel selection, parameter tuning, and the use of SVM in different domains. "Support Vector Machines for Classification and Regression" Steve R. Gunn Published in ISIS Technical Report, 1997. This technical report provides insights into the theory and application of SVMsfor both classification and regression tasks. "LIBSVM: A Library for Support Vector Machines" Chih-Chung Chang and Chih-Jen Lin Published in ACM Transactions on Intelligent Systems and Technology, 2011. Describes LIBSVM, a widely used library for SVMs, discussing its design, usage, and practical considerations. "Support Vector Machines in Machine Learning" B. Schölkopf, C. J. C. Burges, and A. J. Smola Published in Neural Information Processing Systems (NIPS), 1999. A comprehensive survey covering various aspects of SVMs, including their theoretical foundations, algorithmic implementations, and practical applications. These

papers offer a mix of foundational concepts, practical guidance, and applications of SVMs, making them valuable resources for researchers, practitioners, and students interested in understanding and utilizing Support Vector Machines.

Convolutional Neural Networks (CNNs) have been widely researched and applied in the field of computer vision. Here are some influential papers and key literature on CNNs: "ImageNet Classification with Deep Convolutional Neural Networks" Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton Published in Advances in Neural Information Processing Systems (NIPS), 2012. This landmark paper introduces the AlexNet architecture, a deep CNN that achieved a significant breakthrough in image classification accuracyon the ImageNet dataset. "Very Deep Convolutional Networks for Large-Scale Image Recognition" Karen Simonyan and Andrew Zisserman Published in arXiv, 2014. The authors propose the VGGNet architecture, emphasizing the importance of depth in CNNs. VGGNet became a widely used and referenced model in image classification tasks. "Going Deeper with Convolutions" Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich Published in Computer Vision and Pattern Recognition (CVPR), 2015. This paper introduces the Inception architecture (GoogLeNet), emphasizing the use of inception modules to capture multi-scale features efficiently. "Rethinking the Inception

Architecture for Computer Vision" Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna Published in arXiv, 2015. An extension of the Inception architecture, introducing improvements like batch normalization and factorization to enhance training and generalization. "Deep Residual Learning for Image Recognition" Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun Published in Computer Vision and Pattern Recognition (CVPR), 2016. Introduces the ResNet architecture, which employs residual learning to facilitate the training of very deep networks. ResNet has become a pivotal architecture in deep learning. "Densely Connected Convolutional Networks" Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger Published in Computer Vision and Pattern Recognition (CVPR), 2017. This paper introduces DenseNet, which establishes dense connections between layers to address issues of vanishing gradients and feature reuse. "Visualizing and Understanding Convolutional Networks" Matthew D. Zeiler and Rob Fergus Published in European Conference on Computer Vision (ECCV), 2014. The authors propose a visualization technique, called Deconvolutional Networks, to understand and interpret the learned features in CNNs. "Understanding Neural Networks Through Deep Visualization" Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson Published in arXiv, 2015. Explores visualization techniques to gain insights into the representations learned by CNNs, providing a better understanding of their inner workings. "YOLO9000: Better, Faster, Stronger" Joseph Redmon and Santosh Divvala Published in Computer Vision and Pattern Recognition (CVPR), 2017.

Methodology

For convolutional neural networks, the VGG16 CNN design is used. It remains one of the greatest vision model designs to this day. Instead of a slew of hyper-parameters, the 3x3 filter's convolution layers with stride1 and the 2x2 filter's padding and maxpool layer with stride 2 were utilised for VGG16's most distinctive feature. The convolution and max pool layers are placed in the same way throughout the architecture. Two FC (fully connected layers) and a softmax are used for output. VGG16 has 16 weighted layers, as indicated by the number "16" in its name. This network contains approximately 138 million variables shown in Figure 1.

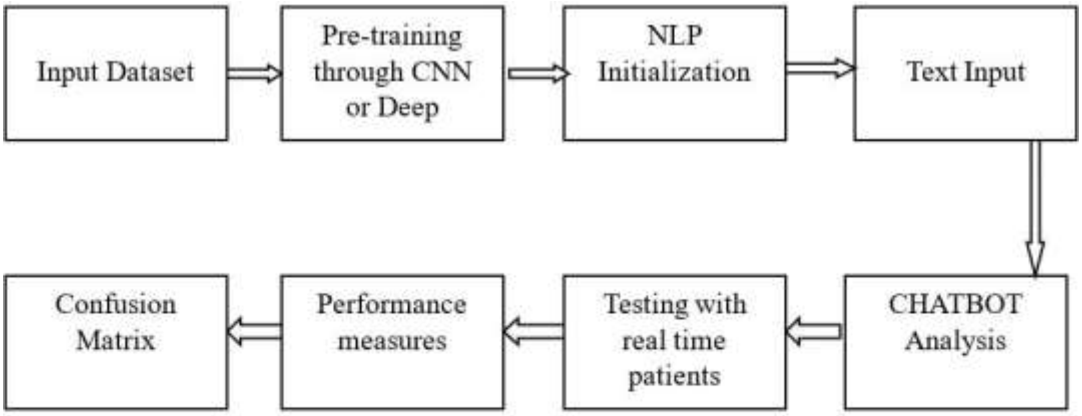


Figure: 1 block diagram

Diagrams in the Unified Modeling Language (UML) based on use cases are known as use cases in the UML. Use cases, actors, and any dependencies between them are all depicted graphically to give a clear picture of the system's capabilities. A use case diagram's primary goal isto explain how the system performs for each actor. System actors can be depicted by their responsibilities in the system figure 1. A Real Time Emergency Diseases Diagnosis System Based on Text Samples Using NLP & Deep Learning Algorithms In this research we are using disease and symptoms dataset to train Deep Learning CNN algorithm and after training user can input symptoms as text based data or voice based data then CHATBOT will predict disease using CNN and display predicted disease to user and at the same time predicted disease details will be saved inside database and EMAIL message will be sent to user registered MAIL ID. To process disease and symptoms text data we have used NLP techniques To implement this project we have designed following modules

1. **Register Here:** using this module user can signup with application and has to give mail ID to received messages after disease prediction
2. **User:** using this module user can signup with application and when system starts up then application will train CNN algorithm
3. **Text Based Chat:** using this module user can enter some symptoms and then CHATBOT will call CNN and NLP to predict disease and then display to user
4. **Voice Based Chat:** Using this module user can speak symptoms and then CHATBOT will recognize voice and then using CNN and NLP will predict disease and display to user. All prediction disease details will be saved in database and sent to user registered EMAIL
5. **Index Page:** this page will perform prediction on TEST data and then calculate accuracy, precision, recall, FSCORE and confusion matrix graph

Results and discussion

In this section results and discussion was explained clearly, the chatbot and deep learning techniques

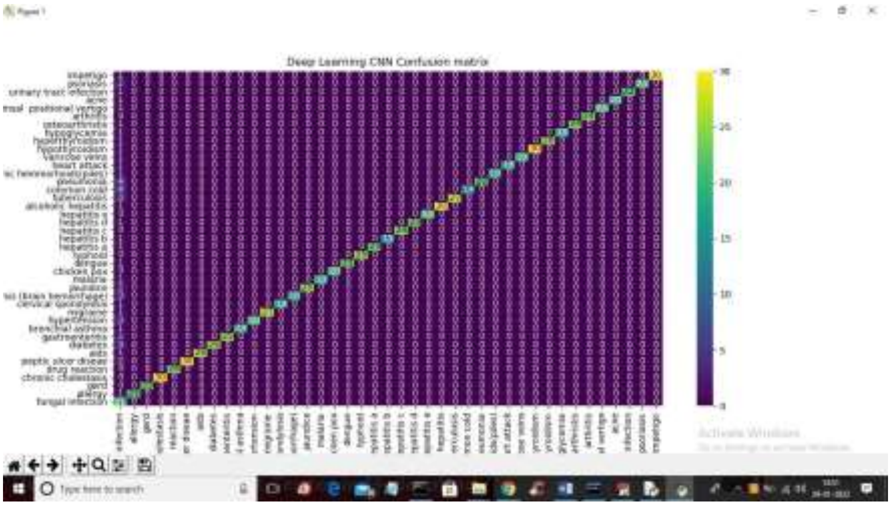
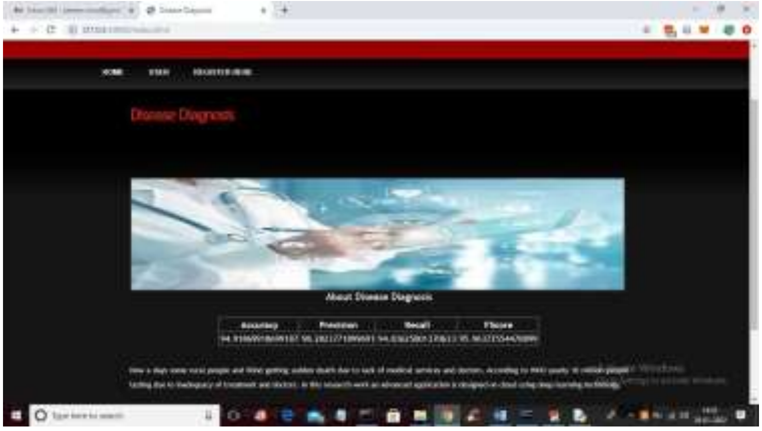


Figure : 2 dataset analysis

In above confusion matrix graph x-axis and y-axis represents disease names and in both x-axis we can seemaximum diseases are correctly predicted and now close above graph to get below



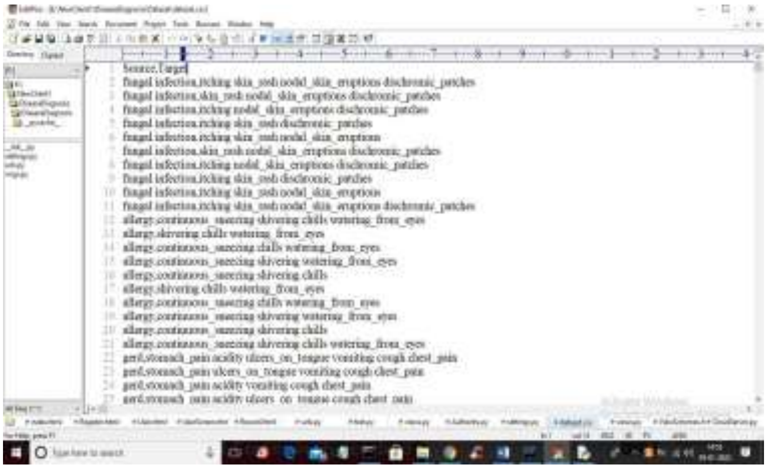
About Disease Diagnosis

Accuracy	Precision	Recall	FScore
[illegible]	[illegible]	[illegible]	[illegible]

screen

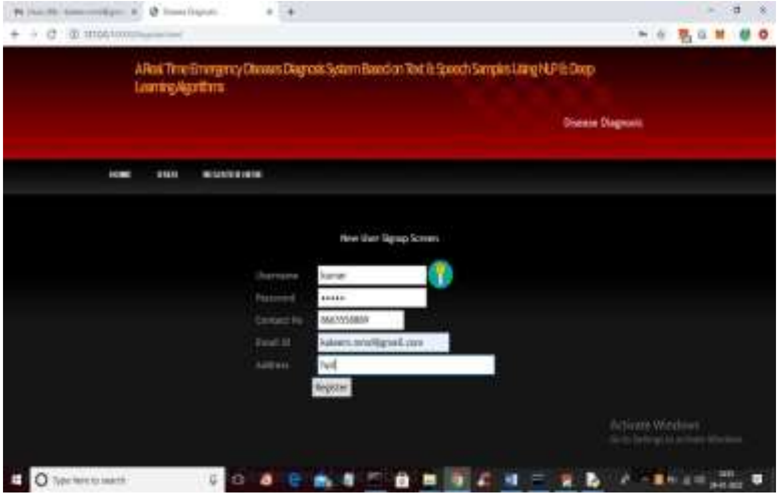
Figure :3 dashboard analysis

In above screen we got CNN accuracy as 94% on disease dataset and below is the dataset screen



```
Source,Target
fungal infection,itching skin_rash nodal_skin_eruptions dischromic_patches
fungal infection,skin_rash nodal_skin_eruptions dischromic_patches
fungal infection,itching nodal_skin_eruptions dischromic_patches
fungal infection,itching skin_rash dischromic_patches
fungal infection,itching skin_rash nodal_skin_eruptions
fungal infection,skin_rash nodal_skin_eruptions dischromic_patches
fungal infection,itching nodal_skin_eruptions dischromic_patches
fungal infection,itching skin_rash dischromic_patches
fungal infection,itching skin_rash nodal_skin_eruptions
fungal infection,itching skin_rash nodal_skin_eruptions dischromic_patches
allergy,continuous_sneezing shivering chills watering_from_eyes
allergy,shivering chills watering_from_eyes
allergy,continuous_sneezing chills watering_from_eyes
allergy,continuous_sneezing shivering watering_from_eyes
allergy,continuous_sneezing shivering chills
allergy,shivering chills watering_from_eyes
allergy,continuous_sneezing chills watering_from_eyes
allergy,continuous_sneezing shivering watering_from_eyes
allergy,continuous_sneezing shivering chills
allergy,continuous_sneezing shivering chills watering_from_eyes
gerd,stomach_pain acidity ulcers_on_tongue vomiting cough chest_pain
gerd,stomach_pain ulcers_on_tongue vomiting cough chest_pain
gerd,stomach_pain acidity vomiting cough chest_pain
gerd,stomach_pain acidity ulcers_on_tongue cough chest_pain
```

used for thisproject. In above screen ‘Source’ represents disease names and ‘Target’ represents symptoms. Now in applicationscreen click on ‘Register’ link to get below screen



A Real Time Emergency Diseases Diagnosis System Based on Text & Speech Samples Using NLP & Deep Learning Algorithms

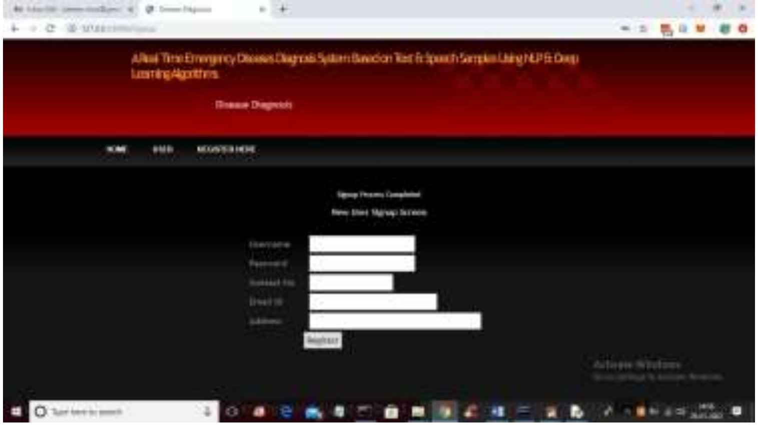
Disease Diagnosis

New User Signup Screen

Username: kumar
Password: *****
Contact No: [illegible]
Email Id: [illegible]
Address: hyd

Figure :4 login details

In above screen user is signup and now press ‘Register’ button to get below output



A Real Time Emergency Diseases Diagnosis System Based on Text & Speech Samples Using NLP & Deep Learning Algorithms

Disease Diagnosis

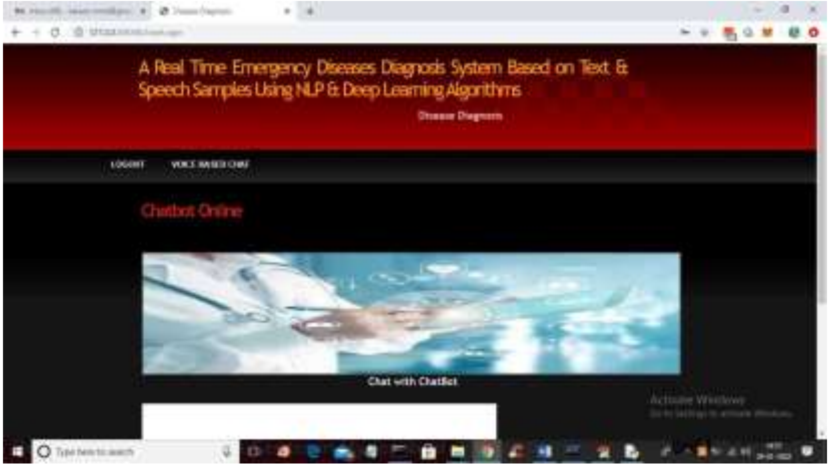
Signup Process Completed

New User Signup Screen

Username
Password
Contact No
Email Id
Address

Figure: 5 login with CNN

In above screen user signup process completed and now click on ‘User’ link to get below login screen



A Real Time Emergency Diseases Diagnosis System Based on Text & Speech Samples Using NLP & Deep Learning Algorithms

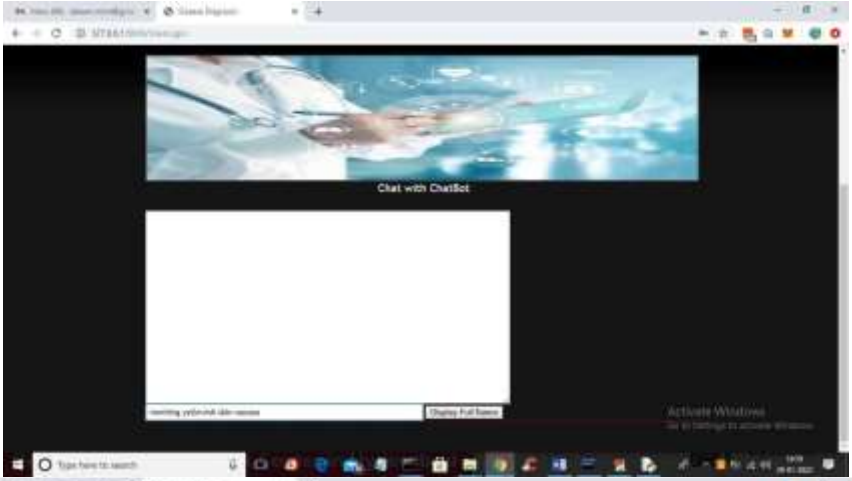
Disease Diagnosis

Chatbot Online

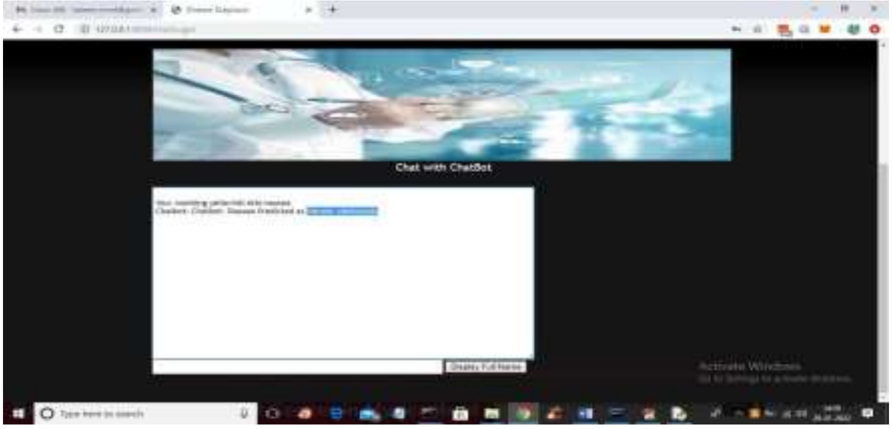
Chat with ChatBot

In above screen click on ‘Voice Based Chat’ if you want to do voice based chat or if you want text based chatthen scroll down above same application screen to enter symptoms like below screen

In above screen I entered symptoms as ‘vomiting yellowish skin nausea’ and then chat bot willgive belowt


Chat with ChatBot

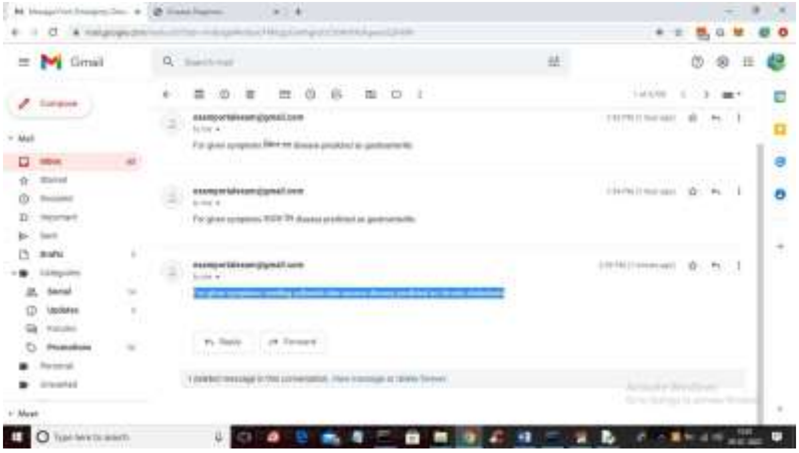
vomiting yellowish skin nausea


Chat with ChatBot

You: vomiting yellowish skin nausea
ChatBot : ChatBot: Disease Predicted as [illegible]

In above screen in blue colour text chat bot displaying predicted disease and similarly you can enter symptomsto get disease details and similarly details will get in EMAIL also like below screen

In above screen in selected blue text we can see user has received mail and similar you can click on 'VoiceBased Chat' link to get below screen



examportalexam@gmail.com

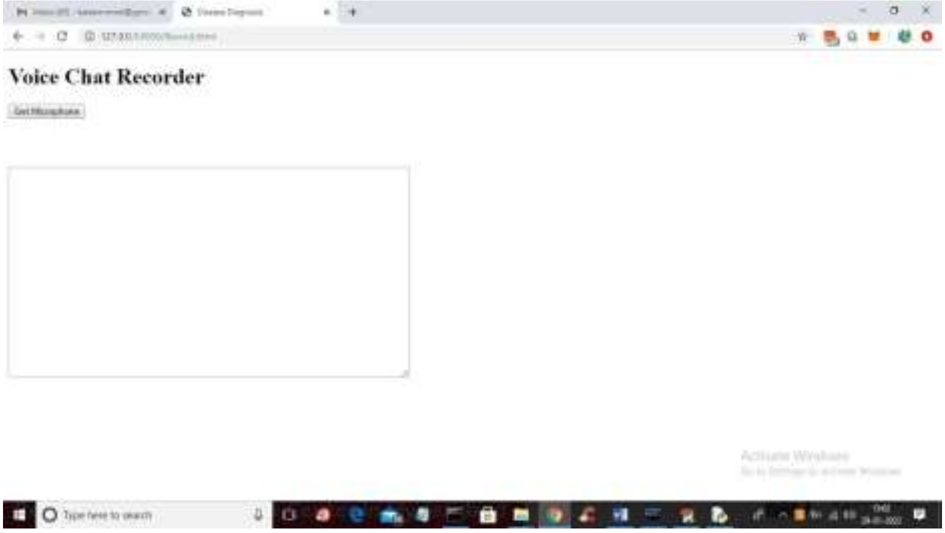
For given symptoms [illegible] disease predicted as gastroenteritis

examportalexam@gmail.com

For given symptoms [illegible] disease predicted as gastroenteritis

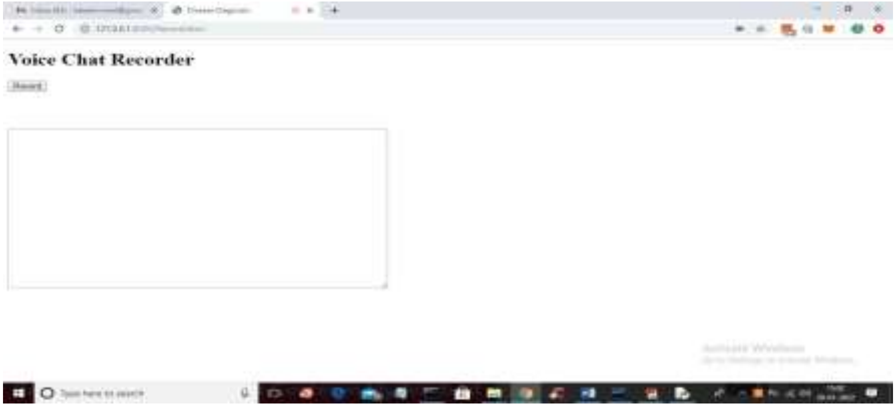
examportalexam@gmail.com

[illegible]



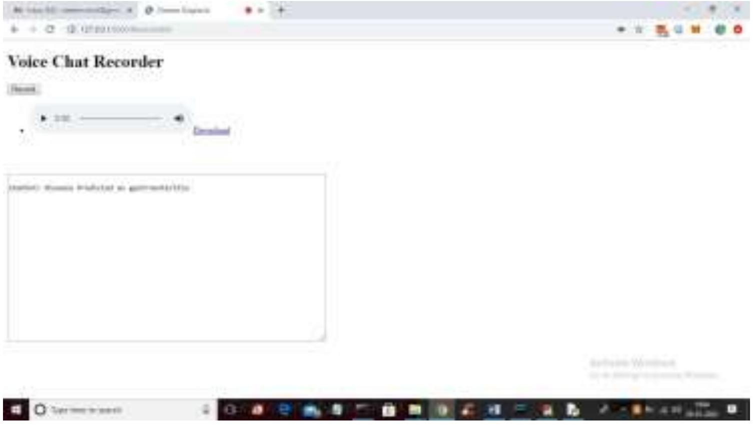
Voice Chat Recorder

In above screen click on 'Get Microphone' to connect to micro phone and to get below screen



Voice Chat Recorder

In above screen user can click on 'Record' and then speak some symptoms and then click stop (which willappear after clicking on ‘Record’ button) button to get disease prediction

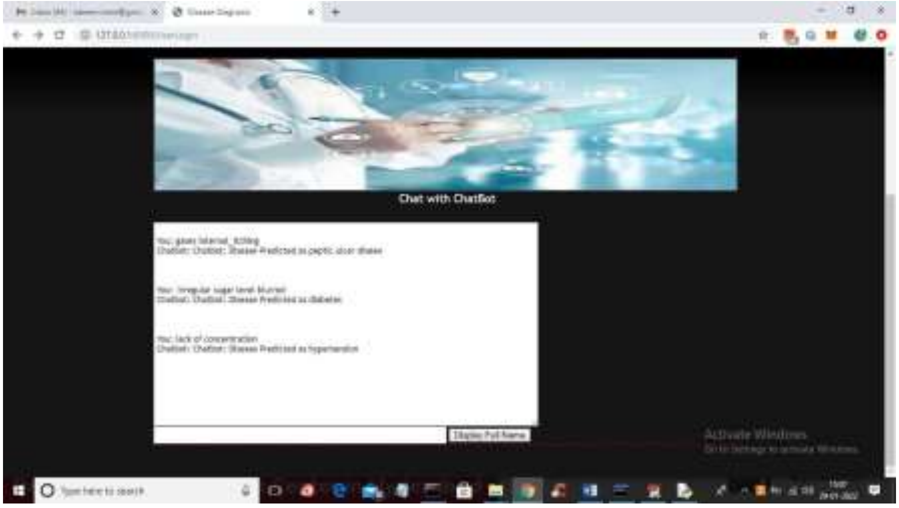


Voice Chat Recorder

Chatbot: Disease Predicted as gastroenteritis

In above screen voice is recorded and you can play that voice also and in chat bot based on symptoms chat botdisplaying disease name as ‘gastroenteritis’ and similarly by using Voice and text you can send any symptoms

In above screen I sent 3 different symptoms and chat bot replied disease for each symptoms



Chat with ChatBot

You: gases internal_itching
Chatbot: Chatbot: Disease Predicted as peptic ulcer diseae

You: irregular sugar level blurred
Chatbot: Chatbot: Disease Predicted as diabetes

You: lack of concentration
Chatbot: Chatbot: Disease Predicted as hypertension

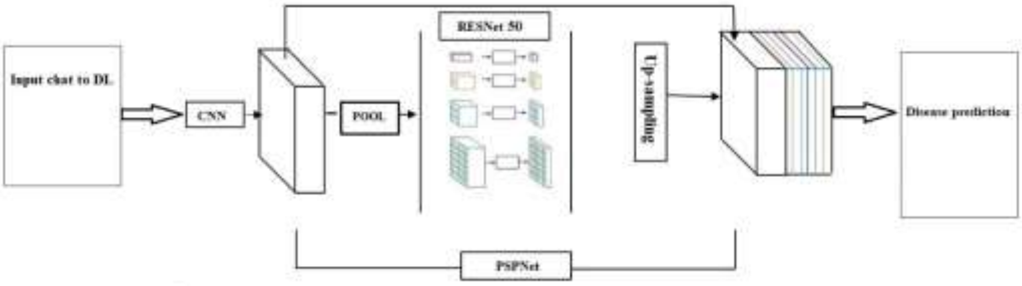


Figure:6 architectural diagram

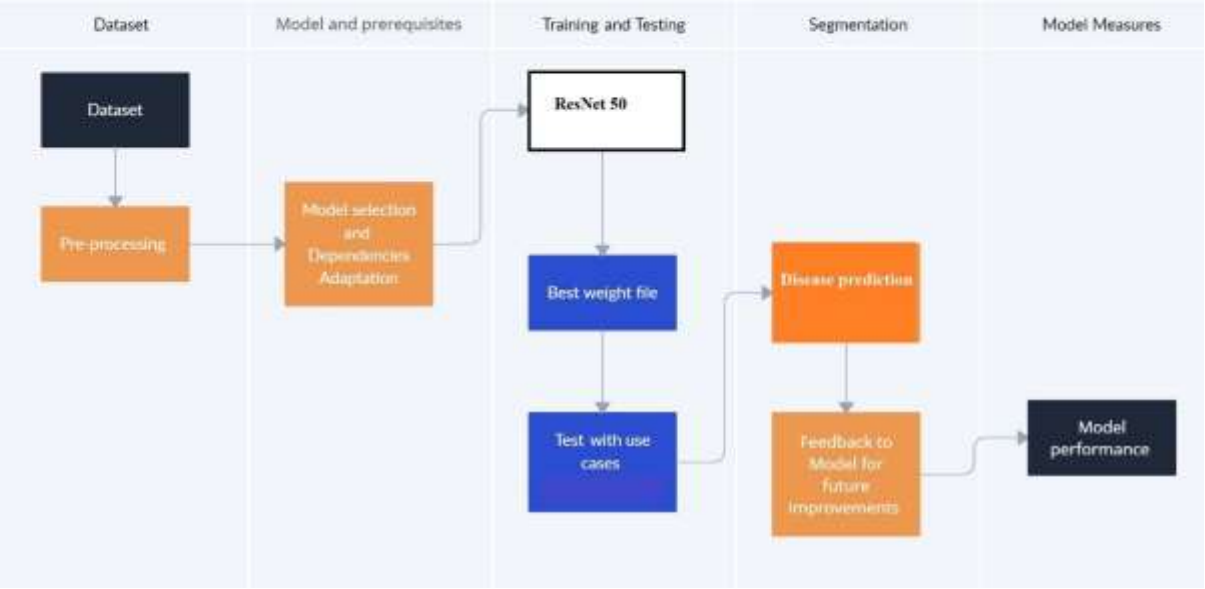
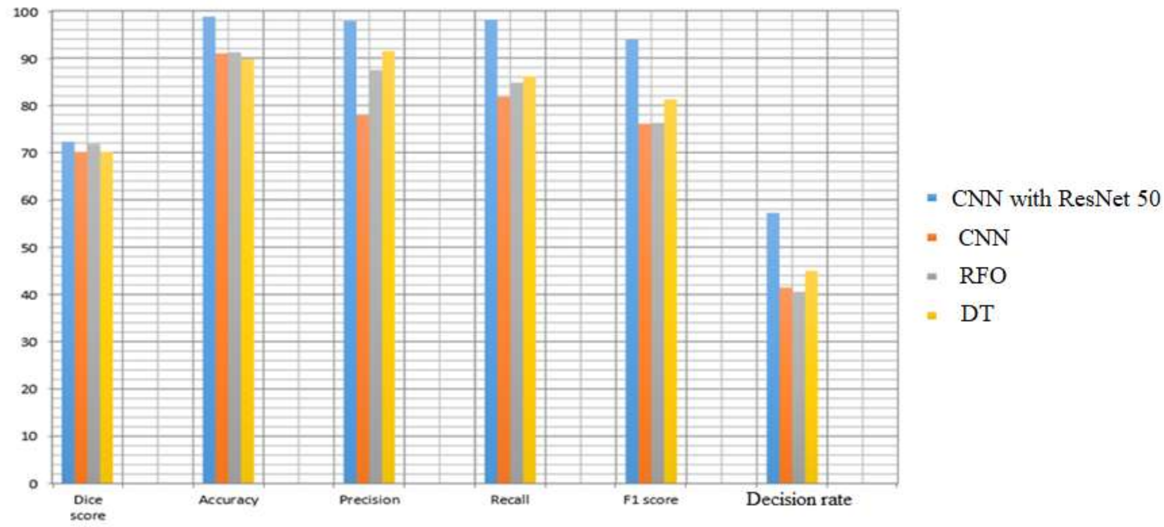


Figure :7 proposed block diagram

We have used disease and symptoms dataset to train Deep Learning CNN algorithm and after training user can input symptoms as text based data or voice based data then CHATBOT will predict disease using CNN and display predicted disease to user and at the same time predicted disease details will be saved inside database and EMAIL message will be sent to user registered MAIL ID.To process disease and symptoms text data we have used NLP techniques.

- ResNet 50 is light weight model.
- It can process high computational operations with less latency.
- The disease deduction is done simultaneously when input is given in the form of text, this is done in fraction of seconds.
- This type of operations supported by ResNet 50.
- In this we have Dense layer, flatten layer hidden layer, max pooling layer. At last we have convolutional layer.
- And with combination of all these layers we have another layer we have Relu layer.



Conclusion

We can predict the disease by giving the disease symptoms. This project is helpful for the people who are in rural areas. In this work patient diseases are taken as input in the form of text or speech samples. Alreadyavailable medical dataset is trained by deep learning algorithms, for testing purpose patient data is using. The patient data and training data is analysed through NLP techniques and the chatbot is giving answers to patient. More over in any emergency cases alert messages can be send to minimum 5 family members. At final calculating test accuracy, F1 score, sensitivity and Recall using confusion matrix. This model is outperformance the technique and compete with present technologies. In this project we got 95% accuracy using deep learning CNNResNet algorithm.

References

[1] Zhao, H. M., Li, D. Y., Deng, W., & Yang, X. H. (2017). Research on vibration suppression method of alternating current motor based on fractional order control strategy. Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering, 231(4), 786-799.

[2] Deng, W., Liu, H., Xu, J., Zhao, H., & Song, Y. (2020). An improved quantum-inspired differential evolution algorithm for deep belief network. IEEE Transactions on Instrumentation and Measurement, 69(10), 7319-7327.

[3] Hoermann, S., McCabe, K. L., Milne, D. N., & Calvo, R. A. (2017). Application of synchronous text- based dialogue systems in mental health interventions: systematic review. Journal of medical Internet research, 19(8), e7023.

[4] Roca, S., Sancho, J., Garcia, J., & Alesanco, Á. (2020). Micro service Chabot architecture for chronic patient support. Journal of Biomedical Informatics, 102, 103305.

[5] Sheth, A., Yip, H. Y., & Shekarpour, S. (2019). Extending patient-chat bot experience with internet-of- things and background knowledge: case studies with healthcare applications. IEEE intelligent systems, 34(4), 24-30.

[6] Doan, S., Maehara, C. K., Chaparro, J. D., Lu, S., Liu, R., Graham, A., & Pediatric Emergency Medicine Kawasaki Disease Research Group. (2016). Building a natural language processing tool to identify patients with high clinical suspicion for Kawasaki disease from emergency department notes. Academic Emergency Medicine, 23(5), 628-636.

[7] Sheth, A., Yip, H. Y., & Shekarpour, S. (2019). Extending patient-Chabot experience with internet-of- things and background knowledge: case studies with healthcare applications. IEEE intelligent systems, 34(4), 24-30.

[8] Leaman, R., Khare, R., & Lu, Z. (2015). Challenges in clinical natural language processing for automated disorder normalization. Journal of biomedical informatics, 57, 28-37.

[9] Žitkus, V., Butkienė, R., Butleris, R., Maskeliūnas, R., Damaševičius, R., & Woźniak, M. (2019). Minimalistic approach to reference resolution in Lithuanian medical records. Computational and mathematical methods in medicine, 2019.

[10] Lu, M., Fang, Y., Yan, F., & Li, M. (2019). Incorporating domain knowledge into natural language inference on clinical texts. IEEE Access, 7, 57623-57632.

[11] Ouerhani, N., Maalel, A., & Ben Ghézela, H. (2020). SPeCECA: a smart pervasive chat bot for emergency case assistance based on cloud computing. Cluster Computing, 23(4), 2471-2482.

[12] Boland, M. V., Chiang, M. F., Lim, M. C., Wedemeyer, L., Epley, K. D., McCannel, C. A., ... & American Academy of Ophthalmology Medical Information Technology Committee. (2013). Adoptionof electronic health records and preparations for demonstrating meaningful use: an American Academy of Ophthalmology survey. Ophthalmology, 120(8), 1702-1710.

[13] Hang, L., Choi, E., & Kim, D. H. (2019). A novel EMR integrity management based on a medical block chain platform in hospital. Electronics, 8(4), 467.

[14] Lee, S., Mohr, N. M., Street, W. N., & Nadkarni, P. (2019). Machine learning in relation to emergency medicine clinical and operational scenarios: an overview. Western Journal of Emergency Medicine, 20(2), 219.

[15] Pineda, A. L., Ye, Y., Visweswaran, S., Cooper, G. F., Wagner, M. M., & Tsui, F. R. (2015). Comparisonof machine learning classifiers for influenza detection from emergency department free-text reports. Journal of biomedical informatics, 58, 60-69.