information

MDPI

Article

Identifying a Medical Department Based on Unstructured Data: A Big Data Application in Healthcare

Veena Bansal [1,*], Abhishek Poddar [2] and R. Ghosh-Roy [3]

[1] Indian Institute of Technology Bhilai, Raipur-492015, India
[2] Indian Institute of Technology Kanpur, Kanpur-208016, India; abhishek.poddar.1511@gmail.com
[3] IBM United Kingdom Limited, London SE1 9PZ, UK; rana.ghosh-roy@btconnect.com
* Correspondence: veena@iitbhilai.ac.in

Received: 5 December 2018; Accepted: 9 January 2019; Published: 11 January 2019

Abstract: Health is an individual's most precious asset and healthcare is one of the vehicles for preserving it. The Indian government's spend on healthcare system is relatively low (1.2% of GDP). Consequently, Secondary and Tertiary government healthcare centers in India (that are presumed to be of above average ratings) are always crowded. In Tertiary healthcare centers, like the All India Institute of Medical Science (AIIMS), patients are often unable to articulate their problems correctly to the healthcare center's reception staff, so that these patients to be directed to the correct healthcare department. In this paper, we propose a system that will scan prescriptions, referral letters and medical diagnostic reports of a patient, process the input using OCR (Optical Character Recognition) engines, coupled with image processing tools, to direct the patient to the most relevant department. We have implemented and tested parts of this system wherein a patient enters his symptoms and/or provisional diagnosis; the system suggests a department based on this user input. Our system suggests the correct department 70.19% of the time. On further investigation, we found that one particular department of the hospital was over-represented. We eliminated the department from the data and performance of the system improved to 92.7%. Our system presently makes its suggestions using random forest algorithm that has been trained using two information repositories-symptoms and disease data, functional description of each medical department. It is our informed assumption that, once we have incorporated medicine information and diagnostics imaging data to train the system; and the complete medical history of the patient, performance of the system will improve further.

Keywords: healthcare; big data; unstructured data; tertiary healthcare

1. Introduction

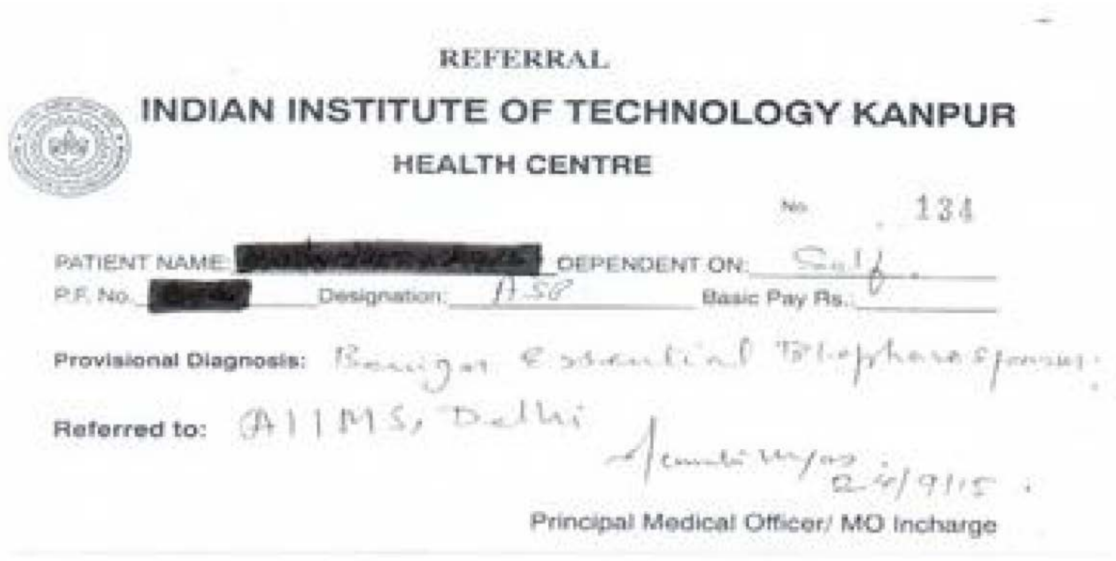
India ranked 143rd among the 188 countries evaluated on 33 health-related Sustainable Development Goal (SDG) indicators [1]. Clearly, India is not doing well in healthcare compared to its peers. India spends only 1.2% of its GDP on healthcare [2] that translates to USD 15 per person whereas USA spends USD 4802 and the UK spends USD 3500. Despite low public spending on healthcare, there are many health care facilities that are funded by the central government.

Healthcare in India is a three-tier system; Primary care is the first line of contact, often between a patient and a doctor. Secondary and Tertiary healthcare centers require a referral from a Primary healthcare center. Tertiary healthcare centers cater for complicated medical conditions and require specialized medical consultations.

A sample referral is shown in Figure 1. The referral has the name of a patient, provisional diagnosis and the hospital name to which the patient has been referred to but without the details of

the department within the hospital. The Tertiary healthcare centers such as AIIMS (All India Institute of Medical Science) have multiple departments, with near unique capabilities in each department for treating ailments. Even medically literate patients often have difficulty in identifying the correct department. The healthcare center's reception staff is often the first port of call and these staff often quickly browse through the medical documents of a patient to identify the appropriate department; this is not foolproof and mistakes are often made, leading to inconveniences for all parties concerned. This is a major bottleneck, especially as the system must deal with many thousands of patients each day. For instance, nearly 10,000 patients line up at the outpatient department everyday at AIIMS, Delhi.

People who have access to the Internet, and have the required skill sets, can collate information about each department before making an online appointment. However, for many people in India, they do not even have access to the Internet and/or are not literate enough to make an online appointment.


REFERRAL

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

HEALTH CENTRE

No. 134

PATIENT NAME [redacted] DEPENDENT ON: Self.

P.F. No. [redacted] Designation: Basic Pay Rs.

Provisional Diagnosis: Benign essential Blepharospasm.

Referred to: AIIMS, Delhi

Principal Medical Officer/ MO Incharge

Figure 1. A Sample Referral to a Tertiary Healthcare System.

Irrespective of the channel used for booking, all walk-in patients face very similar challenges of identifying the correct department to proceed to. We have therefore focused on the walk-in process where most of the errors have been noticed. It was our conclusion that we need to first augment the manual appointment booking process to identify the correct department, thereby making the overall booking process easier and error free for the patients. In this work, we propose a system that will automatically recommend an appropriate department to the patients by looking at their medical documents. We have reviewed the related work in Section 2 and presented a formal model of our proposed system in Section 3. An implementation of our system has been detailed in Section 4. Results and conclusions have been presented in Sections 5 and 6 respectively.

2. Related Work

Over the years, several computational systems for decision making have been used in healthcare. These have either helped humans in reducing their workload or helped in decision making or both. Expert systems have been built to diagnose a disease [3–6].

These systems deploy machine learning models such as decision trees, Bayesian classifiers, artificial neural networks, support vector machines and k-nearest neighbors. These models require supervised learning involving a large amount of labeled historical data. Some domains have an additional challenge if there are little or no standards. In medical domain, standards are still

evolving [7]. Due to lack of standards, it is a challenge to represent medical history of a patient, medicines prescribed and medical diagnostic reports.

Supervised learning techniques have also been used in building expert systems. A decision support system can also be rules or fuzzy rules-based [5]. These systems are used for diagnosing the presence of a disease, or predicting adverse effects of a drug (Fosamax), or predicting the onset of a disease [6,8,9].

Another line of research led to the development of systems that helped patients in managing their diet and medicines [10,11]. Some helped Health Insurance Providers with pre-authorization of insurance requests [12]; others helped doctors in identifying the best possible treatment for a given disease [13], even recommending pathological tests [14] or check the efficacy of an ongoing treatment [15].

All these systems use machine learning models that require supervised learning involving a vast amount of data [16]. With the advent of Big Data [17] framework, it is now possible to handle *volume*, *variety* and *velocity* aspects of the data to draw *value* from the data using machine learning techniques. Big data framework has opened up a new set of possibilities in healthcare [18]. A big data integrated framework has been proposed to help in prevention and control of HIV/AIDS, Tuberculosis and Silicosis in the mining industry [19]. The big data capability of this framework lies in the fact that it can combine several small datasets which could potentially turn into a massive dataset to do analysis and provide useful insights. Big data framework, specifically the Hadoop/Map Reduce framework along with predictive analytics s been used to build a system to predict prevalent Diabetic Mellitus if any, possible complications associated with it and to recommend a treatment [20].

People search the Internet primarily for medical information involving information about a specific condition (97%) and a visit to the doctor (57%) [21]. Making an appointment is a complex social process [22] involving receptionists, appointments allocation rules and sharing of clinical information. In this paper, we present a system that utilizes clinical information available to clearly identify the medical department in which appointment should be made. Our system uses description of each medical department and clinical information of the medical condition to suggest appropriate medical department.

We spoke with three doctors in Secondary and Tertiary healthcare facilities, and they all confirmed that patients are often directed to the wrong department by the Reception staff. One of the authors of the paper ended up in the examination room of the doctor who was not the right doctor for the author. Sometimes, patients are not even able to describe their problems. Often the Reception staff are unable to decipher the medical reports/documents provided by the patients. A patient often therefore ends up wasting his own time; the hospital also ends up wasting its own resources if the patient ends up at the wrong department. We have, hence, decided to build a system that will direct patients to the most appropriate department of the healthcare facility. Our system can be categorized as a text classification system and it is trained using supervised learning. Text Categorization is the task of assigning a natural language text a category from a predefined set of categories (supervised learning). Text categorization and text classification has been used interchangeably in the literature. There have been a surge of text categorization applications in recent years due to massive data generated online in social media and otherwise. Many applications of text categorization including spam filtering and e-mail routing [23] have been built. Generally text categorization is achieved by building a vocabulary of features from the given texts, training a classifier by using the features and then categorizing any new unseen text with the help of the built classifier. Vocabulary of features is built using standard *tfidf* function. *TFIDF* is short for Term Frequency Inverse Document Frequency. This is a method used for representing a set of documents as weighted n-dimensional vectors. Each dimension represents a feature in the vocabulary constructed from the set of documents. Term frequency $tf(t_i, d_j)$ is the number of times a term t_i appears in $document_j$ [24–27].

$tf(t_i, d_j)$ = frequency of ith term in the jth document

Inverse document frequency is calculated by

$$idf(t_i) = log(N) - log(n_{ti})$$

where N and n_{ti} denote the total number of documents and the number of documents that contain the term t_i respectively. Inverse document frequency for a $term_i$ will be close to zero if a term appears in many documents indicating the fact that common words appearing in several documents do not provide much information. The tfidf is

$$a_{ij} = tf(t_i, d_j).idf(t_i)$$

The unique terms that constitute documents form the vocabulary and are represented as a vector of fixed length, say, k. Each document is represented by a feature vector of length k where each term is a_{ij} score of corresponding term in the document. All machine learning algorithms take these feature vectors as input and build classifiers using a decision tree [28], ANN (Artificial Neural Network [29] among many other techniques. Multiple decision tree classifiers can be built and combined into a single classifier to get better performance by using gradient boosting [30] or random forest [31] techniques. The evaluation of a classifier is conducted experimentally by measuring its effectiveness, that is, its ability to take right classification decision. One popular measure used in machine learning literature for effectiveness is classification accuracy [25]. Classification accuracy is defined as follows.

$$Accuracy = \frac{\sum_{i=1}^{|C|}(True\_Positive_i + True\_Negative_i)}{\sum_{i=1}^{|C|}(True\_Positive_i + False\_Positive_i + True\_Negative_i + False\_Negative_i)}$$

where $|C|$ is the number of classes.

The above measure works well for situations where class tuples are more or less evenly distributed [32]. As we will see in Section 5, our data is distributed uniformly across classes and the above measure will serve the purpose.

To build a classifier for the proposed problem, we need labelled data. In medical domain, availability of data and that too in a standard format is an unsurmountable challenge as no standards yet exist. Work is being carried out to create a standard medical language to be used across applications and platforms [7]. We will discuss the data requirements in Sections 3 and 4. We present details of our system and its implementation in Sections 3 and 4. We present results in Section 5 followed by discussion in Section 6.

3. The Proposed System

The objective of our system is to suggest the appropriate medical department to the patient in a tertiary health care system. The block diagram of our proposed system is given in Figure 2. To suggest a medical department to a patient, we need to use medical history of the patient. Since our target is tertiary health care systems, patients have their medical records from primary and secondary health care systems with them. Our system uses all the medical records in possession of the patients. The variety of the data that our system uses makes it a Big Data system. There are 3 Vs associated with big data system- variety, volume and velocity. When patients walk into a Tertiary healthcare center, their documents can be scanned including:

- previous prescriptions from doctors
- drug-store bills of medicines purchased
- diagnostic reports
- medical images

The scanner would then digitize the documents. The digitized documents would be used by the pre-processing module for extracting the information presented in the documents and images. The images will then be processed to identify the organs and other relevant details available in the images [33–35]. Prescriptions, reports, bills etc. will be processed by Optical Character Recognition engines to convert them into searchable and editable text [36–38]. The extracted information will then be passed to the trained machine learning model for recommending a hospital department based on the input. The machine learning model performs a multi-class classification where each class represents a medical department of the hospital. The model uses the input provided and applies the classification algorithm to suggest a medical department.

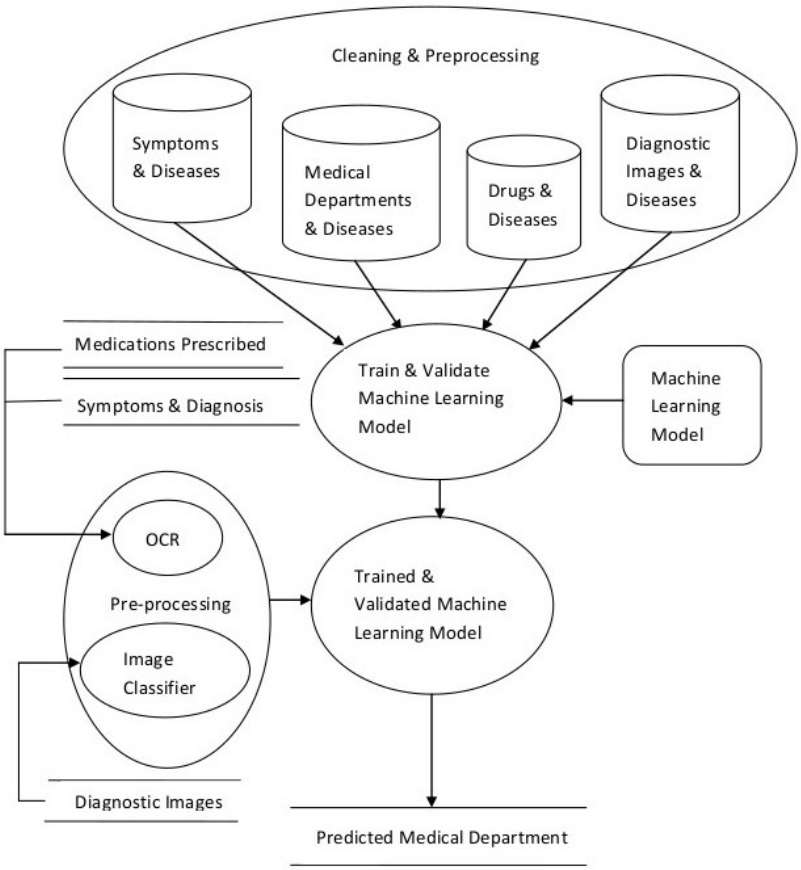


Figure 2. Block Diagram of the Complete System.

The machine learning model is trained using supervised learning. The training process involves the following steps.

- Data Cleaning and Preprocessing
- Scalable Model Building
- Model Validation and Selection
- Preprocessing and integration for updates

3.1. Data Cleaning and Preprocessing

We need a labeled dataset to train the system. Labeled data consists of names of the medical department (label) and diseases treated by the department. A disease is characterized by its symptoms. Symptoms alone are not sufficient for predicting the disease. Additional information in the form of prescriptions, diagnostic images and reports are also required. We want the system to learn disease

symptoms, diagnostic images and medicines prescribed (features) associated with each medical department (label).

If we have the labeled data, we can train the system to learn to suggest a medical department based on the features associated by using one of the machine learning model (discussed in Section 3.2). This process includes creating a profile for each disease based on its symptoms, medicines, and diagnostic reports and then mapping each disease profile to a department. This includes extracting the useful parts of the text, purging the stop-words from the text (Ullman and [39], converting the words into a common form by using stemming [40], feature extraction from the texts [41] and converting the data into a vector space model [42]. The challenging part of the problem is that apart from text data, there are also image data to deal with. According to data types, we have loosely three classes of extracted features—the symptoms or disease name, the medicines taken and processed images. Once we have the labeled data, we will build a model as described next.

3.2. Scalable Model Building

We want to use features to assign a patient to a particular hospital department; this is a multi-class classification problem. There are many machine learning models that can be used for multi-class classification problem. Deep Learning (a multi-layer neural network model trained using back-propagation algorithm), Support Vector Machine and Classification Tree are three popular techniques in machine learning for classification [43]. There are variations of these techniques to improve training time, reduce chances of over-fitting and improve classification accuracy. In our initial attempt, we trained all three models. We used Distributed Random Forest and Gradient Boosting [44] version of classification tree. There are many platforms and libraries available to train machine learning models without doing any programming. We used a platform available in the public domain [45,46]. By using the available platforms, the researcher can focus on building the model instead of programming.

After experimenting with these three models, we finally selected Distributed Random Forest as explained next.

3.3. Model Validation and Selection

Model building involves dividing the available labeled data into training and testing data to a proportion of 65:35. We then train the model using 65% data; let us call this trained model M_1. We then test the performance of M_1 on this very 65% data. Performance may be measured in terms of classification accuracy or log loss [47]. We tune hyperparameters (parameters supplied by the user; not learnt by the model) till we get acceptable classification performance on the training data. We then test performance of M_1 on 35% test data. If the classification performance on test data is comparable to classification performance on the training data, we accept M_1 as a potential model. There are many hyperparameters in each machine learning model that get tuned during this training. We build models using all three classification techniques to create a set of potential models M_2 and M_3. The model that gives best performance is selected as final model. The results of our model building exercise are presented in Section 5.

3.4. Processing and Integration for Updation

We start using the trained model for classifying unseen instance to classify it into one of the known classes. Our system will store the unseen instances along with labels assigned by our system for future use. These samples can be used for training the system for enhanced performance. The system should make use of the unseen samples to enhance the accuracy of the system over time as it sees and learns from more and more real use cases. We have not worked on this component at present and it remains as an agenda for the future.

4. System Implementation

Block diagram of the system that we have already implemented is shown in Figure 3.

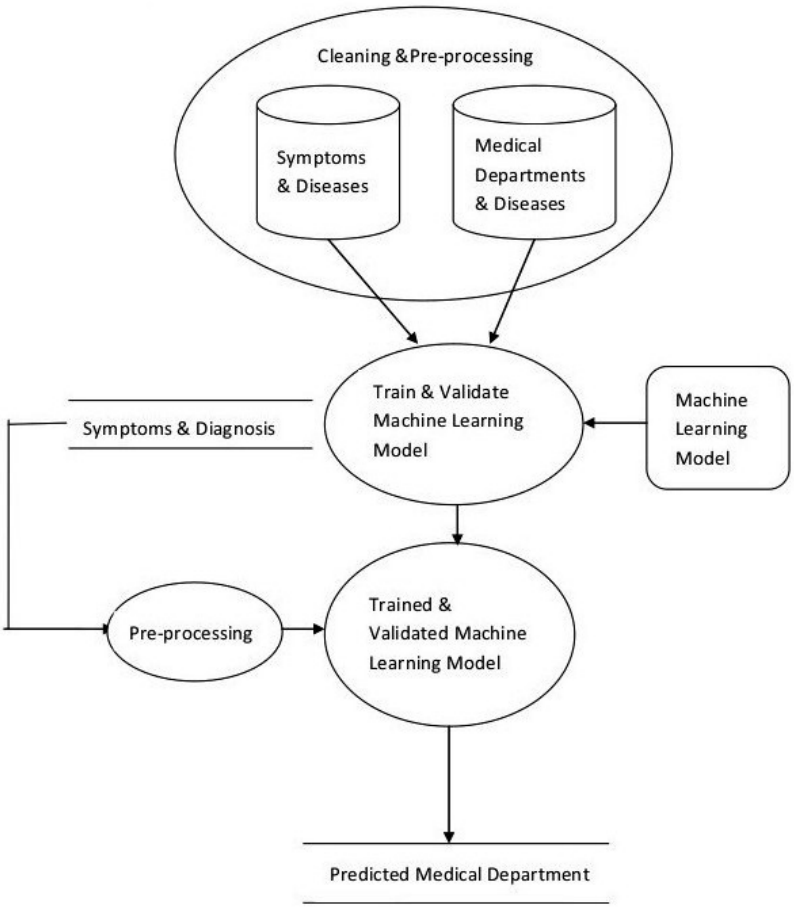


Figure 3. Block Diagram of the System Implemented.

The complete proposed system (shown in Figure 2) needs the following four types of labelled data for training.

1. Symptoms (features) and name of the Disease (label) Data: Disease_Description,
2. Diseases treated by a department (features) and name of the Departmental (label): Functional_Description,
3. Name of medicines (features) and Disease (label),
4. Diagnostic Images (features) and Diseases (label).

We have used the first two datasets in the present implementation whereas the last two datasets will be integrated in future.

The required form of Symptoms and Disease data is as follows.

$$< symptom_1, symptom_2, \ldots, symptom_n(Features) >< disease1(Label) >$$

Each department is represented as follows.

$$< disease_1, disease_2, \ldots, disease_m(Features) >< medical_aeptt(Label) >$$

We found labeled data for diseases (labels) along with symptoms (features) for 10,612 diseases [7]. Given below are descriptions of two diseases; namely *Intestinal strongyloidiasis* and *Hepatic Torque Teno Virus Infectious Disease*.

Intestinal strongyloidiasis (label) A strongyloidiasis that involves infection of intestine with Strongyloides stercoralis, which results in abdominal pain, diarrhea, ileus, massive gastrointestinal bleeding, severe malabsorption, and peritonitis (features).

Hepatic Torque Teno Virus Infectious Disease A viral infectious disease that results in infection located in liver, has material basis in Torque teno virus, which is transmitted by blood transfusion. human circovirus infectious disease Transfussion transmitted virus liver infection TT virus liver infection.

We found descriptions [48,49] of departments of hospitals as well. The following example shows description of a hospital department; namely *Gastroenterology*.

Gastroenterology The gastroenterology unit deals in digestive system including Mouth. Pharynx. Oesophagus. Stomach. Intestines. Small intestine. Duodenum. Jejunum. Ileum. Large intestine. Cecum. Colon. Rectum. Anus. Liver. Specialising in bowel-related medicine, upper and lower gastrointestinal diseases, pancreas, bile duct inflammatory bowel and swallowing problems.

Diseases *Intestinal strongyloidiasis* and *Hepatic Torque Teno Virus Infectious Disease* both are treated by *Gastroenterology* department. It is obvious from this example that it is not a simple one to one mapping between a disease and a hospital department. We worked with two doctors and two nurses to help us associate 10,612 diseases with one of the twenty medical departments (shown in Table 1). While working with the doctors, we realized that if map a disease to the department and do not retain symptoms then we are relying on the existing diagnosis. The doctors, we were working with, suggested that we should associate disease along with its symptoms to a department. Based on their advise, the required dataset that has the following form.

$department_1$: $disease_1, symptom_1, symptom_2, \ldots, symptom_m$
$department_1$: $disease_2, symptom_1, symptom_2, \ldots, symptom_l$
$department_1$: $disease_3, symptom_1, symptom_2, \ldots, symptom_n$

However, symptoms to diseases and symptoms to departments mappings are many-to-many which renders a problem that can be solved using machine learning techniques. For instance, we can build a classifier that uses symptoms and diseases as features and deperatment as label. An unseen sample may have symptoms or disease or both and the classifier predicts the department. Hence, the system implementation includes the following phases:

1. Create a dataset that has disease information (possibly including their names, associated symptoms, types, synonyms etc.) and name of the concerned medical department.
2. Converting the above dataset into vectors.
3. Identify suitable machine learning models and train them.
4. Test the models and select the best performing model.

The datasets disease_description and functional_description contain information about 10,612 diseases and 20 healthcare departments (listed in Table 1) respectively. We removed stop words, performed stemming, used heuristics to handle synonymous, homonyms, etc. For instance, the pair of words electrocardiogram and cardiomyopathy are essentially the same whereas hypertension and hyperbola are totally unrelated. Our word dictionary consists of 1000 words. Table 2 lists top twenty words. We used python to implement the preprocessing phase of the system.

The dataset is converted into a vector space model using term-frequency-index and document-frequency technique (tf-idf). The labeled dataset presented as vectors have been used to train and test machine learning models. Our problem is essentially a multiclass classification task [43]. We have department names as our classes and the objective of our model is to learn a mapping between diseases along with their symptoms and departments. We split the datasets into two parts: 65% for training, 35% for testing. We trained three machine learning algorithms: Gradient Boosting Machine, Distributed Random Forest and Deep Learning. We implemented the system using an open-source big data analysis platform (H_2O [46]).

Table 1. List of medical departments in a hospital.

Serial No.	Department Name
1	Anesthetics
2	Breast Screening
3	Cardiology
4	Ear, nose and throat (ENT)
5	Elderly services department
6	Gastroenterology
7	General Surgery
8	Gynecology
9	Hematology
10	Neonatal Unit
11	Neurology
12	Nutrition and dietetics
13	Obstetrics and gynecology units
14	Oncology
15	Ophthalmology
16	Orthopedics
17	Physiotherapy
18	Renal Unit
19	Sexual Health
20	Urology

Table 2. Top twenty words that describe diseases.

Variable	relative_importance	scaled_importance	Percentage
prostate	11,215.2275	1.0	0.0486
female	9913.1689	0.8839	0.0429
dysfunction	9881.0342	0.8810	0.0428
cyst	8901.1807	0.7937	0.0386
metabolism	2098.9890	0.1872	0.0091
sexual	2009.0562	0.1791	0.0087
mitochondrial	1720.9409	0.1534	0.0075
mutation	1659.1718	0.1479	0.0072
gene	1541.5609	0.1375	0.0067
variation	1524.2786	0.1359	0.0066
cataract1	1416.9362	0.1263	0.0061
peripheral	1387.7133	0.1237	0.0060
autosomal	1293.4233	0.1153	0.0056
depletion	1286.6816	0.1147	0.0056
cataract	1286.4379	0.1147	0.0056
chromosome	1245.4380	0.1110	0.0054
arthropathy	1202.5996	0.1072	0.0052
pressure	1081.8126	0.0965	0.0047
characterized	1059.2102	0.0944	0.0046
autosomal1	1023.6373	0.0913	0.0044
cancer	993.5568	0.0886	0.0043
recessive	979.1630	0.0873	0.0042
region	959.4741	0.0856	0.0042
compound	939.4285	0.0838	0.0041
deafness	922.4604	0.0823	0.004

We chose the model with the lowest misclassification rate as our final model. The results from our model building, validation and selection phase have been discussed in the next section.

5. Results

There are many machine learning models that one can use for classification as explained in Section 2. We have used three machine learning models, namely Distributed Random Forest, Gradient Boosting Machine and Deep Learning as mentioned in the last section. Based on the dataset, one model may perform better than others, One needs to experiment and pick the model that works the best. Each of these models require the user to specify hyperparameters and then during training, the model learns the parameters. Distributed Random Forest and Gradient Boosting Machine are both ensemble methods. Hyper parameters for both include the total number of trees to grow (ntrees), maximum tree depth (max_depth), stopping criteria, and the number of predictors randomly sampled as candidates for each split (mtries).

Deep Learning model needs user to specify activation function, architecture (number of hidden layers, number of neurons in each layer), stopping criteria for training, learning rate, dropout ratio, loss function etc. [50–53].

We specify hyperparameters and train the system with 65% randomly selected samples. There are various ways of reporting results [54], especially when data across classes are not balanced and importance associated with classes vary. In this paper, we have reported results using classification accuracy where every class and every error is given the same importance [55]. The data is also balanced across classes. Table 3 summarizes the results that we have obtained from these three models. As mentioned in the previous section, we have used 10,612 diseases mapped to 20 hospital departments. It is obvious from the results that, using just the descriptions of departments and diseases, the system made errors 10.18% of the time and was able to suggest correct department 89.82% of the time using Distributed Random Forest on the training data. However, when we run the system, it made errors 39.81% of the time and is able to suggest the correct department 70.19% of the time only. We were very puzzled with these results. We tried various combinations of hyper-parameters but the performance continued to be poor. We then went back to our data and closely examined it to discover that one department was over represented. Let us look at an example to understand the impact of over representation of a class. Let us say, there are two classes, namely A and B. There are 100 samples in total, 90 samples belong to class A and 10 samples belong to class B. If the classifier declares that every sample belongs to class A, it will be correct 90% of the time. However, if the data is balanced across classes, such a classifier will be correct only 50% of the time. If a class is over represented in the training data, the classifier will not be generalizable. We did class-balancing by deletion and re-trained the Distributed Random Forest which gave the best results (refer to Table 3).

The hyperparameters and corresponding errors for Distributed Random Forest are shown in Table 4. The performance improved to 92.7% on training data and quite close for test data as shown in Table 4.

Table 3. Initial Results: Training and Validation errors in percentage for five different settings for hyper parameters for three different models (Gradient Boosting Machine, Deep Learning and Distributed Random Forest).

Parameter Setting	GBM (Training)	GBM (Validation)	DL (Training)	DL (Validation)	DRF (Training)	DRF (Validation)
1	21.94	39.03	34.80	35.55	15.38	34.14
2	51.34	55.04	33.42	34.52	15.02	32.91
3	30.40	43.74	32.90	33.96	12.14	33.57
4	17.72	31.95	32.43	33.78	10.99	35.96
5	13.82	39.35	33.3	33.47	10.18	39.81

Table 4. Final Results: Training and Validation errors in percentage for five different settings for Distributed Random Forest.

Parameter Setting	Number of Trees	Max Depth of Tree	DRF (Training)	DRT (Validation)
1	128	5	5.23	8.4
2	128	8	6.12	7.3
3	128	10	8.32	7.6
4	500	10	6.91	7.9
5	1000	10	9.54	8.6

6. Discussion

We set out to create a system that will suggest a medical department to patients at a tertiary healthcare system based on their medical records. The medical records consist of diagnosis done at primary and secondary healthcare systems, medicines prescribed and diagnostic images. We have built a system that uses diagnosis records to suggest a medical department. It was a challenge to get the data required as there are no standard database available for describing diseases. We have achieved an accuracy of 92.7%. We need to incorporate drug and medicine information as well as diagnostics imaging data to train the system. The system should process the user's prescriptions and diagnostic reports as well. Once we incorporate everything, the performance of this system will improve.

We are now working on integrating image data. We have experimented with image datasets of eyes and lungs. We have been able to classify the organ in the image with near 100% accuracy. We also want to look at the prescriptions and run them through OCR (Optical Character Recognition system) to gain more information about the treatment that the patient has received. Another important thing to note here is that generally the doctors prescribe the same medicine from different brands from time to time. Our knowledge base should be able to map between the brand names and the constituent medicinal compounds. Ideally, our knowledge base should contain a list of medicinal compounds and the common diseases which they treat, and a list of medicine names from different brands for the same compound. In conclusion, we have implemented and tested parts of this system wherein a patient enters his symptoms and/or provisional diagnosis; the system suggests a department based on this user input. Our system suggests the correct department 92.7% of the time. Our system presently makes its suggestions using random forest algorithm that has been trained using two information repositories-symptoms and disease data, functional description of each medical department. It is our informed assumption that, once we have incorporated medicine information and diagnostics imaging data to train the system; and the complete medical history of the patient, performance of the system will improve significantly. The data collected can be put to further use to develop disease prediction models. The data can also help hospital management to evaluate load on different departments and plan accordingly.

Author Contributions: Conceptualization, V.B.; Methodology, A.P.; Writing—review & editing, R.G.-R.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Murray, C. Measuring the health-related Sustainable Development Goals in 188 countries: A baseline analysis from the Global Burden of Disease Study 2015. *Lancet* **2016**, *388*, 1813–1850.
2. World Bank Report. Available online: http://data.worldbank.org/indicator/SH.XPD.TOTL.ZS (accessed on 12 April 2016).

3. Naser, S.A.; Al-Dahdooh, R.; Mushtaha, A.; El-Naffar, M. Knowledge Management in ESMDA: Expert System for Medical Diagnostic Assistance. *ICGST-AIML J.* **2010**, *10*, 31–40.
4. Tenório, J.M.; Hummel, A.D.; Cohrs, F.M.; Sdepanian, V.L. Artificial intelligence techniques applied to the development of a decision—Support system for diagnosing celiac disease. *Int. J. Med. Inf.* **2011**, *80*, 793–802. [CrossRef] [PubMed]
5. Rahaman, S.; Hossain, M.S. A belief rule based clinical decision support system to assess suspicion of heart failure from signs, symptoms and risk factors. In Proceedings of the International Conference on Informatics, Electronics and Vision (ICIEV), Dhaka, Bangladesh, 17–18 May 2013.
6. Ibrahim, N.; Belal, N.; Badawy, O. Data Mining Model to Predict Fosamax Adverse Events. *Int. J. Comput. Inf. Technol.* **2014**, *3*, 934–941.
7. Northwestern University, Centre for Genetic Medicine; University of Maryland School of Medicine Institute for Genome Sciences. *Doid-Non-classified.obo, Format-Version: 1.2*; Data-Version: Released/2017-04-13. Available online: http://www.disease-ontology.org/ (accessed on 15 May 2017).
8. Ephzibah, E.P.; Sundarapandian, V. A Neuro Fuzzy Expert System for Heart Disease Diagnosis. *Comput. Sci. Eng.* **2012**, *2*, 17. [CrossRef]
9. Jain, V.; Raheja, S. Improving the Prediction Rate of Diabetes using Fuzzy Expert System. *J. Inf. Technol. Comput. Sci.* **2015**, *7*, 84–91. [CrossRef]
10. Caballero-Ruiz, E.; García-Sáez, G.; Rigla, M.; Villaplana, M.; Pons, B.; Hernando, M.E. A web-based clinical decision support system for gestational diabetes: Automatic diet prescription and detection of insulin needs. *Int. J. Med. Inform.* **2017**, *102*, 35–49. [CrossRef]
11. Goethe, J.W.; Bronzino, J.D. An expert system for monitoring psychiatric treatment. *IEEE Eng. Med. Biol.* **1995**, *15*, 776–780. [CrossRef]
12. Araújo, F.H.; Santana, A.M.; Neto, P.D.A.S. Using machine learning to support healthcare professionals in making preauthorisation decisions. *Int. J. Med. Inform.* **2016**, *94*, 1–7. [CrossRef]
13. Delias, P.; Doumpos, M.; Grigoroudis, E.; Manolitzas, P.; Matsatsinis, N. Supporting healthcare management decisions via robust clustering of event logs. *Knowl.-Based Syst.* **2015**, *84*, 203–213. [CrossRef]
14. Alonso-Amo, F.; Perez, A.G.; Gomez, G.L.; Montens, C. An Expert System for Homeopathic Glaucoma Treatment (SEHO). *Expert Syst. Appl.* **1995**, *8*, 89–99. [CrossRef]
15. McAndrew, P.D.; Potash, D.L.; Higgins, B.; Wayand, J.; Held, J. Expert System for Providing Interactive Assistance in Solving Problems Such as Health Care Management. U.S. Patent 5,517,405, 14 May 1996.
16. Davenport, T.H. *Big Data at Work: Dispelling the Myths, Uncovering the Opportunities*; Harvard Business Review Press: Cambridge, MA, USA, 2014.
17. Aruna Sri, P.S.G.; Anusha, M. Big Data Survey. *Indones. J. Electr. Eng. Inform. IJEEI* **2016**, 74–80. [CrossRef]
18. Schultz, T. Turning healthcare challenges into big data opportunities: A use-case review across the pharmaceutical development lifecycle. *Bull. Assoc. Inf. Sci. Technol.* **2013**. [CrossRef]
19. Jokonya, O. Towards a Big Data Framework for the Prevention and Control of HIV/AIDS, TB and Silicosis in the Mining Industry. *Procedia Technol.* **2014**, *16*, 1533–1541. [CrossRef]
20. Kumar, N.M.S.; Eswari, T.; Sampath, P.; Lavanya, S. Predictive Methodology for Diabetic Data Analysis in Big Data. *Procedia Comput. Sci.* **2015**, *50*, 203–208. [CrossRef]
21. McMullan, M. Patients using the Internet to obtain health information: How this affects the patient-health professional relationship. *Patient Educ. Couns.* **2006**, *63*, 24–28. [CrossRef] [PubMed]
22. Gallagher, M.; Pearson, P.; Drinkwater, C.; Guy, J. Managing patient demand: A qualitative study of appointment making in general practice. *Br. J. Gen. Pract.* **2001**, *51*, 280–285.
23. Busemann, S.; Schmeier, S.; Arens, R.G. Message classification in the call center. In Proceedings of the Sixth Conference on Applied Natural Language Processing, Seattle, WA, USA, 29 April–4 May 2000; pp. 158–165.
24. Salton, G.; Buckley, C. Term Weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
25. Sebastiani, F. Machine Learning in Automated Text Categorization. *Comput. Surv.* **2002**, *34*, 1–47. [CrossRef]
26. Jing, L.; Huang, H.; Shi, H. Improved feature selection approach TFIDF in text mining. In Proceedings of the International Conference on Machine Learning and Cybernetics, Beijing, China, 4–5 November 2002; Volume 2, pp. 944–946.

27. Debole, F. Sebastiani, Supervised Term Weighting for Automated Text Categorization. In *Text Mining and Its Applications. Studies in Fuzziness and Soft Computing*; Sirmakessis, S., Ed.; Springer: Berlin/Heidelberg, Germany, 2004; Volume 138, pp. 81–97.
28. Safavian, S.R.; Landgrebe, D. A survey of decision tree classifier methodology. *IEEE Trans. Syst. Man Cybern.* **1991**, *21*, 660–674. [CrossRef]
29. Hopfield, J.J. Artificial neural networks. *IEEE Circuits Devices Mag.* **1988**, *4*, 3–10. [CrossRef]
30. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
31. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
32. Gopal, M. *Applied Machine Learning*; Mc Graw Hill: New York, NY, USA, 2018; pp. 61–62.
33. Filipovych, R.; Davatzikos, C. Semi-supervised pattern classification of medical images: Application to mild cognitive impairment (MCI). *NeuroImage* **2001**, *55*, 1109–1119. [CrossRef] [PubMed]
34. Kucheryavski, S. Using hard and soft models for classification of medical images. *Chemom. Intell. Lab. Syst.* **2007**, *88*, 100–106. [CrossRef]
35. Antonie, L.; Zaiane, O.R.; Alexadru, C. Application of Data Mining Techniques for Medical Image Classification. In Proceedings of the Second International Conference on Multimedia Data Mining in Conjunction with ACM SIGIKDD Conference, San Francisco, CA, USA, 26 August 2001; pp. 94–101.
36. Bansal, V.; Sinha, R.M.K. Integrating knowledge sources in Devanagari text recognition system. *IEEE Trans. Syst. Man Cybern.-Part A Syst. Hum.* **2000**, *30*, 500–505. [CrossRef]
37. Ciregan, D.; Meier, U.; Schmidhuber, J. Multi-column deep neural networks for image classification. In Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, USA, 16–21 June 2012; pp. 3642–3649.
38. Patel, D.K.; Som, T.; Yadav, S.K.; Singh, M.K. Handwritten Character Recognition Using Multiresolution Technique and Euclidean Distance Metric. *J. Signal Inf. Process.* **2012**, *3*, 208–214.
39. Ullman, J.; Rajaraman, A. Mining of Massive Datasets. 2014. Available online: http://infolab.stanford.edu/~ullman/mmds/book.pdf (accessed on 11 January 2019).
40. Lovins, J.B. *Development of a Stemming Algorithm, Mechanical Translation and Computational Linguistics*; 11(1 and 2); Defense Technical Information Center: Belvoir, VA, USA, 1968.
41. Guyon, I.; Elisseeff, A. An Introduction to Feature Extraction. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
42. Ripley, B.D. *Pattern Recognition and Neural Networks*; Cambridge University Press: Cambridge, UK, 1996.
43. Aly, M. Survey on 0 Classification Methods. *Neutral Netw.* **2005**, *19*, 1–9.
44. Click, C.; Malohlava, M.; Candel, A.; Roark, H.; Parmar, V. *Gradient Boosting Machine with H_2O*; H2O.ai, Inc.: Mountain View, CA, USA, 2015. Available online: https://h2o-release.s3.amazonaws.com/h2o/master/3157/docs-website/h2o-docs/booklets/GBM_Vignette.pdf (accessed on 11 January 2019).
45. Candel, A.; Parmar, V.; Ledell, E.; Arora, A. Deep Learning with H_2O. March 2015. Available online: http://h2o.ai/resources (accessed on 10 January 2019).
46. H_2O, (10 January 2016). Available online: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/drf.html (accessed on 11 January 2019).
47. Collier, A.B. Making Sense of Logarithmic Loss. 2015. Available online: https://datawookie.netlify.com/blog/2015/12/making-sense-of-logarithmic-loss/ (accessed on 11 January 2019).
48. Henderson, R. Available online: http://www.netdoctor.co.uk/health-services/nhs/a4502/a-to-z-of-hospital-departments/ (accessed on 11 January 2019).
49. Mayoclinic. Available online: http://www.mayoclinic.org/departments-centers/index (accessed on 10 January 2016).
50. Kalman, B.L.; Kwasny, S.C. Why tanh: Choosing a sigmoidal function. In Proceedings of the International Joint Conference on Neural Networks, Baltimore, MD, USA, 7–11 June 1992; Volume 4, pp. 578–581.
51. Hahnloser, R.; Sarpeshkar, R.; Mahowald, M.A.; Douglas, R.J.; Seung, H.S. Digital Selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature* **2000**, *405*, 947–951. [CrossRef]
52. Goodfellow, I.J.; Warde-Farley, D.; Mirza, M.; Courtville, A.; Bengio, Y. Maxout networks. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 1319–1327
53. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

54. Collins, G.S.; Reitsma, J.B.; Altmana, D.G.; Moons, K.G.M.; TRIPOD: A New Reporting Baseline for Developing and Interpreting Prediction Models. *Art. Ann. Internal Med.* **2015**, *162*, 73–74.
55. Pal, M. Random forest classifier for remote sensing classification. *Int. J. Remote Sens.* **2005**, *26*, 217–222. [CrossRef]