2018 IEEE International Congress on Big Data

Diagnosis Recommendation Using Machine Learning Scientific Workflows

Ishtiaq Ahmed, Shiyong Lu, Changxin Bai, Fahima Amin Bhuyan
Big Data Research Lab, Department of Computer Science
Wayne State University, Detroit, USA
$[ishtiaq; shiyong; changxin; fahima.amin]$@wayne.edu

***Abstract*—Diagnosis recommendation plays a significant role in healthcare, where a clinician infers an optimal diagnosis for a patient. This problem has a major impact on improving patients' quality of life. Existing machine learning techniques for solving this problem require many labeled instances, which are not readily available. To overcome this limitation, in this paper, we present a scientific workflow for representing a semi-supervised clustering based diagnosis recommendation model. In this approach, initial clusters are formed from a labeled dataset; then imposing certain relative threshold to a cluster, frequent patterns and their corresponding labels are obtained. Subsequently, unlabeled instances are labeled by assigning them to the most similar clusters. Finally, we form clusters on the generated new datasets and recommend the diagnosis label by applying a certain minimum threshold. To evaluate our model, we perform extensive experiments on the *i2b2* datasets and compared our proposed algorithms with the self-training and co-training methods. The experimental results show that our proposed algorithm outperforms the mentioned methods in most cases. The proposed workflow is implemented in the DATAVIEW system.**

***Index Terms*—DATAVIEW, Personalized Healthcare, Workflow Recommendation, Health Informatics, Semi-supervised Learning**

I. Introduction

The problem of recommending diagnosis from medical data is an essential problem with significant impact on patients' lives and economy. This task becomes more challenging and arduous when we want to recommend diagnosis from unstructured textual datasets. Since this kind of data contains vast amount of information in an unstructured format, it is non-trivial to extract all the information and build the recommendation model for the corresponding patients. Making a recommendation model from unstructured textual dataset raises several questions:

- *Unstructured.* In most cases, learning models use structured datasets. How do we train a model from an unstructured dataset?
- *Imbalance problem.* Manually labeling a large dataset is labor-intensive and expensive in the healthcare domain. How do we develop a model from a small number of labeled datasets and a large number of unlabeled datasets?
- *Noisiness.* Due to potential inaccurate entries by a human operator, datasets might be noisy. How do we extract features from such noisy raw textual datasets?

In real-world applications, medical documents contain various information including patients' family history, medication, diagnosis, environmental reports, and so on. Since recommending diagnosis plays a vital role in treatment, it is considered as one of the significant attributes. However, diagnosis reports are not available for all patients. Additionally, prescribing inaccurate diagnosis to the patient is prohibitive. Therefore, we want to develop a recommendation model to suggest proper diagnosis labels to corresponding patients according to their significance.

Recommending the optimal diagnosis label can save thousands of lives around the world. Furthermore, it can also reduce thousands of dollars for a patient for post-treatment assessment. In the United States, various diseases are the leading cause of death according to several statistics. The main reasons for premature deaths throughout the country are heart diseases, smoking, cancer, obesity, etc. Over 60,000 people die every year from heart diseases alone [1]. More than one-third (34.9% or 78.6 million) of the whole population of the US are obese [2]. One in every three adults has high blood pressure [2]. Similarly, one out of three American adults has hypertension [3]. This high blood pressure costs the nation a sum of 46 billion dollars each year. This total cost includes medication to treat high blood pressure, cost of health care service and missed days at work. Moreover, the expenditure of overall health care has been rapidly increasing, and loss of productivity exceeds 208.9 billion dollars [4]. Hence, identifying the right disease can prevent the total mortality death as well as financial hazards. Optimal diagnosis of disease is one of the critical ways of determining the illness.

Extracting information from clinical notes has been one of the expanding fields drawing interest from various research communities [5]. An enormous amount of patient's medical data is stored in the electronic health record (EHR) format. EHRs can be partitioned into two forms: structured and unstructured. Unstructured EHRs are also called *clinical notes*. However, unstructured EHRs enlist every information in raw textual format. It contains patients' medical history, diagnosis labels, medications, treatment plans, immunization dates, allergies, laboratory test results, etc. Usage of these records can potentially improve health care safety, effectiveness, and accessibility. However, the narrative text used by a physician in electronic health records is difficult to comprehend. As these

978-1-5386-7232-7/18/$31.00 ©2018 IEEE
DOI 10.1109/BigDataCongress.2018.00018

IEEE computer society

texts contain ambiguous abbreviations, telegraphic styles, inappropriate sentence formation; understanding and extraction process becomes very arduous. Thus, various kinds of natural language processing techniques have been introduced to deal with this unstructured data. To extract information from unstructured datasets, MedLee (Medical Language Extraction and Encoding system) [6], MetaMap, UIMA-based applications cTAKES, Kepler [7], TexTractor [8] were introduced to obtain unstructured information. However, these models use domain-specific terminology, which suffers from overall performance degradation.

We introduce a novel semi-supervised method to recommend diagnosis labels from medical text documents. Our method can be executed on unstructured data leveraging clustering and frequent pattern mining based approaches. Initially, features are generated by various regular expressions, phrase and dictionary based lookup. After finishing this preprocessing step, clustering is performed on the labeled dataset. Next, by imposing a minimum support on each cohort, frequent patterns are generated. These frequent itemsets will be considered as the corresponding label for each cluster. Therefore, unlabeled instances are labeled according to their closest clusters. Thus, with the help of labeled and newly labeled dataset, new clusters are formed, and by re-imposing certain minimum support, diagnosis is recommended. Unlike co-training and self-training models where features are dependent on each other, the performance of our proposed methodology does not rely on the feature dependency. Moreover, unlike co-training, labeling is only possible when the cluster is fully formed and avoid initial labeling errors. Besides, the feature values are extracted from the unstructured dataset (i2b2[1]) by performing the preprocessing steps. The contributions of this paper are summarized as follows:

- We propose a novel semi-supervised method to recommend diagnosis labels with the help of clustering and frequent pattern mining.
- The proposed method is evaluated using the i2b2 dataset, which shows a better performance than self-training and co-training in most cases throughout different labeled dataset ratios. The ($P@K$) (Precision at K) performance metric is used in the evaluation.
- We design a scientific workflow to automate the semi-supervised diagnosis recommendation process and implement it in the DATAVIEW system [9] to ensure computational reproducibility.

The rest of this paper is organized as follows. Section II discusses related work on supervised, unsupervised, and semi-supervised machine learning approaches. Section III presents a comprehensive description of the proposed methodology. Section IV reports our experimental results. Section V concludes the paper and points out some future research directions.

II. Related work

Supervised learning has been used extensively in last several decades for categorizing medical textual datasets. Various statistical and machine learning algorithms are used to classify medical patient records and digital libraries. Several supervised machine learning techniques [10], [11], [12], [13], [14], [15] have been employed to train models automatically in various domains. In electronic text data, numerous supervised methodologies such as [16] have also been proposed. After preprocessing the unstructured data, feature generation is essential since traditional machine learning algorithms require structured values. Additionally, some of these datasets are difficult to label manually. Labeling a training dataset is sometimes monotonous and time-consuming. Meanwhile, the performance of supervised learning suffers when the size of a training dataset is small. On the contrary, several unsupervised learning, especially clustering algorithms have also been developed in the past several decades. Multiple kinds of clustering algorithms [17] deal with unlabeled data. Though training these algorithms does not need any labeled dataset, its performance is not as competitive as supervised learning. Hence semi-supervised learning (SSL) attempts to combine the best merits of these two paradigms.

Semi-supervised learning(SSL) uses both labeled and unlabeled data to perform learning tasks since supervised learning based on labeled data whereas unsupervised learning deals with unlabeled. This approach uses a small number of the labelled dataset and relatively a massive amount of unlabeled dataset to train the model. Since supervised learning employs a labeled dataset, SSL can perform well with the help of easily obtained unlabeled dataset, and hence this strategy can be applied in various applications. There exist multiple kinds of semi-supervised methodologies [18], [19] to train machine learning systems. The classic examples of SSL are self-training and co-training [20]. Since co-training [21] works with two views and these views are typically considered as independent, this approach fails to perform correctly when the features are dependent on each other. On the other hand, in self-training [21], early mistakes could reinforce themselves and degrade the overall performance. Moreover, further information regarding convergence is not available on this strategy. In generative modeling, it is often difficult to verify the correctness of the model. Besides unlabeled dataset might be mislabeled if the generative model is wrong [21]. In graph-based SSL, the model fails to perform when some of the datasets are mislabeled. This procedure is flexible and disagrees with given labels occasionally. The performance of graph-based SSL correlates with the construction of the graph, and it is highly sensitive to graph structure and edge weights. Semi-supervised support vector machines [22] (SVM) assume that the decision boundary is placed in a low-density region concerning unlabeled data. Since SVM does not approximate the exact, globally optimal solution of the non-convex problem association, it is relatively not supported. Graph-based semi-supervised learning [23] suffers from fundamental limitations

[1] https://www.i2b2.org/NLP/DataSets/Main.php

on general local to global eigenvectors of the corresponding similarity matrix.

III. Our proposed methodology

In this section, we present the working principle of our proposed methodology on labeled and unlabeled records with the help of clustering and frequent pattern mining. This method consists of several steps, and each step is elaborated in the following subsections. Figure 1 depicts the flowchart of the proposed semi-supervised methodology used for the recommendation of medical text record.

The flowchart of the proposed approach is divided into three different components: A) *Preprocessing*, B) *Labeling Unlabeled Instances*, and C) *Diagnosis Recommendation*. All these components are closely coupled with each other and a step cannot start its execution until the previous step completes. Initially, unstructured plain text datasets are provided to the *Preprocessing* step and this phase consists of three subparts: 1) *Subsection selection*, 2) *Feature selection*, and 3) *Feature extraction*. The correct paragraph for extracting information is identified in the *Subsection selection* step. When this specific area is selected, features are carefully chosen in the *Feature selection* step. The corresponding values for each feature are extracted from the raw text dataset. We use various regular expressions, dictionary-based lookup, and phrase-based parsing to obtain values in the *Feature extraction* step. Afterwards, diagnosis labels are constructed by the same procedure. In the *Labeling Unlabeled Instances* step, we work with both labeled and unlabeled instances. This component handles the labeled and unlabeled datasets with the help of clustering and frequent pattern mining. The features collected in the previous step are used to train the clustering methods. After the completion of this step, we obtain both labeled and newly labeled instances to run clustering methods and frequent pattern mining approach under a certain relative minimum threshold to find frequent patterns. This 1-itemset will be considered for recommending diagnosis labels for each cluster, and this process is conducted in the *Diagnosis Recommendation* component.

A. Preprocessing

Since the i2b2 dataset is available in raw text format, the *Preprocessing* step plays a crucial role in the overall procedure. This component contains three parts. An example of text description is shown in Figure 2.

1) Subsection Selection: In original records, every distinct paragraph is divided into separate paragraphs followed by "*:*". This step separates corresponding values with a given header which is called *key*. Usually *key* can be discovered in the same line where "*:*" is found. Similarly, the next corresponding values are followed by "*:*" until the next *key* is found. This section plays a significant role in finding accurate information from certain cohort because looking over the admission section will not guarantee to obtain the discharge date from the hospital.

2) Feature Selection: Completion of the previous sub-step leads to select features from various domains. However, most of the information is not relevant to run the machine learning methods. We choose 27 features from this vast amount of information, and these elements are stored in a hash map according to their feature names. We used 27 different features to construct a vector space model. These features are a mixture of nominal, ordinal and ratio values. The complete list of features are as follows: age, sex, days in hospital, diabetes, glucose, Coronary Artery Disease (CAD), CAD event Myocardial Infarction (MI),CAD event cardiac arrest, CAD event chest pain, hyperlipidemia, hyperlipidemia cholesterol, obesity, Prolol, cancer, hypertension, ace inhibition, aspirin, diuretic, insulin, medication, marital status (single, married, divorced), heart rate, respiratory rate, blood pressure, drinking habit, smoking status, and diagnosis label.

3) Feature Extraction: Once we identify the features, the overall performance of the system depends on how accurately the corresponding values are extracted. Obtaining values from features is accomplished by various regular expressions, phrase and dictionary based lookup. As various kinds of string matching operations are needed, regular expressions significantly reduce the task because patterns, rather than a fixed structure, are used to retrieve their corresponding values. For example, in order to retrieve the age of a specific patient, we develop the following patterns: $[0-9]1,3-year(s?)-old$, $[0-9]1,3\ year(s?)\ old$, $[0-9]1,3\ y\ \ o$, $[0-9]1,3\ yr\ old$, $[0-9]1,3\ year(s?)-old$, $[0-9]1,3\ yo$.

Besides using regular expressions, we also exploit phrase-based lookup for string matching. When a string is matched by this process, we also consider 50 characters forward and backward to validate our findings. Going through these extra characters can discover negative words such as "does not" or "not" to nullify the effect of extracted value. Similarly, another string matching method called the "dictionary based lookup" is introduced where words are directly tested for finding exact matching of predefined words. By applying this method, the diagnosis labels are extracted from the "Diagnosis" subsection. Some example of diagnosis labels are as follows: *asthma*, *alcohol abuse*, *adenocarcinoma*, *benign mucinous cystadenoma*, *breast cancer*, *carcinoma*, etc. The total number of diagnosis labels are approximately 90. These labels are mainly extracted from the "Principle Diagnosis" subsection. After the preprocessing step, a feature vector space is created based on the extracted values. A sample vector space is shown in Table I, illustrating the internal structure of a labeled dataset.

B. Labeling Unlabeled Instances

In this step, the unlabeled dataset B is labeled with the help of labeled dataset A. First, clusters are formed based on the characteristics of the labeled dataset. Then the frequent 1-itemset of diagnosis labels are calculated by a given threshold α for cluster C_i. Afterwards, the unlabeled dataset is assigned to the specific group according to their characteristics and labeled their diagnosis items based on frequent itemsets. The overall procedure is outlined in Algorithm 1. Line 1 forms

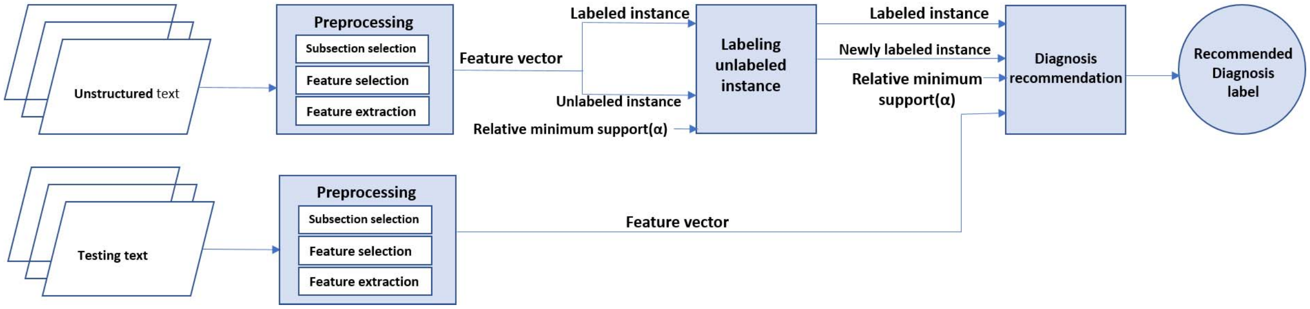
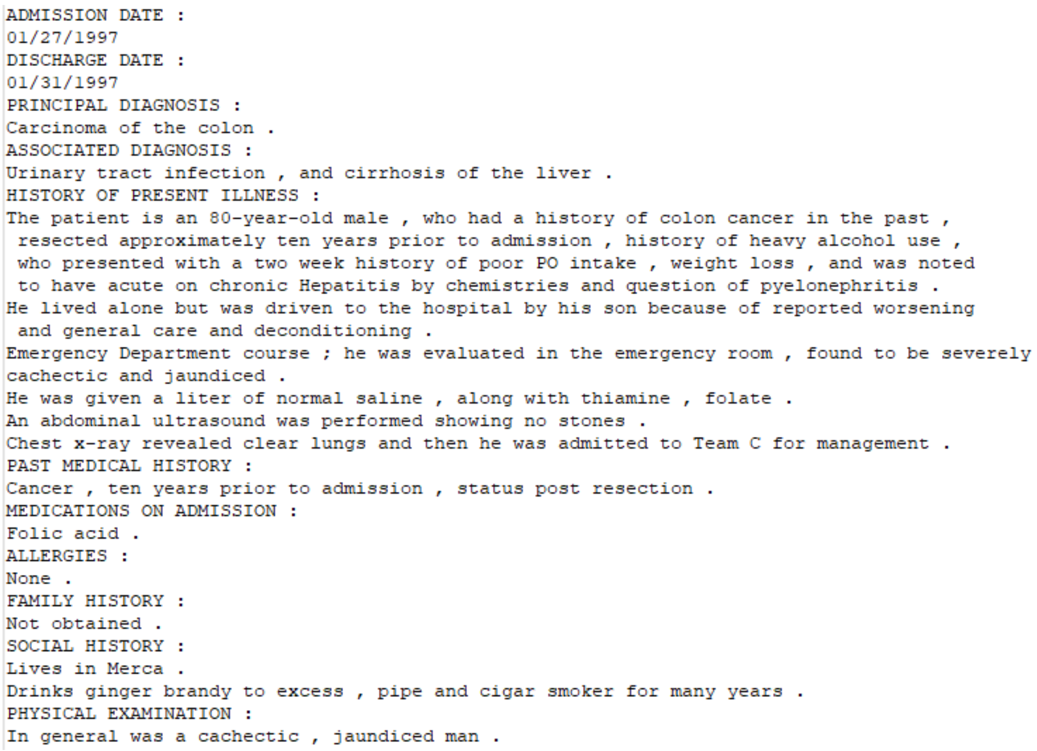


Fig. 1: Proposed flowchart of the overall workflow.



```
ADMISSION DATE :
01/27/1997
DISCHARGE DATE :
01/31/1997
PRINCIPAL DIAGNOSIS :
Carcinoma of the colon .
ASSOCIATED DIAGNOSIS :
Urinary tract infection , and cirrhosis of the liver .
HISTORY OF PRESENT ILLNESS :
The patient is an 80-year-old male , who had a history of colon cancer in the past ,
 resected approximately ten years prior to admission , history of heavy alcohol use ,
 who presented with a two week history of poor PO intake , weight loss , and was noted
 to have acute on chronic Hepatitis by chemistries and question of pyelonephritis .
He lived alone but was driven to the hospital by his son because of reported worsening
 and general care and deconditioning .
Emergency Department course ; he was evaluated in the emergency room , found to be severely
cachectic and jaundiced .
He was given a liter of normal saline , along with thiamine , folate .
An abdominal ultrasound was performed showing no stones .
Chest x-ray revealed clear lungs and then he was admitted to Team C for management .
PAST MEDICAL HISTORY :
Cancer , ten years prior to admission , status post resection .
MEDICATIONS ON ADMISSION :
Folic acid .
ALLERGIES :
None .
FAMILY HISTORY :
Not obtained .
SOCIAL HISTORY :
Lives in Merca .
Drinks ginger brandy to excess , pipe and cigar smoker for many years .
PHYSICAL EXAMINATION :
In general was a cachectic , jaundiced man .
```

Fig. 2: A representative snippet of raw text description of a patient.

TABLE I: A list of labeled instances.

Id	Age	Heart Rate	Is Obese	Smoker	Glucose	....	Diagnosis Label
1	60	104	No	No	150	...	Alcoholic cirrhosis, Liver disease, Liver cancer
2	45	80	Yes	Yes	300	...	Respirator failure, Lung cancer
3	65	102	No	No	140	...	Alcoholic cirrhosis, Anxiety
4	50	75	Yes	Yes	250	...	Liver disease, Respirator failure, Liver cancer
5	58	105	No	No	160	...	Liver disease, Hypotension
6	52	85	Yes	Yes	280	...	Respirator failure, Lung cancer, Liver cancer
7	61	106	No	No	170	...	Liver cancer, Pancreatitis
8	53	84	Yes	Yes	270	....	Lung cancer, Trachea, kidney
9	62	104	No	Yes	155	...	Alcoholic cirrhosis, Liver disease, Liver cancer

clusters C from labeled datasets A. Line 2 collects all the labels from the instances. Line 3 - 4 finds the appropriate diagnosis labels from each of the instances. In the end, the unlabeled instances are given the diagnosis labels according to the closest cluster's diagnosis labels.

TABLE II: A list of unlabeled instance.

Id	Age	Heart Rate	Is Obese	Smoker	Glucose	....	Diagnosis Label
10	59	105	No	No	147	...	?
11	42	77	Yes	Yes	305	...	?
12	63	100	No	No	142	...	?
13	44	79	Yes	Yes	289	...	?
14	60	102	No	No	161	...	?
15	50	82	Yes	Yes	276	...	?

In order to illustrate Algorithm 1, let us consider that we have both labeled and unlabeled instances. Table I and Table II depict the structure of labeled and unlabeled instances, respectively. We can also observe that five features such as Age, Heartrate, Is Obese, Smoker, and Glucose are mentioned in Table I. Similarly, various diagnosis labels are included for

Algorithm 1: Labeling Unlabeled Instances.

```
Input  : Labeled dataset $A$, Unlabeled dataset $B$,
         Threshold $\alpha$
Output: Newly labeled dataset $B'$
$C \leftarrow$ a set of clusters of $A$ using a clustering algorithm.
Let $L$ be all the labels in $A$
foreach $C_i \in C$ do
    Labelset $(C_i) \leftarrow \emptyset$
    foreach Label $L_j \in L$ do
        if freq $(L_j, C_i) \geq \alpha \times |C_i|$ then
            add $L_j$ to Labelset($C_i$)
    end
end
$B' \leftarrow B$
foreach instance $I \in B'$ do
    assign I to the closest Cluster $C_i$
    Labelset $(I) \leftarrow$ Labelset $(C_i)$
end
```

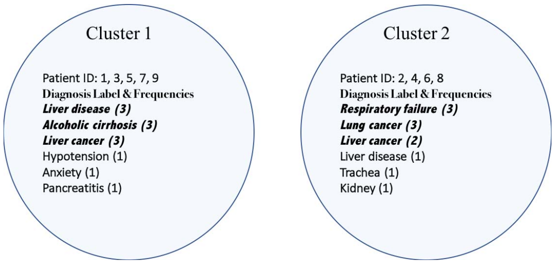


Fig. 3: A set of cluster formulation with frequent 1-itemset diagnosis labels.

every patient. Applying our clustering algorithm creates two groups such as $Cluster1$ and $Cluster2$. Afterwards some patients (ID: 1, 3, 5, 7) are enlisted for $Cluster1$ whereas others are categorized to $Cluster2$ and Figure 3 depicts the cluster formation. Then, applying relative minimum threshold 0.5 on diagnosis labels, we find optimal diagnosis labels in bold and italic font for both groups.

C. Diagnosis Recommendation Algorithm

At the beginning of this step, the labeled dataset A, newly labeled dataset B' from Algorithm 1 are merged. These two labeled datasets are used to form a final clustering model consisting of several clusters C according to the characteristics of the dataset. Then, a minimum relative support α is imposed to find out the 1-itemset frequent items of diagnosis label for each of the cluster. An unknown testing instance is categorized to its closest cluster and recommend frequent 1-itemset diagnosis labels. The detailed procedure is described in Algorithm 2.

Algorithm 2: Diagnosis Recommendation.

```
Input  : Labeled dataset $A$, Newly labeled dataset $B'$,
         Threshold $\alpha$, Test dataset $T$
Output: Predicted Label $T'$
$Tr \leftarrow A \cup B'$
$C \leftarrow$ a set of clusters from $Tr$ using a clustering
  algorithm.
Let $L$ be all the labels in $Tr$
foreach $C_i \in C$ do
    Labelset $(C_i) \leftarrow \emptyset$
    foreach Label $L_j \in L$ do
        if freq $(L_j, C_i) \geq \alpha \times |C_i|$ then
            add $L_j$ to Labelset($C_i$)
    end
end
$T' \leftarrow T$
foreach instance $I \in T'$ do
    Assign $I$ to the closest cluster $C_i$
    Labelset $(I) \leftarrow$ Labelset $(C_i)$
    Recommend Labelset $(I)$
end
```

IV. Case Study and Experimental Results

In this section, we conduct comprehensive experiments to evaluate the performance of our recommendation model along with the experimental platform. First, the experimental platform DATAVIEW [9] is discussed with a recommendation workflow followed by a case study. Then, an empirical evaluation is conducted by varying the minimum support value, different ratios of the labeled dataset, and the number of clusters for the model. Finally, a comparison is made with two different state-of-the-art semi-supervised algorithms: self-training and co-training.

A. The DATAVIEW System

DATAVIEW [9] is a popular big data scientific workflow system that supports the storage and sharing of datasets with integrated Dropbox support and the efficient execution of big data workflows in the clouds. Amazon EC2 is used as the computational clouds for running DATAVIEW workflows. Clinicians can compose different diagnosis recommendation workflows by the user-friendly interface provided by DATAVIEW. This interface assists users to reuse existing task components effectively and efficiently. This workflow design system provides an intuitive GUI for users to design, configure, and implement workflow executors [24] [25] [26]. It also supports the design and execution of data mining and machine learning workflows [27].

B. Case study

In Figure 4, the unstructured dataset is read from Dropbox content (linked with an user credential), and then the features with their corresponding values are extracted in the “Extraction” step. In this step, the features are selected and then various regular expressions, dictionary-based lookup approaches are used to extract the corresponding values. The second step,

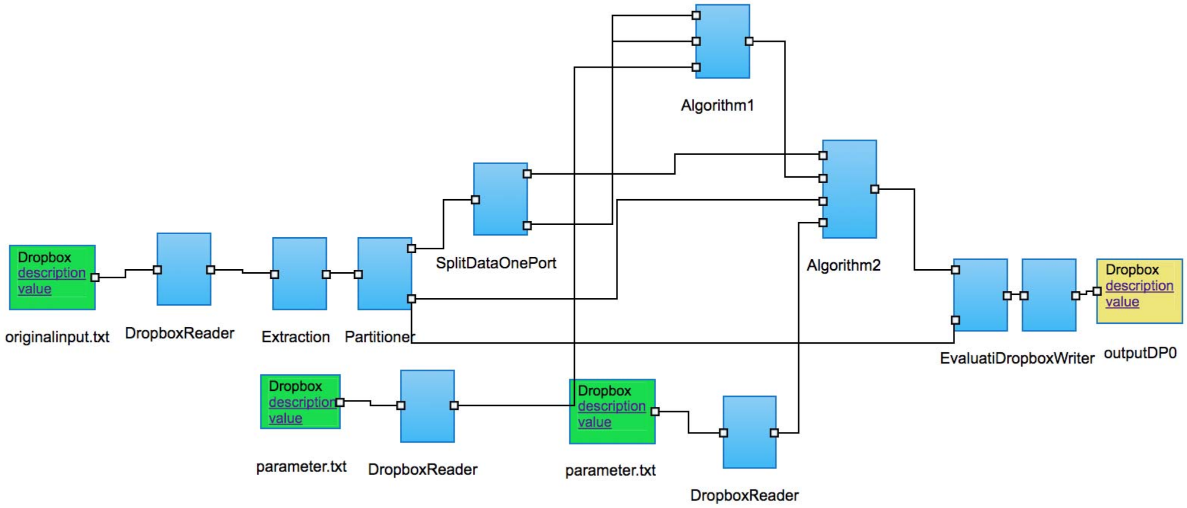


Fig. 4: A diagnosis recommendation scientific workflow implemented in DATAVIEW.

"Partitioner", splits the overall dataset into training and testing: 2/3 for the training and 1/3 for the testing. The training dataset is used for the input in the third step, "Split". This part splits the training dataset into two datasets for the "Algorithm1" step, and one of the parts is synthetically assumed as unlabeled. The fourth step is known as "Algorithm1," and it takes three inputs. Among these inputs, two of them are coming from the previous step, and the last one is a relative minimum threshold α. The first input of this step is used for constructing clusters, and α is used for finding 1-itemset frequent diagnosis labels from each of the clusters. Last, an unlabeled instance is categorized to a certain cluster and is assigned the diagnosis labels from that cluster. "Algorithm2" also takes four inputs from previous steps. Both labeled, and the newly labeled datasets are used for the final construction of the clustering model, and thus the unlabeled dataset is recommended according to their nearest group's diagnosis label. Finally, in the "Evaluation" step, a test dataset is assessed according to the model concerning top precision and recall values. To meet the golden standard of the evaluation procedure, the 10-fold cross-validation is used.

C. *Description of the dataset*

In order to conduct experiments, we use the publicly available *i2b2* 2006 dataset. This text corpus contains three different files, named *1A*, *1B*, and *1C* with various records of patients in XML format. Each of the datasets contains 889 unannotated discharge summaries. Each patient record contains 1500 words on average. We combine these three XML files into a single file maintaining the same format, and this format consists of two components: 1) *RECORD ID*, and 2) *TEXT*.

D. *Experimental settings*

Since DATAVIEW is implemented in Java, we choose Java as our core programming language to implement the proposed workflow tasks. As the original files are in XML format, we extract the information from the textual dataset using the Java document builder factory library. To evaluate various kinds of regular expressions and string matching, we use the regular expression and pattern library in the java regex matcher. The core proposed methods are implemented in core Java. Similarly, to develop various machine learning clustering algorithms, such as Expectation/Maximization (EM), K-means, and hierarchical clustering, we use the open source WEKA [28] library. Since we need to know the total number of clusters before running the K-means algorithm, we use NBCust [29], an open source implementation in R. We also use publicly available source code from github for self-training [2] and co-training [3].

E. *Performance evaluation*

As our task is to recommend diagnosis labels based on the characteristics of the individual patient, it is relatively suitable to use information retrieval measurements. Out of

[2] https://github.com/tmadl/semisup-learn
[3] https://github.com/jjrob13/sklearn_cotraining

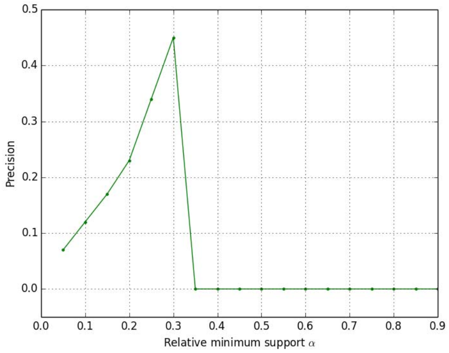


Fig. 5: Precision vs minimum support α.

various kind of measurement techniques, we use Precision at K recommendations ($P@K$) and Recall at K recommendations ($R@K$) where K is the top K recommendations among the retrieved diagnosis labels. Since our proposed methodology is based on various parameters such as the minimum support, the total number of clusters, we conduct multiple experiments to study the impact of these values.

F. Parameter sensitivity to minimum support (α)

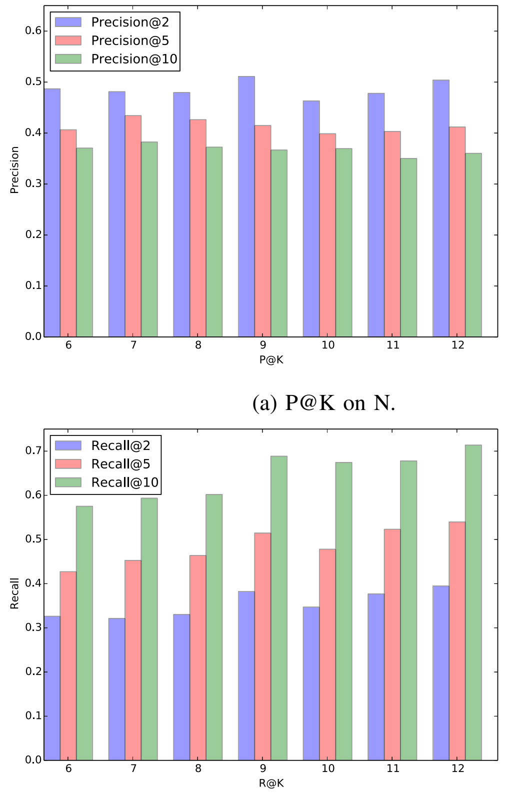
The α value plays a very crucial role in the overall performance, therefore selecting the right value of α is very important. As we know from the literature review of frequent pattern mining, when α is very low, we get too many frequent itemsets. As a result, our methodology fails to produce satisfactory precision. However, as we increase α, the overall precision result is increased first, and after a certain point, precision dramatically goes down to zero. The reason behind this is the unavailability of frequent itemsets since none of the items passes the threshold. The whole scene is depicted in Figure 5 where the best precision is reached when α is 0.30, and afterward the precision drastically becomes zero.

G. Parameter sensitivity to number of clusters (N)

Since the total number of clusters plays a crucial role in the effectiveness of our model, we experiment with a various number of clusters and compare the performance among them. This experiment indicates the robustness of our algorithm $R@K$ on a various number of clusters N. We conduct our experiment by varying N from *+3* to *-3* than the standard value of *N*. Since, NBClust produces the total number of clusters as 9, Figure 6 includes N from 6 to 12 to compare the obtained results. The overall P@K and R@K values are very stable with respect to different numbers of clusters. Hence, our proposed approach is relatively robust with respect to *N*.

H. Different clustering algorithm performance measurement

In this subsection, we discuss the performance comparison of our proposed methodology on various clustering algorithms in terms of $P@K$ and $R@K$, where K = 2, 5, 10. To perform this experiment, we use three different clustering algorithms



(a) P@K on N.

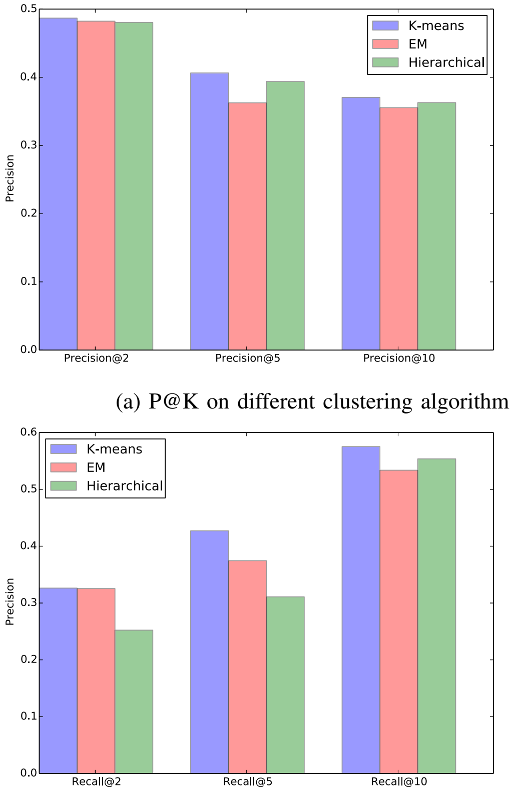
(b) R@K on N.

Fig. 6: P@K and R@K comparison on various N.

1) K-Means, 2) EM, and 3) Hierarchical clustering. We use the source code from WEKA to implement these algorithms. Figure 7 illustrates three different clustering algorithms performance in precision and recall scale. It is precisely depicted that $P@K$ and $R@K$ of K-Means is consistently better than EM and Hierarchical clustering for this dataset. We also notice that $P@K$ is decreased when K is increased; whereas $R@K$ is increased. While EM and Hierarchical clustering algorithms are both competitive, EM produces a slightly better result than the Hierarchical clustering algorithm.

I. Performance comparison with existing methods

In this subsection, we compare the performance of our proposed algorithm with that of self-training and co-training on different labeled and unlabeled datasets to verify the precision of our proposed algorithm. Since our task is to recommend diagnosis labels, $P@K$ is the best fit to justify the performance of these three methodologies, where K = 2, 5, 10. Out of the full dataset, we randomly select 2/3rd of the dataset for training and the rest of the 1/3rd dataset for testing purpose. Then out of the training dataset, we randomly select Ψ% dataset as labeled and remaining $(1-\Psi)$% as unlabeled U. We also use 10-folds cross-validation to evaluate P@K. Throughout this experiment, Ψ ranges from 15% to



(a) P@K on different clustering algorithms.

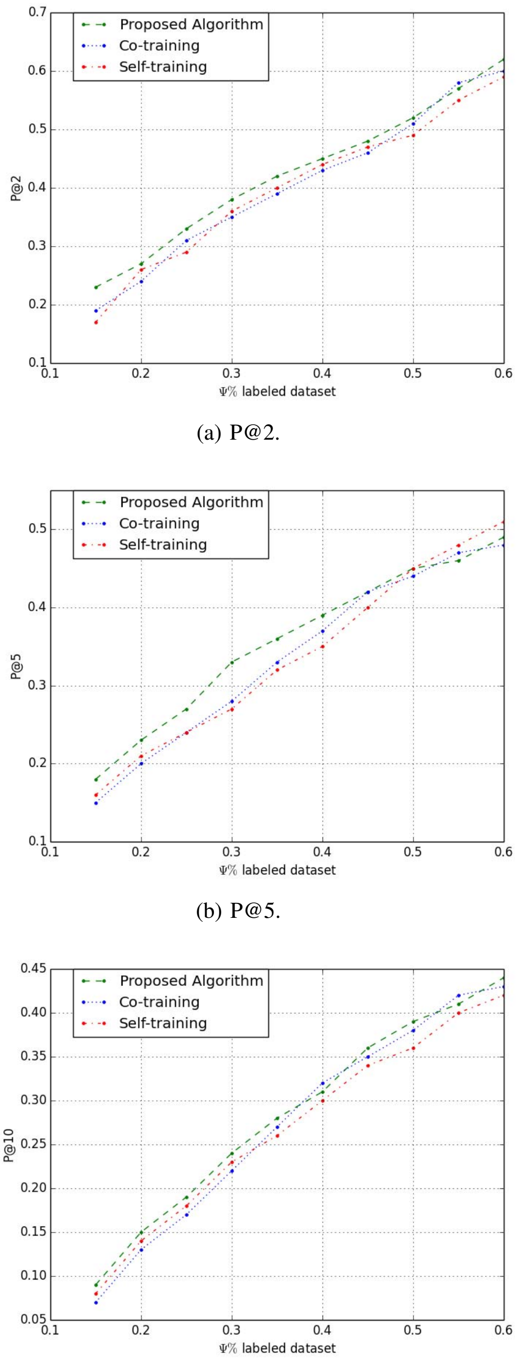
(b) R@K on different clustering algorithms.

Fig. 7: P@K and R@K comparison on various clustering algorithms.

60% having an equal interval of 5% to construct a different ratio of the labeled dataset.

Figure 8 depicts P@K comparison among our proposed algorithm, co-training, self-training in different Ψ settings and we consider $\alpha = 0.215$ for our algorithm to experiment. As expected, we can see the progression of P@K values when the labeled dataset ratio is increased for each of the methods. It is also observed that our proposed strategy produces consistently better results and outperforms the other two methods in most cases. Figure 8a shows the highest P@2 (0.62) when $\Psi = 0.6$ and the lowest 0.23 at $\Psi = 0.15$. Similarly, P@5 and P@10 results indicate that our algorithm is highly competitive with others.

We also observe that co-training performs better than self-training when Ψ is higher. However, both of these algorithms are very competitive throughout various ranges of Ψ. The learning rates for co-training and self-training are relatively slower than the proposed methodology, but both of these approaches outperform in some cases. Both of the models fail to produce a good result when Ψ is small. Because self-training uses model's output as a proxy for labels and since these labels are not sufficient with the small proportion of Ψ, it mimics unsupervised learning. Similarly, as co-training



(a) P@2.

(b) P@5.

(c) P@10.

Fig. 8: P@K comparison on different Ψ ratio.

construct models with two different views, it fails dramatically when any one of the views makes mistakes.

V. Conclusions and future work

In this paper, we presented a frequent itemsets and clustering based semi-supervised approach to recommending diagnosis labels with the help of labeled and unlabeled datasets. Initially, features are extracted from unstructured plain text data by regular expressions, dictionary and phrase based lookup. After generating clusters by labeled dataset, diagnosis labels are attached by frequent itemsets imposing certain minimum support to different clusters. Then, the unlabeled dataset is obtained with most similar cluster and labeled with the corresponding diagnosis labels. Finally, with the help of previously labeled dataset along with newly labeled dataset, new clusters are formed, and various diagnosis labels are assigned. These labels are the recommendation of that cluster. The overall recommendation workflow is implemented in DATAVIEW, one of the leading big data workflow systems in the community. This research work has several future research directions. First, we like to explore other clustering methods and evaluate the impact of choosing a clustering method on the performance of the proposed model. Second, since we observe different minimum supports produce different results, choosing an appropriate minimum support for different corpora would be one possible parameter tuning opportunity. Finally, as medical data also comes with other unstructured data such as medical images, a multimodality dataset driven approach might be one promising direction to explore.

Acknowledgment

This work is supported by National Science Foundation under grants NSF OAC-1738929, OAC-1443069, and CNS-1747095.

References

[1] M. P. Heron, "Deaths: leading causes for 2011," 2015.

[2] C. L. Ogden, M. D. Carroll, B. K. Kit, and K. M. Flegal, "Prevalence of childhood and adult obesity in the united states," vol. 311, no. 8, pp. 806–814, 2014.

[3] Q. Gu, "Hypertension among adults in the united states: National health and nutrition examination survey, 2011–2012," 2013.

[4] P. A. Heidenreich, J. G. Trogdon, O. A. Khavjou, J. Butler, K. Dracup, M. D. Ezekowitz, E. A. Finkelstein, Y. Hong, S. C. Johnston, A. Khera *et al.*, "Forecasting the future of cardiovascular disease in the united states: a policy statement from the american heart association," *Circulation*, vol. 123, no. 8, pp. 933–944, 2011.

[5] S. M. Meystre, G. K. Savova, K. C. Kipper-Schuler, and J. F. Hurdle, "Extracting information from textual documents in the electronic health record: a review of recent research," *Yearb Med Inform*, vol. 35, no. 8, pp. 128–144, 2008.

[6] D. B. Aronow, F. Fangfang, and W. B. Croft, "Ad hoc classification of radiology reports," *Journal of the American Medical Informatics Association*, vol. 6, no. 5, pp. 393–411, 1999.

[7] M. H. Nguyen, D. Crawl, T. Masoumi, and I. Altintas, "Integrated machine learning in the kepler scientific workflow system," in *In Proc. of the International Conference on Machine Learning (ICML)*, 2016, pp. 2443–2448.

[8] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, and C. G. Chute, "Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications," *Journal of the American Medical Informatics Association*, vol. 17, no. 5, pp. 507–513, 2010.

[9] A. Kashlev and S. Lu, "A system architecture for running big data workflows in the cloud," in *In Proc. of IEEE International Conference on Services Computing (SCC)*, 2014, pp. 51–58.

[10] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. M. Mitchell, K. Nigam, and S. Slattery, "Learning to extract symbolic knowledge from the world wide web," in *In Proc. of the Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference, AAAI*, 1998, pp. 509–516.

[11] T. Joachims, "A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization," in *Proceedings of the Fourteenth International Conference on Machine Learning (ICML*, 1997, pp. 143–151.

[12] D. D. Lewis and K. A. Knowles, "Threading electronic mail: A preliminary study," *Information processing & management*, vol. 33, no. 2, pp. 209–217, 1997.

[13] G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen, "Fuzzy artmap: A neural network architecture for incremental supervised learning of analog multidimensional maps," *IEEE Transactions on neural networks*, vol. 3, no. 5, pp. 698–713, 1992.

[14] H. Chen, H. Zhao, J. Shen, R. Zhou, and Q. Zhou, "Supervised machine learning model for high dimensional gene data in colon cancer detection," in *In Proc. of IEEE International BigData Congress*, 2015, pp. 134–141.

[15] I. Ahmed, D. Guan, and T. C. Chung, "Sms classification based on naive bayes classifier and apriori algorithm frequent itemset," *International Journal of machine Learning and computing*, vol. 4, no. 2, p. 183, 2014.

[16] G. Hripcsak and D. J. Albers, "Next-generation phenotyping of electronic health records," *Journal of the American Medical Informatics Association*, vol. 20, no. 1, pp. 117–121, 2012.

[17] C. Ding and X. He, "K-means clustering via principal component analysis," in *In Proc. of the twenty-first international conference on Machine learning*, 2004, p. 29.

[18] I. Ahmed, R. Ali, D. Guan, Y. K. Lee, S. Lee, and T. Chung, "Semi-supervised learning using frequent itemset and ensemble learning for sms classification," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1065–1073, 2015.

[19] A. Subramanya, S. Petrov, and F. Pereira, "Efficient graph-based semi-supervised learning of structured tagging models," in *In Proc. of the Empirical Methods in Natural Language Processing*, 2010, pp. 167–176.

[20] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *In Proc. of the 33rd annual meeting on Association for Computational Linguistics*, 1995, pp. 189–196.

[21] X. Zhu, "Semi-supervised learning tutorial," in *In Proc. of the International Conference on Machine Learning (ICML)*, 2007, pp. 1–135.

[22] O. Chapelle, V. Sindhwani, and S. S. Keerthi, "Branch and bound for semi-supervised support vector machines," in *In Proc. of the Advances in neural information processing systems*, 2007, pp. 217–224.

[23] A. Subramanya and P. P. Talukdar, "Graph-based semi-supervised learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 8, no. 4, pp. 1–125, 2014.

[24] A. Mohan, M. Ebrahimi, S. Lu, and A. Kotov, "A nosql data model for scalable big data workflow execution," in *In Proc. of IEEE International BigData Congress*, 2016, pp. 52–59.

[25] A. Kashlev, S. Lu, and A. Mohan, "Big data workflows: A reference architecture and the dataview system," *Services Transactions on Big Data (STBD)*, vol. 4, no. 1, pp. 1–19, 2017.

[26] M. Ebrahimi, A. Mohan, S. Lu, and R. Reynolds, "Tps: A task placement strategy for big data workflows," in *In Proc. of IEEE International conference on Big Data*, 2015, pp. 523–530.

[27] F. Bhuyan, S. Lu, I. Ahmed, and J. Zhang, "Predicting efficacy of therapeutic services for autism spectrum disorder using scientific workflows," in *In Proc. of IEEE International conference on Big Data*, 2017, pp. 3847–3856.

[28] G. Holmes, A. Donkin, and I. H. Witten, "Weka: A machine learning workbench," in *In Proc. of Second Australian and New Zealand Conference on Intelligent Information Systems*, 1994, pp. 357–361.

[29] M. Charrad, N. Ghazzali, V. Boiteau, A. Niknafs, and M. M. Charrad, "Package 'nbclust'," *Journal of Statistical Software*, vol. 61, pp. 1–36, 2014.