Precision Based Rough Set Based Hybrid Recommender for Scalable Top-K Drugs

K.A.Vidhya
Department of Computer Science
Anna University
Chennai
avidhya06@gmail.com

T.V.Geetha
Department of Computer Science
Anna University
Chennai
tv_g@hotmail.com

Abstract— **The exponential growth of information on healthcare in current years, steered to the mining of unstructured and structured health data, putting information about drugs and their adverse effects (in relation to particular symptoms and diseases) into the hands of pharmacists, doctors and patients, information that can be applied to make appropriate recommendations. Current health recommendation systems opt either for a collaborative recommendation or content-based recommendation where the recall is higher, compared to the precision of the drugs recommended. A scalable hybrid recommendation system has been proposed which uses the rough set for pruning the decision tree to improve precision without comprising much on recall. This work aims to assist healthcare practitioners to recommend drugs to patients based on review ratings provided by similar patients with similar illnesses. Patients allergic to certain drugs can referto the drug toxicity level for medications recommended by the system using the same user-mined review ratings from health care blogs. Based on patients' personal symptoms, drugs are mined through a large knowledge base for extracting all the possible ones, including synonymous drugs, using linked open data. However, in content-based filtering, forming rules using decision trees for the complete linked open data from large, connected graph structures requires proper scalability using MapReduce framework. The decision tree is converted to a set of rules further pruned by the rough set pruning method, the rough set theory being a soft computing technique that deals with uncertainty in data. The resulting rules are further pruned using the rough set theory, and the precision improved by 99.4% more than state-of-the-art methods.**

Keywords- Rough set theory, Linked Open Data (LOD), Big Data, Recommender System

I. INTRODUCTION

The availability of online health resources, such as electronic health records and drug related features - like drug-drug interaction, drug-toxicity level, drug rating, adverse drug effects and information on the most frequently used drugs - calls for an advanced health care recommendation system. Adequate familiarity and acquaintance in the field of medicine and the ever-increasing growth of online purchases of (over-the-counter) drugs have led to increasing numbers of patients or persons related to the patients trying to buy drugs through e-purchase in pharmacies. One of the examples we knew is Drugstore [9]. However, there are issues in buying medication through e-shopping even though it is very easy and comfortable to buy. With respect to the date of expiry or the strength of the drugs in question, the issue of risk factor might be higher of purchasing non-effective medicine due to the unethical and exaggerated information on the website/s concerned. These factors have paved the way for the development of personalized healthcare systems using the recommender system.

The chief issues pertaining to medical datasets have to do with identifying the number of drugs available in the market; analyzing the exact dosage level for the selected drug, ascertaining the adverse effects of the selected drug and, finally, going through the molecular combination in the drug (drug discovery). The future of the semantic web infrastructure includes both linked open data [23] and big data where enormous amount of data are connected and linked by a Uniform resource identifier (URI). This work aims to present the general public, including practitioners and pharmaceutical companies, information on drug dosage, adverse drug effects, and drug side effects. Interaction with the system helps one 9on effectiveness and toxicity level. The ratio depends on the drug's composition, i.e., the set of molecules as well as contraindications. The ultimate function of the system is to provide a drug database, arrived at after years of study in the fields of medicine and pharmacology, which helps describe drug characteristics in plainly-written text easily understood by the average layman for its simplicity.

The proposed work fetches similar drugs and looks at their side effects in terms of toxicity levels so they can be identified and recommended to medical practitioners. For examining drug-drug interaction, drug toxicity, drug indication, and adverse drug effects, existing standard, linked open terabyte-sized datasets have been used alongside linked graphs, matched or aligned with four biomedical ontologies [4]. To address the issues above, a hybrid recommendation system has been proposed which advocates top-k related drugs from a user-rating matrix and content-based recommendation using various bio-medical ontologies and semi-structured data. The paper is organized as follows: Section 2 discusses the related work. Section 3 discusses in detail the pruning technique involved in the proposed system. In Section 4, the recommendation system to handle uncertainties in large datasets has been presented. In Section 5, implementation details are offered, followed by Section 6 that provides various experimental results and discusses the evaluation of the system. Finally, the conclusion and future work are provided in Section 7.

II. Ease of Use

The era of big data analytics was set in motion because of the continuing digitization of health records, together with the (EHR) Electronic Health Record provide novel ways and means to analyse many clinical problems and administrative issues. Clinical decision support has been a popular area of research where there is a need for big data technologies that address a significant aggregate of information to identify, to segregate, to gain knowledge, and foresee the outcomes or to suggest an alternate way for curing to the professionals and the patients . Predictive data mining process of personalised health care and the diagnostic result can influence personalised health care in real time for the effective treatment of patients.

Several recommendation methods have been implemented in healthcare applications to provide healthcare services. In [10], Duan et al. proposed a nurse care recommender system. Kim et, al., [11], in their work, developed a personalised health care service system to suggest item recommendation with reference to the context-aware model. Chen, et. Al., [12], in their study, proposed a diet recommendation system based on domain experts about certain chronic illness and recommended diet food chart for the user with the facts on their fitness. Although these studies illustrate the value of healthcare personalization, most suggest treatments tailored from just raw data, primarily pharmaceutical and commercial. Yin Zhang et al. [34] proposed a novel cloud-based medicine recommendation (COMER) which intends to find the appropriate medication from the online store. Public health records need big data analytic solutions that can mine web-based and data to predict outbreaks of flu: based on consumer searches, social content [1] and query activity. To provide personalized healthcare services, extensive data should be mined using machine learning techniques intended to develop useful healthcare applications. Using various methods, many recommendations that have come within reach are developed for sophisticated use.

The personalized recommendation technologies generally make use of similarities based on users or items recommended obviously there should be a collaborative environment to decide on the collective similarity measure. In COMER [34], collaborative filtering is used with the Tucker tensor decomposition algorithm to recommend medicines. Yin Zhang et al. proposed a cloud-based medical recommendation system which can suggest top-N related medicines to users, in accordance with their symptoms. Firstly, drugs are clustered into several groups in line with functional description information and recommendations made based on user-collaborative filtering. The results arrived are enriched by modelling and thereby representing the relationship of the user, symptom, and medicine with the help of tensor decomposition. A major drawback of the collaborative filtering is data sparsity, leading to problems with cold starts.

In context-aware recommender systems [16], the HOSVD [32] is applied to eliminate the difficulty of sparsity and exact recommendations that have very few available non-zero context based user selection alleviating problems with excessive sparsity are generated using the higher-order singular value decomposition (HOSVD) technique. At the end point, the process is checked and calculated each contextual influence coefficient that every single type of context factor that is influencing user selection and then built a new N-order tensor with the help of the weighted linearization process to suggest positive recommendations. Knowledge-based recommendation usually needs a foundation of knowledge in the specific field related to the items. During the process of data analysis transaction, the association rule based recommendation quiet often appears that helps to discover links to recommended objects by creating personalized recommendation. In a diabetes medication recommendation system [24], Chen et al. proposed domain ontology for diabetes that predicts the corresponding symptoms and efficiently chooses the most suitable drug from the given pool of drugs.

Content-based recommendation discerns user interest based on characteristics of the user's past behavior, and makes recommendations according to the degree the user's interest and the items to be predicted can be matched. In 2014, a web-based system [8] was proposed by Y.Chang et al. that not only recommends alternative drugs to the user but also avoids certain drugs with a high level of toxicity when combined with others. The authors use both semantic web and data mining technologies for feature generation from user inputs and drug datasets, followed by an analysis of interactions using rule-mining. In recently times, the use of the semantic web to solve the problem of recommendation is becoming increasingly popular [2]. It is, however, necessary to understand what linked open data is, in tandem with work related to the integration of linked data and linked data feature extraction [14]. The use of ontological approaches [26] in healthcare is an evergreen research area. As new drugs come in, and the purpose of a drug‘s use differs somewhat from its use earlier, brand new areas of work come into play for researchers. In semantic information integration with linked data mashup approaches [5], a detailed investigation of semantic mashup research and application approaches in information integration are given, and an application is presented as an illustration of the proposed method.

Though many strategies are being in existence, a comparison of various strategies [10] for generating proportional characteristics such as; numeric, binary, or nominal from linked open data had been presented in the year 2014 comprising binary element creation, count feature creation, relative count feature selection, and tf-idf feature selection. A novel approach in the use of recommender systems in the medical domain has been proposed in OpenSelfMed [26], a web application that keeps people better informed while treating the medical ailments which are not diagnosed by the physicians a.k.a over-the-counter drugs, i.e., self-medication.

III. Proposed System

A host of medical databases is available with a large mass of information for scholars, doctors and medical practitioners to explore. The explosion of data in such incremental ratios is of little help to people looking for quick and accurate references, so semantic-based knowledge discovery offering intelligent, and quick references are needed. Such knowledge-based, intelligent medical healthcare systems will help practitioners and lay users diagnose possible ailments and select an appropriate combination of drugs for the patient concerned. For data processing, with the help of a single data model, the operations are being carried out with the application logics for mashup developers. The RDF model is being carried out using the vocabulary that brings about a general perception between the domain experts, users and machines. It is usually considered that a linked data MashUp is comprised of various pieces of technology.

Terabyte-sized medical linked open data like Sider [5], drugbank [9], and dailymed are mashed up or integrated using a silk workbench. Standard biomedical ontologies like the FMA, NCIT and SNOMEDCT are used to extract similar drugs, drug-drug interaction, drug-allergy interaction using the toxicity level, and additional parameters contributing to the result. FMA, DOID, SNOMEDCT, NCIT, and UMLS are the various ontologies used. The number of rdfs is of size 19982 triples. The ontologies are given to the FalconAO tool to be matched, and the tool generates alignments and matchings and gives the output in the RDF format. The output of FalconAO tool has to be parsed in order to be used. The file in the RDF format is given to an XML-like parser which extracts classes and labels. Standard biomedical ontologies like the FMA, NCIT and SNOMEDCT are used to extract similar drugs, drug-drug interaction, drug-allergy interaction using the toxicity level, and additional parameters contributing to the result. For example, if the user inputs “heart attack,” the synonym “myocardial infarction” is retrieved from the linked open data.

Example: Heart attack → Myocardial Infarction.

QUERYING LINKED OPEN DATA

```
SELECT DISTINCT ?drug_uri ?drug ?drug-type ?name ?drug-interaction ?dosage
 FROM <http://linkedlifedata.com/resource/drugbank>
 WHERE {
   ?drug_uri a drugbank:drugs .
   ?drug_uri rdfs:label ?label .
   OPTIONAL { ?drug_uri drugbank:drug ?drug . }
   OPTIONAL { ?drug_uri drugbank:drug-type ?drug-type . }
   OPTIONAL { ?drug_uri drugbank:name ?name . }
   OPTIONAL { ?drug_uri drugbank:synonym?synonyms . }
   OPTIONAL { ?drug_uri drugbank:dosage ?dosage . }
   {{FILTER regex(?label, "Acetaminophen", "i")}
     UNION
    {FILTER regex(?brandName, "Acetaminophen", "i")}
   }
 }
```

A. *Content-Based Recommendation*

Drug dataset resources available on the linked open data cloud are utilized in designing a recommender system [10] to create an integrated drug database. The recommender system also uses content-based filtering with machine learning techniques to suggest drugs to patients based on their illnesses. The C4.5 algorithm is used to form rules for retrieving similar drugs as the data size is huge.

The MapReduce framework is used for drugs A-Z, splitting them into {A-G , J-M , N-R and S-Z} four mapper classes. A decision tree is formed in an individual mapper class and the rules converted into attributes and class tables for decision tree-pruning using a rough set theory, which is the mathematical tool for dealing with uncertainty. As the rules are converted into a decision tree, the left, and the right child of a node which does not contribute to the information system are pruned using the rough set theory concept. In the following sections, modules detailing content-based filtering and addressing the issue of linked open big data have been presented.

B. *MapReduce Framework*

The input data is segregated into suitably sized splits and the map method generally considers a sequence of key or sequence of value pairs. The value pairs or processed key are sent to a specific reducer in which some portion functions a little bit later, soon after the querying lod the features are sorted and shuffled. Now the Reduce process iterates all the way through the values which associate with exact keys and turns out either some more outputs or zero.

Querying linked open data and profiling records from biomedical datasets require a highly scalable framework. The disease, symptoms, drugs, drug-drug interaction, gene ID, PharmaKB and so on are chosen attributes directly queried using four mapper classes, and the building of the individual profile is done in the reducer phase with de-duplication. The testing dataset consists of a random sampling of records from the training dataset, and the system performance tested using the test records. The results of testing are summarized below. Based on the profile above, identifying a particular drug for a given symptom from the content requires processing of the entire attribute list and fetching the list of drugs.

C. *Decision Tree*

To construct a decision tree from linked open dataset profiles which have a very large list of attributes, and identifying the correct set of attributes for the requested query can be done using a threshold value. However, an optimal threshold value for a decision tree to branch is hard to decide. The decision tree is allowed to grow recursively succeeding to the algorithm, for pruning the tree soon after to produce an excessive consistent tree by abandoning a single or more sub-trees by substituting them by the leaves, or replacing the sub-tree with one of its most frequently used branches.

When considering for an attribute choice of selection calculation is a heuristics for choosing the splitting criterion for splitting the attributes, which effectively divides the data provided for the partition, ‘D’ from the class , labelled as training tuples into separate unique classes. The attribute with best score, is the splitting attribute for the given tuples. Partition ‘D’ is labelled that is framed out of the tree node with the assistance of the splitting creation. The segregation as branches developed for every output of the criteria, as a result, the partitions have been done on the tuples accordingly. The C4.5 utilise the gain ratio according to the attribute selection measure. Inspite of utilising the information gain as in the case of ID3, the information gain ratio goes away with a dilemma of attribute selection for many values. Let 'C' denote the number of classes, and p(I, n) the proportion of instances in 'I' that are assigned to n^{th} class. Therefore, the entropy of attribute 'S' is calculated as:

$$\mathrm{Entropy}(S) = -\sum_{n=1}^{c} p(I,n) * \log(p(I,n)) \quad (1)$$

$$\mathrm{Gain} = \mathrm{Entropy}(s) - \sum_{v \epsilon \mathrm{values}(T_s)} \frac{|T_{S,v}|}{|T_S|} * \mathrm{Entropy}(I_V) \quad (2)$$

where $T_S$values T is the set of values of I in T , T_S the subset of T induced by S , and $T_{S,V}$ the subset of T in which attribute I has a value of v. Therefore, the information gain ratio of attribute I is defined as:

$$\mathrm{GainRatio}(I,T) = \frac{\mathrm{Gain}(I,T)}{\mathrm{SplitInfo}(I,T)} \quad (3)$$

$$\mathrm{SplitInfo}(S,T) = -\sum_{v \epsilon \mathrm{values}(T_I)} \frac{|T_{I,v}|}{|T_I|} * \log \frac{|T_{I,v}|}{|T_I|} \quad (4)$$

Using this, the rough set (U,A,S,f) is used to prune such rules in order to further reduce the 'm' classes for a set of data in which number 'n,' with 'nc' being the class 'c' with the highest number of data. If it is forecasted that all upcoming examples will be in class c, the following equation is identified for predicting the expected error rate 'Em' where, 'm' is the number of classes for all data and E_k is the expected error.

$$E_m = (n - nc + m - 1) / (n + E_k) \quad (5)$$

The other way is transforming the final decision tree to a collection of decision rules, where one rule for each leaf in the tree can easily be rewritten, otherwise the actual set of rules raises with the leaf nodes of the tree. p_s / t_o ‰ t_o is the number of postivie instances handled by rule p_s. The following rule set apparently dont address the negative instances as fast as possible and generate special rules which can handle the noisy instances.

Information gain is given by

$$p_s (\log(p_s / t_o) - \log(P/T)) ‰ \quad (6)$$

P and T are the positive and total numbers before the new condition is added. Information gain emphasizes positive rather than negative instances.

D. Rough Set for Decision Tree Pruning

Decision tree rules are pruned using a rough set theory that applies the rule-generation algorithm for pruning the rules necessary. Thereafter, using the query string, all the retrieved drugs connect information across multiple linked open data and the tree structure is visualized using a number of rules that vary, based on the information in the knowledge base. The rules formed are pruned using the rough set minimum rule generation algorithm. Reduct set is represented as a minimal subset of the initial dataset. The concepts of positive region, feature dependency, and significance are used for the computation of reducts.

Using the above concepts from the literature, the reduct R_{min} is modelled as a minimal subset R which is the reduct set of the initial attribute set C such that for a given set of decision attributes *D*, $\gamma_R(D) = \gamma_C(D)$. From the literature [35], R is a minimal subset, if $\gamma_{R-\{a\}}(D) \neq \gamma_R(D)$ for all $a \in X$. A number of attributes can be removed from the subset without affecting the degree of dependency. Hence, a minimal subset by this definition may not be the global minimum. A given dataset may have many reduct sets, and the collection of all reducts is denoted by

$$R_{all} = \{X \mid X \subseteq C, \gamma_X(D) = \gamma_C(D)_{\gamma X-\{a\}}(D) \neq \gamma_X(D) \forall a \in X\} \quad (7)$$

The intersection of all sets in $\boldsymbol{R_{all}}$ is called the *core*, the elements of which are those features that cannot be eliminated without introducing more contradictions to the representation of the dataset [2]. For many tasks, a reduct of minimal cardinality is ideally searched for attempts to locate a single element of the reduct set $\boldsymbol{R_{min}} \subseteq \boldsymbol{R_{all}}$:

$$R_{\min} = \{X \mid X \in R_{all} \forall Y \in R_{all}, |X| \leq |Y|\} \quad (8)$$

The core set is retained, and the reduct set from the upper approximation can be applied for the pruning method, wherein the data in the boundary can be rejected or neglected.

E. Collaborative filtering

The recommender system uses a collaborative filtering technique to recommend top-N related medicines to patients, according to their symptoms. Collaborative filtering is done using a patient database created using data crawled from health and social circles.

RSDP- Rough Set-based Decision Tree Pruning Process

1. Query from the end user (User, disease, and symptom)
2. Form an alignment with ontologies and linked data (L1,L2,L3…………Ln).
   Input: UMLS, LOD dataset, and SNOMED
   Find OWL: same as terms {S1, S2, S3, S4, S5…} for the given disease
   Group all the synonyms and find the different drugs prescribed {D1, D2, D3, D4 and D5}.
3. The Mapper takes {M1.....M4} linked open data as input.
   a. Pre-processing of LOD data
   b. SPARQL query on Sesame reports. Retrieve features of selected drugs using labels.
   c. <Mapper and Shuffler> Rule determination
      Input: Given drugs and retrieved drugs
      Output: Drug with interaction rules
         Get interaction between every pair of drugs from the drug: interaction parameter.
         Eg:
         drug 1, drug 2 , effect 1 > toxic
         drug 1, drug 2, effect 2 > not toxic

drug 1, drug 3, effect 1 > toxic

d. <Reducer> Reduced pool
Input: Pool of drugs
Output: Final reduced drug pool
Compare interaction of the given drug to the set of interaction rules. Remove from HashMap drugs with toxic on RHS.
<HashMap> ={D1,D2,D3,,,,,Dn}
Compare side effects with a set of severe side effects and remove from the set. Compare with user input and remove the directly allergic and those already taking drugs.

4. A decision tree construction with the input above.
   a. Convert into attributes and table structure with the set of rules {R1,R2....................Rn}.
   b. Prune rules using rough set rule pruning algorithm where the rules are not significant (identified using entropy calculation). Use this as information gain criteria for the rule pruning method. p (log(p_s / t_o) – log(P/T))

By taking in to account of the inadequacies of the collaborative filtering, like expensive computing with cold starts, and data sparseness, another approach is considered where medication is suggested by modelling and representing the user-symptom-medicine using tensor matrix. Drugs are clustered using the K-means clustering to check the drug's molecular combination and the description of drug information, following which the user rating (previous history) is applied to rate drugs for a particular symptom. Given that collaborating filtering has been used to achieve this, every time a user inputs symptoms of an illness, the nearest drug matching is obtained as a top-k drugs, most suited to the user's needs. From the said drug list, the user score of the drug is retrieved, processed and saved as a user-drug-rating tensor matrix.

User drug rating matrix construction is done by creating a file with a user_id, drug_id, symptoms, rating and constructing a matrix using Hashmap. The similarity measure is calculated using the Pearson correlation coefficient taking the input as a user_id (existing user) or a symptom (new user) and calculating similarity, that is among the two variables X and Y there exists a linear correlation (dependence). The positive correlation is indicated as 1, the 0 indicates no correlation and –1 indicates negative correlation. Drugs are recommended using the compute weighted sum applying the similarity matrix, and top N drugs are returned as recommended. The list of recommended drugs, drug ids and symptoms are sent to the Sesame triple store, just as links for the given drug_ids, as well as drugs for given symptoms, are retrieved.

The clustering of drugs is done according to the efficacy of the patient's treatment. The K-means clustering algorithm is used here to perform drug clustering. The output of the algorithm is clusters of drugs, with symptoms as a tag of the clusters. Features extracted from LOD are converted to feature vectors. Features are symptoms the drug can cure with its efficacy. The features are extracted using the vector space model. Feature vectors and the number of clusters are given as inputs to the K-means clustering algorithm, used to obtain drug clusters based on features (symptoms).

F. *Tensor decomposition*

A tensor is an N-dimensional vector. For example, a 3rd-order tensor can represent three kinds of data, and each coordinate can represent one kind of data [16]. The matrix, represented by two-dimensional data, can be also comprehended as a two-dimensional tensor. When there is more than three-dimensional tensor data, it is termed a higher-order tensor. Through tensor decomposition, the values obtained in the three-dimensional tensor can have applications in the field of predictive and personalized recommendation. The hierarchical structure of the XML documents has to be exploited using the multidimensional representations, such as those offered by tensor objects which is much desired by the medical web pages available today. Instead of the standard singular vector decomposition method for term document matrix representation, a tensor decomposition matrix method has been used which represents the drug-symptom-allergy as a 3-dimensional tensor. The tensors are represented as the multi-dimensional array for the given problem. In this paper, tensor vectors are denoted by 'a.' Matrices (tensors of order two) are denoted by '**A.**' Higher-order tensors (order three or higher) are denoted by ***T***.

During the process, either the order or the rank of the tensor is determined by the count of indices required to select from the dimension of the array. The order 2 tensor can be depicted as $\boldsymbol{T}_{ij}$ where i and j are the dimension of the related vector space. A real p-th order tensor A$\epsilon \otimes_{i=1}^{p} R^{n_i}$ is the member of the tensor product of Euclidean spaces $R^{n_i}, i\epsilon[p]$. The general restriction to the case where $n_1 = n_2 = n_3 = \cdots \ldots. = n_p = n$ is simply written A$\epsilon \otimes^{p} R^{n}$ to denote its p-order tensor.

Collaborative Algorithm for TopK-Drug Selection

Input: Crawled medical documents Output: Top-KDrugs with features

1. Calculate the similarity calculation among users using the Pearson co-efficient

$$\mathrm{sim}(x,y) = \frac{\sum_{i\epsilon I}(S_{x,i} - \overline{S_x})(S_{y,i} - \overline{S_y})}{\sqrt{\sum_{i\epsilon I}(S_{x,i} - \overline{S_x})^2 * \sum_{i\epsilon I}(S_{y,i} - \overline{S_y})^2}}$$

where 'í' is the set of all drugs and $S_{x,i}$ and $S_{y,i}$ represent the rating 'i'of the drug from user x and user y respectively. $\overline{S_x}$ and $\overline{S_y}$ represent the average rating of user x and user y respectively.

2. Identify features using the VSM algorithm.

3. Model drug-user-rating matrix using tensor factorization as {d1-u1-r1,d1-u2-r3,……………...}. Given the users and a set of similar users U_n={u_1,u_2…u_n}, the preference of u for an unseen rating

of a drug can be predicted using the predicted degree of adverse effects, rating or preference from the user U_n. R is the predicted degree of interest and 'sim' the similarity between the users.

$$R(u,d_i) = \sum_{j=1}^{n} sim(u,u_j) R(u_j,d_i)$$

4. The drug with top ratings is clustered using the K-Means algorithm.

The first step is tensor modeling, which takes drug clusters with the symptom tag as input and produces the User-Drug-Rating 3D tensor as output. Using the weight for each drug to multiply the score of the drug given by the user, the weight is designed to perform tensor modeling. Thus, the rank-3 tensor matrix is constructed with the user, drug, and rating which reduces sparseness in the matrix. The next step is the actual tensor decomposition using the Tucker tensor decomposition algorithm, where certain elements change from zero to non-zero by higher order singular value decomposition (HOSVD).

The weights of the new elements obtained represent the predicted scores that the users give to the drugs, according to the rating. The module analyses interactions between drugs that can be recommended and drugs that have already been administered, and identifies interactions that cause adverse side effects, toxicity or allergies. These drugs are eliminated from the pool of drugs that can be recommended, with the input given by the patient also considered for their elimination. The first step is retrieving the base pool of drugs. All drugs related to user inputs are retrieved from the integrated drug database, along with the drugs that interact with them. From this, the interaction of each drug with other drugs from the pool is observed, the toxicity parameter retrieved and drugs that cause these interactions removed from the HashMap. Next, the side effect label is retrieved and drugs that cause severe allergic side effects are eliminated. Further, drugs that are already being taken by the patient are eliminated along with drugs that contain molecules the patient is allergic to. Consequently, we obtain a reduced set of drugs for the recommendation.

IV. Implementation Details

As the first step in implementation, ontologies are aligned for feature extraction. There are five different biomedical ontologies used to create new alignments and mappings, presented in Table 1. The final mapped ontologies RDF file is of size 19982 triples. The tool used is FalconAO tool which utilizes the LMO and GMO to align ontologies, satisfying the demands of the linguistic matching of ontologies (LMO). Here, the input is ontologies and the output matched entities, and the process carried out through string similarity (SS), edit distance of entity names, document similarity, and Tf-IDF F with cosine similarity for the LMO method. In the case of graph matching of ontologies (GMO), the input is matched entities and the output additional matched entities, and the process rather like structural mapping using built-in properties (OWL, RDFS, RDF).

A. *Linked Open Data MashUP*

The first step is the pre-processing stage, wherein the Sesame workbench is downloaded and dependencies configured. New repositories are created and RDF triple dumps of LOD datasets added. The next step is link generation, where actual links between datasets are generated and established. The SPARQL end points of datasets are added to the silk workbench, along with the RDF output file of the ontology mapping module. The source dataset and target dataset are specified in the link task, with the link specification language (LSL) being used to generate same-as links between datasets. Finally, the silk mapping files generated are converted to actual links between datasets using the PHP. Scripts are written to connect to a D2R server, retrieve mappings, match URIs and generate actual links. The updated URIs are also added to the Sesame triple store.

The final mappings done are sider-DBpedia, drugbank-DBpedia, sider-drugbank, diseasome-DBpedia, dailymed-DBpedia, sider-dailymed, diseasome-dailymed [17] and diseasome-drugbank. Drug data extraction creates a database of patients, symptoms and ratings of drugs for patients by crawling and extracting data from various health websites. This patient database is used to find similar users to perform collaborative filtering. The necessary websites are identified and their page structure analysed. These websites are specified as seed URLs from which all the links are extracted using the JSoup library.

Inessential links such as Contact Us, or Twitter, are stripped and the remaining links explored recursively using JSoup. Drug ratings, symptoms and diseases are obtained by parsing the HTML page and, finally, text processing done to eliminate irrelevant words and retain only the drug with its symptoms/diseases and ratings. Two major websites like walgreens.com and druglib.com are crawled to retrieve patients' drug ratings. The largest drug retailing chain in US is the Walgreen Company. Drugstore.com, a part of Walgreens, contains drugs with consumer ratings. 20,000 records have been retrieved. The most comphrehensive drug database is owned by druglib.com with relevance to specific drugs. 19,835 records have been retrieved. JSoup parser returns to extract information from the seed URL and the link. Tensor decomposition is implemented in Java and the sparse matrix constructed to outwit problems with the collaborative filtering's cold starts.

V. Experiments and Evaluation

For evaluating the performance of top-K drugs recommendation systems, the most popular metrics used is precision and recall. When the user ratings are considered, it has to be split up in to a training set and a test set, the train algorithm applied on where the user ratings are calculate, split them into a training set and a test set, train the algorithm on the training set, and thereafter predict the top N items from that particular user's test set. The precision is called the number of retrieved instances, while recall is the fraction of relevant instance that are retrieved. Metrics and accuracy are calculated as follows,

$$Precision = \frac{\sum_{i=1}^{M} \frac{User\ Rated\ drug_i \cap Lod\ drug_i}{N}}{M} \qquad (9)$$

where M is the number of users, recommended medicine$_i$ the predicted rating, and N the number of drugs recommended. The recall rate calculation function is shown below:

$$Recall = \frac{\sum_{i=1}^{M} \frac{user\ rated\ d_i \cap LOD\ drug_i}{H_i}}{M} \qquad (10)$$

where M is the number of users, hitmedicine i the predicted rating, N the number of drugs recommended, and Hi the real rating given by the user. The 10 most similar users are selected to calculate all the ratings of drugs not scored by the given user, following which the top-N drugs with the highest predicted score are recommended. In content-based filtering, the average score of the user is utilized with association rule mining to predict ratings for drugs similar to the ones already rated by the user. For different lengths of the recommendation list, namely, 1, 3, 5, 10, 15, 20, 25, and 30.

we get a comparison of the experimental results of collaborative filtering (CF), collaborative filtering (CF) with tensor decomposition (tensor), and content-based filtering (content) and combination of both content-based, as well as collaborative filtering techniques, improves the performance of the system in recommending relevant drugs, Figure 2 depicts the precision rate chart and the recall rate chart and the F1 measure chart. The experimental results make clear that the cold start problem in recommendation is solved using the tensor decomposition when compared with the normal collaborative system. For implementing the content-based filtering system, a decision tree using the C4.5 algorithm is initially constructed, given the infeasibility of constructing one for the entire linked open drug that has been aligned. Consequently, the map-reduce framework is used for parallelizing the work and the drug-symptom-disease kind of relation rules constructed using a distributed file system.

The construction of rules is done in each map task; likewise, the work is carried out in 2 clusters, based on the alphabetical order of the drugs. There are two master nodes and 10 data nodes, each of five, connected to a single name node. Since the default block size 4MB cannot support the decision tree for rule construction (attribute) and final decision attribute, the block size is changed to 64MB. The shuffle phase of the Map Task combines all decision values and sets of attributes and, finally, the reducer removes the pruning of redundant rules using the rough set theory. In the reduce phase, along with the removal of recurring rules, the removal of the reduct set is computed using the Quickreduct from the literature reviewed.

The construction of rules is done in each map task; likewise, the work is carried out in 2 clusters, based on the alphabetical order of the drugs. There are two master nodes and 10 data nodes, each of five, connected to a single name node. Since the default block size 4MB cannot support the decision tree for rule construction (attribute) and final decision attribute, the block size is changed to 64MB.

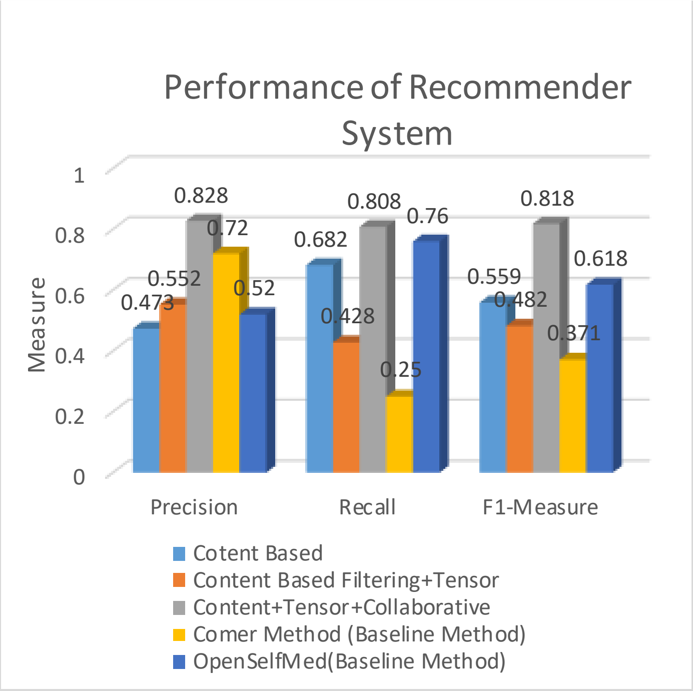

Fig1: Precision, Recall and F1-Measure Metrics for the Designed System Chart for Collaborative, Content-based and Collaborative with Tensor Decomposition

The shuffle phase of the Map Task combines all decision values and sets of attributes and, finally, the reducer removes the pruning of redundant rules using the rough set theory. In the reduce phase, along with the removal of recurring rules, the removal of the reduct set is computed using the Quickreduct from the literature reviewed.

The dataset for testing consists of a random sampling of records from the training dataset. The dataset comprises about a million evaluation records chosen randomly and excluded from the training dataset. The system's performance is tested using the test records, and the results summarized below. The drug dataset created from the linked open is of a size approximating 5 million, out of which the decision tree is formed using the procedure above.

Since the efficiency of the C4.5 is theoretically and empirically proved, the concern in the study is with the time efficiency of a parallel version of the C4.5 in a big data environment. In the linked open dataset, there are 120 attributes, the class label is diseases, and the decision yes or no for the set of drugs for the corresponding symptoms.

Fig 3 shows the execution time of different sizes of sample datasets where the x-axis denotes the numbers of instances in training data. Moreover, Fig 4 provides the speed-up performance of miscellaneous numbers of training instances as the number of nodes escalates, where speedup is a popular

TABLE 1: QUERYING LOD DRUG-DISEASE TARGET

Query	Chemical Component	Diseases Targeted	Precision	Recall	F-measure
Q1	Fumarate clobazam progabide	Epilepsy	0.782	0.892	0.833
Q2	Enalkiren	Hyperproteinemia	0.632	0.832	0.718
Q3	Estriol	Migraine	0.784	0.845	0.813
Q4	Amiodarone pindolol Arbutamine esmolol	Congestive heart failure.	0.872	0.863	0.867
Q4	Cyanocobalamin	Methylmalonic aciduria	0.765	0.856	0.808
Q5	Pindolol chloroquine	Asthma	0.876	0.876	0.876
Q6	Mianserin	Schizophrenia	0.932	0.756	0.835
Q7	Docetaxel Azathioprine Sunitinib	Leukemia	0.786	0.695	0.738
Q8	Fluvastatin	Attenuated cholesterol lowering	0.654	0.856	0.741
Q9	Nitroarginine Deserpidine	Alzheimer's Disease	0.759	0.945	0.842
Q10	Phenmetrazine	Brunner_Syndrome orthostatic intolerance	0.653	0.763	0.704
Q11	Chloroquine norgestimate	Migraine	0.765	0.856	0.808
Q12	Rofecoxib	Williams-Beuren Syndrome	0.876	0.776	0.823
Q13	Vasopressin	Diabetes_insipidus	0.932	0.856	0.892

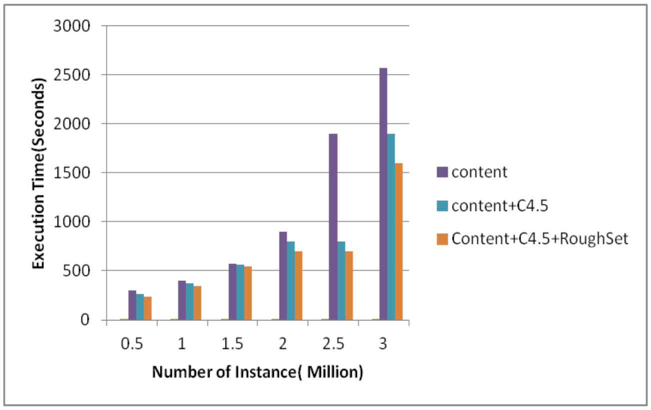

Fig 2: Performance of Map-Reduce on a Single Node

measurement of all the task tracker has been defined as the ratio of the execution time to that of specific numbers datanodes. From Figures 2 and 3, the observations are made: the larger the training dataset used, the more the cost of the execution time; the more nodes used, the shorter the execution time; and if enough nodes are leveraged, even if the size of the dataset is big, performance can be close to optimal. Thus the hadoop infrastructure maintains an implicit control over the execution time for all the number of nodes

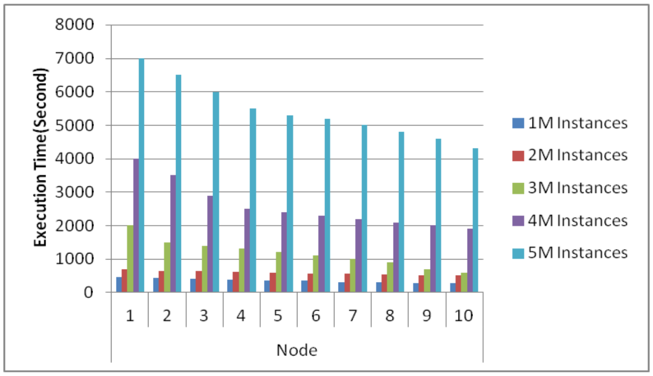

Fig 3: Performance of Map-Reduce on 10 Nodes

VI. [1]Conclusion and Future Work

A scalable hybrid recommendation system has been proposed for handling large medical datasets to meet the challenges of efficiently suggesting top k-drugs. The tensor recommendation overcomes the shortage of collaborative filtering when dealing with massive and sparse medical data. Furthermore, the system positively provides effective medication recommendation to satisfy the demands of versatile requests even though linked open data processing is time-consuming. Queries pertaining to drug-drug interaction, drug-allergy analysis, drug adverse effects, gene-diseases association and user ratings of a particular drug are obtained from the semantic web using the MapReduce framework and rough set decision pruning method. Patient-specific drugs tailored to treat a particular disease are suggested using the proposed system. Linked open data has been processed to increase the efficiency of selecting user-appropriate drugs, and also exploit its vastness, structure and detail. For future work, we are planning to develop a treatment recommendation system which suggest the latest treatment based on the patient review and the topological details which can be done by exploiting the clinical trials data.

Acknowledgment

I am greatly thankful for UGC in supporting this study, which was funded by DST PURSUE UGC, NEW DELHI, INDIA (9500/PD2/2014-2).

References

[1] P. Lops, M. de Gemmis, and G. Semeraro, “Content-based recommender systems: State-of-the-art and trends,” in Recommender Systems Handbook. Springer, pp. 73–105, 2011.

[2] Dipti D. Patil, V.M. Wadhai, and J.A. Gokhale, "Evaluation of Decision Tree Pruning Algorithms for Complexity and Classification Accuracy," International Journal of Computer Applications (0975 – 8887) Volume 11– No.2, December 2010.

[3] Prudhommeaux, E. and Seaborne, A. (2008) Sparql Query Language for RDF, W3C Recommendation. (2014) Sustainment Guide for Adverse Drug Events, Partnership for Patients Compaingn, Healthcare Government.

[4] Ceusters, W., Capolupo, M., Moor, D. and Devlies, J. (2008) Introducing Realist Ontology for the Representation of Adverse Events. Proceedings of the 2008 Conference on Formal Ontology in Information Systems (FOIS 2008), 237-250.

[5] Jentzsch, A. (2013) Sider. http://datahub.io/dataset/fu-berlin-sider/resource/e84dd6f3-f22e-4d4d-9ee8-e5b004eb654c

[6] A. B. Barrag´ans-Mart´ınez, E. Costa-Montenegro, J. C. Burguillo, M. Rey-L´opez, F. A. Mikic-Fonte, and A. Peleteiro, “A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition,” Information Sciences, vol. 180, no. 22, pp. 4290–4311, 2010.

[7] W. Carrer-Neto, M. L. Hern´andez-Alcaraz, R. Valencia-Garc´ıa, and F. Garc´ıa-S´anchez, “Social knowledge-based recommender system: Application to the movies domain,” Expert Systems with Applications, vol. 39, no. 12, pp. 10 990–11 000, 2012.

[8] Y. Zhang and E. Cheng, “An optimized method for selection of the initial centers of k-means clustering,” in Integrated Uncertainty in Knowledge Modelling and Decision Making. Springer, pp. 149–156, 2013.

[9] Jentzsch, A. (2013) Drugbank. http://datahub.io/dataset/fu-berlin Drugbank/resource/8fc23108-81d0-45f2-81ec-22ea41485f49

[10] L. Duan, W. N. Street, and E. Xu, “Healthcare information systems: Data mining methods in the creation of a clinical recommender system,” Enterprise Information Systems, vol. 5, no. 2, pp. 169–181, 2011.

[11] J. Kim, and K. Y. Chung, “Ontology-based healthcare context information model to implement ubiquitous environments,” Multimedia Tools and Applications, pp. 1–16, 2013.

[12] R.-C. Chen, Y.-D. Lin, C.-M. Tsai, and H. Jiang, “Constructing a Diet Recommendation System Based on Fuzzy Rules and Knapsack Method,” Recent Trends in Applied Artificial Intelligence, pp. 490–500, 2013.

[13] A. Jøsang, G. Guo, M. S. Pini, F. Santini, and Y. Xu, “Combining recommender and reputation systems to produce better online advice,” in Modeling Decisions for Artificial Intelligence. Springer, pp. 126–138, 2013.

[14] Y. S. Cho, S. C. Moon, S.-p. Jeong, I.-B. Oh, and K. H. Ryu, “Clustering method using item preference based on rfm for recommendation system in u-commerce,” in Ubiquitous Information Technologies and Applications. Springer, pp. 353–362, 2013.

[15] J. Liu, P. Musialski, P. Wonka, and J. Ye, “Tensor completion for estimating missing values in visual data,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 35, no. 1, pp. 208–220, 2013.

[16] Middleton, E., Shadbolt, N. and Shadbolt, D.C. (2004) Ontological User Profiling in Recommender Systems. Proceedings of ACM Transactions on Information Systems (TOIS), 22, 54-88.

[17] X. Cai, M. Bain, A. Krzywicki, W. Wobcke, Y. S. Kim, P. Compton, and A. Mahidadia, “Collaborative filtering for people-to-people recommendation in social networks,” in AI 2010: Advances in Artificial Intelligence. Springer, pp. 476–485, 2011.

[18] A.J.G. Gray, P. Groth, A. Loizou, S. Askjaer, C. Brenninkmeijer, K. Burger, C. Chichester, C.T. Evelo, C. Goble, L. Harland, S. Pettifer, M. Thompson, A. Waagmeester, and A.J. Williams, Applying linked data approaches to pharmacology: Architectural decisions and implementation, Semant. Web J. (2012).

[19] Adomavicius, G. and Tuzhilin, A. (2005) Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. Proceedings of IEEE Transactions on Knowledge and Data Engineering, 17, 734-749. http://dx.doi.org/10.1109/TKDE.2005.99

[20] Matthias Samwald, Anja Jentzsch, Christopher Bouton, Claus Stie Kallese, Egon Willighagen, Janos Hajagos, M. Scott Marshall, Eric Prudhommeaux, Oktie Hassenzadeh, Elgar Pichler, and Susie Stephens, Linked open drug data for pharmaceutical research and development, J. Cheminform. 3 (19) (2011).

[21] Peter Bloem and Gerben KD de Vries, “Machine learning on linked data: A position paper,” Linked Data for Knowledge Discovery, p. 69, 2014.

[22] R Celebi, O Gumus, and Y Aydin Son, “Use of open linked data in bioinformatics space: A case study,” In Health Informatics and Bioinformatics (HIBIT), 2013 8th International Symposium on, pp. 1–5. IEEE, 2013.

[23] Rung-Ching Chen, Yun-Hou Huang, Cho-Tsan Bau, and Shyi-Ming Chen, “A recommendation system based on domain ontology and swrl for anti-diabetic drugs selection,” Expert Systems with Applications, vol. 39, num. 4, pp. 3995–4006, 2012.

[24] Olivier Cure,´ “On the design of a self-medication web application built on linked open data,” Web Semantics: Science, Services and Agents on the World Wide Web, vol. 24, pp. 27–32, 2014.

[25] Yin Zhang, Long Wang, Long Hu, Xiaofei Wang, and Min Chen, “Comer: Cloud-based medicine recommendation,” In Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine), 2014 10th International Conference on, pp. 24–30. IEEE, 2014.

[26] K.A.Vidhya, T.V.Geetha, “Rough Set Theory For Document Clustering:A Review”, Journal of Intelligent and Fuzzy Systems vol. 32, no. 3, pp. 2165-2185, 2017, 10.3233/JIFS-162006.