An Approach of Hierarchical Concept Clustering on Medical Short Text Corpus

Wei Li
Key Laboratory of Medical Image Computing of Ministry of Education, Northeastern University, Shenyang, China

Dazhe Zhao and Jinzhu Yang
Key Laboratory of Medical Image Computing of Ministry of Education, Northeastern University, Shenyang, China

Longbing Cao
Advanced Analytics Institute, University of Technology, Sydney, Sydney, Australia

***Abstract*—Hierarchical clustering and conceptual clustering are two important types of clustering analysis methods. A variety of approaches have been proposed in previous works. However, seldom methods are designed to run on the medical short text database and construct a hierarchical concept taxonomy. This paper proposes a new clustering method of Hierarchical Concept Clustering on Medical Short Text corpus (HCCST), which presents a new solution on actionable disease taxonomy construction from the actual medical data. Our approach has three advantages. Firstly, HCCST takes a new similarity method which covers all the problems in medical short text distance computing. Secondly, an adaptive clustering method is proposed for synonymous disease names without predefining the size of clusters. Thirdly, this paper uses a mutual information based potential hierarchy concept pair recognition method which improves the subsumption method to create hierarchical disease taxonomy. The evaluation is conducted on Chinese medical disease name text data set and the result shows that HCCST achieves satisfactory performance.**

***Index Terms*—Hierarchical clustering, concept clustering, short text clustering, medical disease taxonomy.**

I. Introduction

A regular complete medical case usually contains the disease diagnoses that appear in the entire clinical process and these data are saved in Electronic Medical Record (EMR for short). The doctor gives the initial diagnosis and differential diagnosis once the patient is admitted in hospital. Additional diagnosis and confirmed diagnosis are made when more clinical evidences are captured during hospitalization. The medical record is archived with discharge diagnosis when patient discharged from hospital. Therefore, an EMR usually records as many as a dozen or even dozens of disease names.

Usually, disease naming and written formats always meet some rules. e.g. disease names may refer to the ethology, pathology, clinical manifestations (including signs and symptoms, stages, sub-type, gender, age, anxious chronic onset time) and anatomical location. Therefore, a multi-species and multi-level disease taxonomy is presented in medical domain.

Clinical text is the most abundant data source, but is also the most difficult data to be resolved for advanced analysis [2]. There are many domain or author specific idiosyncrasies, acronyms and abbreviations, as well as spelling and typing errors, so does the disease names in the EMR. Therefore, an accurate and unified disease taxonomy which is constructed automatically from the disease name texts in the EMR is the basis of the medical data cleaning and advanced analysis in practise.

The disease names in EMR data (Chinese EMR only in the research work) have many problems due to the reasons as the diverse and complex disease names, the lower intelligence of the electronic medical record systems and the doctor's writing habits.

- *Doctor's writing habits.* Different doctors in different hospitals will behave differently when writing the same disease names. For example, "轻度心力衰竭 (mild heart failure)" is written as "心力衰竭轻度 (heart failure mild)" which is different on the point of word position in expressing text. "急性心肌梗死 (acute myocardial infarction)" and "急性心梗 (acute myocardial infarction)" are same meaning that the latter is the abbreviation of the first one.
- *Complex disease hierarchy taxonomy.* Some doctors give the more detail information in disease names while others may do not, such as "血管炎 (vascular inflammation)" vs " 过敏性血管炎 (hypersensitivity vascular inflammation)".
- *Disease synonymous expression.* Disease may have many synonymous names. E.g. "婴儿腹泻 (infant diarrhoea)", " 婴幼儿腹泻 (infant diarrhoea)" and "小儿腹泻 (infantile diarrhoea)" are the same disease. Synonyms expressions are very common in medical domain. Hypertension has at least 36 kinds of expression methods [7].
- *Typing error.* In the actual medical records, the disease names usually have problems as typos, homophone, mixed by Chinese and English words, missing characters. Such as "脑梗塞 (cerebral infarction)" vs. " 脑梗赛 (cerebral infarction)"; "结肠癌 (colon

978-1-4799-2761-6/13/$31.00 ©2013 IEEE

IEEE

cancer)" vs. "结肠 cancer(colon cancer)".

- *Data preprocessing error.* Errors will occur when disease name texts are extracted automatically by system tools. E.g. the disease name of "术后诊断宫颈癌晚期 (Postoperative diagnosis Cervical cancer late)" should be removed the text of "术后诊断 (Postoperative diagnosis)" which is not part of disease name.

In order to standardize diseases classification, the World Health Organization(WHO) has proposed the disease taxonomy. The current international statistical classification and coding standards, ICD-10 (International Classification of Diseases, version 10), is drafted in Geneva in 1994 which is a multi-dimensional classification system. Medical insurance reimbursement, health statistics and reports are carried out based on this standard in many countries.

However, different countries and hospitals will make the substantial expansion according to the actual situations [21]. Moreover, there are also a lot of important implementation issues that due to the difference between the versions[22], coding accuracy [1]. The standard is only used in payment and financial audit rather than clinical care in United States [2]. In China, the usage of ICD-10 in clinical care is also in limited applications. A disease name may be matched to many categories in ICD-10 standard by computer because the disease text is too short. The computer is usually unable to specific the correct classification of the disease, so text matching based disease name standardization method cannot solve the problems described above. Another way to standardize the disease text is clustering method, including the hierarchical clustering [3][4]. and conceptual clustering methods. There are some previous methods on the conceptual clustering such as FCA [13], COBWEB [12], subsumption [7][8] etc.. However, these methods are not designed to run short text database and seldom algorithm can construct a hierarchical taxonomy without interaction.

In this paper, a *Hierarchical Concept Clustering for Short Text* (HCCST) algorithm is proposed, which can automatically standardize the disease name and construct the Hierarchical Disease Taxonomy (HDT) from disease names corpus. The HDT is represented by a directed acyclic graph in which a concept has zero or more subordinate concepts and may also has zero or more superior concepts. The HDT can be applied in multi-level association rule mining task and ontology support of medical records data retrieval, group statistics and other clinic data mining tasks.

The Hierarchical Disease Taxonomy created from actual medical data in this paper is a subset of ICD-10 proposed by WHO. However, the HDT is more actionable than ICD-10 in clinical workflow of medical information system. As the reason described above, the computer hardly knows how to find a standard disease name in real clinical text. Meanwhile, the HDT takes all the synonymous name and non-standard names into consideration so that HDT is more usefull in the tasks of the medical data cleaning and advanced analysis in practice.

The following sections in this article is organized as follows: Section II gives the definitions of research problem and methods. A new similarity measure method on short text is proposed in Section III. The two main steps of HCCST are described in Section IV and Section V. The evaluation is shown in Section VI and we make the conclusion in Section VII.

II. Problem Statement

In order to present the problem clearly, some notations are defined as table I and the definitions are interpreted as following.

TABLE I
Notations

Notation	Notes
D	The short text corpus
T	The term set
C	The concept set
R	The concept relationship
s_i	The ith short text
t_i	The ith term
c_i	The ith concept
w_i	The ith word in short text

Definition 1: (Central Concept) Given a synonymous short text set S, which is subset of the short text corpus D. A new text c, which is extracted from S and has the same semantic S, is called Central Concept of s.

Definition 2: (Hierarchy Concept Pair) Given concepts c_1 and c_2, if the concept c_1 has all the features of the concept c_2, and the concept c_2 also has some unique features that the concept c_1 does not have, then we call the concept c_2 is lower hierarchical concept of concept c_1. The concept of c_1 is called the "parent concept", denoted as: $c_2 \rightarrow c_1$. Concepts c_1 and c_2 make a Hierarchy Concept Pair. The relations between concepts are formed the concept relationship set, R .

Definition 3: (Hierarchical Concept Graph) Given a concept set C, for every concept pair $c_1 \rightarrow c_2$ where $c_1 \in C$ and $c_2 \in C$, then a directed acyclic graph is created from C, which is called Hierarchical Concept Graph. Any concept without parent concept in the hierarchical concept graph is assigned the "root" as its parent concept.

Therefore, the problem to be solved in this paper can be expressed as follows: given a short text collection D, a Hierarchical Concept Graph is created automatically by the central concept set C, and the concept relationships R, from D. Three sub-problems can be addressed in this discussion.

1) Central concept set, C, extraction with synonymous text in short text data, D.
2) The relationship set, R, identification between concepts in C.

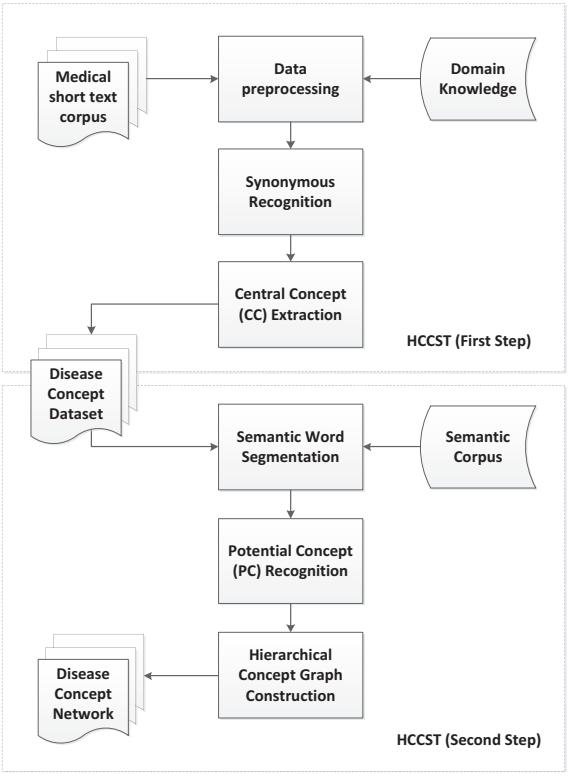

Fig. 1. The framework of HCCST algorithm

3) Hierarchical Concept Graph construction on C and R automatically.

The approach proposed in this paper is shown as Fig. 1. The HCCST algorithm is divided into two sub-processes: the central concept clustering procedure and the hierarchical concept clustering procedure.

- *Central Concept Clustering Procedure*: The main task of this stage is to find all the disease diagnosis text with synonymous. Then central concept extraction is executed on these synonymous disease text sets. The difficulty in this process is mainly on the synonymous text preprocessing and central concept extraction without human interaction. This process takes advantage of the domain characteristics of the medical text itself.
- *Hierarchical Concept Clustering Procedure*: The main task in this procedure is hierarchy concept pair identification which uses semantic features and subjective statistical methodology. The hierarchical concept graph is constructed based on the hierarchy concept pairs.

The HCCST algorithm is conducted on the short text corpus about the disease diagnosis in medical domain, and it can also be applied in other domains. The main innovative contributions of this paper are shown as following.

- This paper proposes an efficient set based short text similarity measure method.
- An adaptive clustering method is shown which can automatically determine the number of clusters.
- A mutual information based hierarchical concept clustering method is described.

III. Set Based Similarity Measure for short text

The disease name texts have different length that generally ranges from a few words to the dozens of characters, E.g. the disease of "先天性心脏病（肺动脉瓣狭窄左室发育不良三尖瓣关闭不全）(Congenital heart disease (pulmonary valve stenosis hyperplastic left heart tricuspid regurgitation))" and "肺炎 (Pneumonia)" have 27 and 2 characters. An efficient method is required in clustering procedure to measure the similarity between two short texts. Some previous works have been conducted. The most popular approach is the vector space model in text retrieval task. For example, TF-IDF value for every term in the text is computed [5], then a weighted matrix of term is got and similarity analysis method is applied on the matrix , e.g. LSA [9], cosine similarity [15], SVSM [10], KLD [11] and etc. However, such approaches usually encounter the sparse matrix computation problem, because the texts are very short other than the long documents texts in regular information retrieval application. There are also many other distance computing methods between two text strings in present research works, e.g. the Hamming distance [16], Levenshtein distance [17], the Jaccard index [18] etc.. The medical texts are usually expressed by abbreviation or disorder terms as the problems described in the first section. It is hard to get satisfied result in many cases. At the same time, the computing efficiency will become a problem when the dataset is very large, such as the Levenshtein distance.

This paper uses a Set Based Similarity(SBS) measure method which uses the set concept. The SBS measure is defined as follows.

Given two short texts s_1 and s_2, where $s_1 = [w_{11}, w_{12}, \cdots, w_{1p_1}]$, $s_2 = [w_{21}, w_{22}, \cdots, w_{2p_2}]$, then the similarity between s_1 and s_2, $d(s_1, s_2)$ is defined as equation (1).

$$d(s_1,s_2) = \begin{cases} 0 & \text{if } |set(s_1)| = 0 \\ & \text{or } |set(s_2)| = 0 \\ 1 & \text{if } set(s_1) \subseteq set(s_2) \\ & \text{or } set(s_2) \subseteq set(s_1) \\ \frac{|set(s_1) \cap set(s_2)|}{|set(s_1) \cup set(s_2)|} & \text{others} \end{cases} \quad (1)$$

where $set(s)$ standards for words set in s, $|set(s)|$ is the length of $set(s)$.

The SBS computing equation can be viewed as a special Jaccard index which represent the similarity between two collections. In SBS method, the word set of the short text is obtained and then the Jaccard index value is calculated. In this paper, We use the distance as the

similarity between two short texts. Table II shows the comparison of several similarity calculation methods.

TABLE II
THE COMPARISON OF THE DIFFERENT SIMILARITY COMPUTING METHODS

Pairs of the short Texts	Hamming	Levenshtein	Jaccard	SBS
“双眼糖网 (Both eyes and diabetic retinopathy)” vs “双眼糖尿病视网膜病变 (Both eyes and diabetic retinopathy)”	-	0.5 $(1-\frac{7}{14})$	0.29 $(\frac{4}{14})$	1.0
“双眼视网膜动脉硬化 (Binocular retinal arteriosclerosis)” vs “双眼动脉硬化性视网膜病变 (Eyes arteriosclerotic retinopathy)”	-	0.42 $(1-\frac{7}{12})$	0.75 $(\frac{9}{12})$	1.0
“双眼动脉硬化性视网膜病变 (Eyes arteriosclerotic retinopathy)” vs “双眼动脉硬化性视网膜变化 (Eyes arteriosclerotic retinal changes)”	0.83 $(1-\frac{2}{12})$	0.83 $(1-\frac{2}{12})$	0.85 $(\frac{11}{13})$	0.85

the SBS shows the best performance as shown in Table II. The Hamming distance requires strict condition that the length of the input text must be same. The computing complexity of the Levenshtein distance is high especially when the data set is too large. Jaccard Index is not compatible with the special condition as the first two examples in Table II. The SBS method overcomes these problems which is better than other methods.

IV. Central Concept Clustering

There are many synonymous texts for a disease name in the data set. Two problems should be solved that these short texts should be clustered together and an unified disease name is assigned to them. The two problems are described in this section in detail.

A. Adaptive Cluster Recognition

In general clustering analysis task, two objects are treated in a same cluster when the two objects are near enough. SBS similarity calculation method is used to determine whether the two short texts are synonymous. However, the number of clusters in the data set is unknown for us which is different with present clustering algorithms, such as the K-Means algorithm, which generally requires predetermined value of K. The Aggregate clustering algorithm also needs to determine an interception threshold.

In this paper, an adaptive cluster recognition method is proposed to divide the data object automatically which can determine the number of clusters. Given a cluster is represented as $c = (header, members)$ where $header$ is the cluster center and $members$ is the set of objects in the same cluster. The similarity between the two short texts s_1 and s_2 in the data set D is denoted as $d(s_1, s_2)$ using SBS. The clustering process, SplittingCluster, is defined as Algorithm 1.

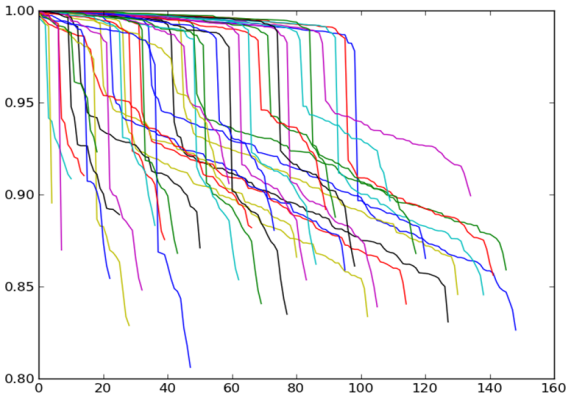

Fig. 2. Similarity curves in splitting process

Algorithm 1: SplittingCluster. *sc*

Data: D - Data set; SBS - Similarity computing procedure.
Result: Cluters set, $C = \{c_1, \cdots, c_m\}$, m is size of clusters.

```
1  begin
2      Initialize set C = {}
3      while D is not empty do
4          Select a object s from D randomly;
5          Compute all similarities between s and s' in D by
           SBS(s, s'), and a similarity list, SimList, is
           created;
6          Sort SimList in descending, and computing the
           threshold, T_sim, by T(SimList);
7          Create a new Cluster, c, and add c into C where
           c = {s} ∪ {s'|SBS(s, s') > Tsim};
8          Remove all the objects which are in set of
           {s'|SBS(s, s') > Tsim} from D.
9      Return C
10     end
```

The threshold value T_{sim} is a key point in the clustering procedure of Algorithm 1 because it controls the algorithm to create a new cluster or not. A differential calculation method is used to determine the threshold on the discrete similarity sequence. We observed that the distribution of the similarity sequence shows obvious pattern as shown in Fig. 2.

The data set in Fig. 2 is Iris data obtained from UCI, and the horizon axis is similarity sequence and the vertical axis stands the similarities. Each curve represents an iteration in the above algorithm. From the figure, it shows that the objects which are similar with the current object s are distributed in a small range of the similarity sequence and then the curve value is suddenly reduced that there is a point with high gradient. Therefore, it is known easily that the threshold is obtained at the curve inflection point

whose second derivative is equal to zero. A new cluster is created that similarity value is less than the threshold T_{sim}.

The complexity of SplittingCluster algorithm is in $[O(n), O(n^2)]$ where n is the size of data set, which depends on the actual number of categories of data sets.

Proof: In the worst case, each cluster only has one member in the data set, then the total number of calculations is $n * (n+1)/2 \sim O(n^2)$; In the best case, it is assumed that the data set has only one cluster, so the members size in the cluster is n, it only need to calculate n times that is $O(n)$. ■

We can further improve the computation efficiency of the clustering process which is particularly important for relatively large data sets, or the length of the text is relatively long. In this paper, the data set is divided into many small data sets according the anatomy, because there are 73.35% disease name texts containing the human organ/tissue name. The priori classification knowledge of the body's organ/tissue is relatively easy to obtain. Therefore, the data set is separated into dozens of small data sets and each sub data set is easier to be processed than the original collection.

B. Central Concept Extraction

A number of clusters are obtained using the Algorithm 1 and each cluster includes the synonymous texts. However, two problems should be resolved as following:

- There is no unified disease name in each cluster. The texts in each cluster have same meaning, but there is no one expression which is called **central concept** to represent the semantic of the cluster. For example, a cluster whose members are {"右眼糖尿病视网膜病变 (right eye, diabetic retinopathy)", "左眼糖尿病视网膜病变 (left eye, diabetic retinopathy)", "双眼糖网 (eyes of diabetic retinopathy)", "双眼糖尿病视网膜病变（V 期）(eyes of diabetic retinopathy - stage V)", "双眼糖尿病视网膜病 (eyes of diabetic retinopathy)", "双眼增殖性糖尿病视网膜病变 (Eyes with proliferative diabetic retinopathy)"} need to extract the central concept {糖尿病视网膜病变 (diabetic retinopathy)}.
- There are clusters which have same central concept or some members are more close to other central concepts. Therefore, it is need to merge the clusters once the central concept is extracted and the merging procedure is iterative until the size of clusters is not changed any more.

The texts in a cluster have similar language lexical, and the term frequency in the text can be obtained. A term co-occurrence based concept extraction method is proposed in this paper. We use the sequence pattern analysis methodology to extract the central concept. The text is seen as a time sequence of word. e.g. "右眼糖尿病视网膜病变 (the right eye, diabetic retinopathy)" is translated into a time series {右，眼，糖，尿，病，视，网，膜，病，变}. The task of central concept extraction is turned into another problem that how to find the longest frequent subsequence in the cluster. It is known that there are many ways to find the frequent subsequence in the sequence database, e.g. GSP [19], PrefixSpan [20] and etc.. The example of central concept extraction process is shown in Table III.

TABLE III
FREQUENCY SEQUENCE MINING BASED CENTRAL CONCEPT EXTRACTING

Words sequence of short text	Frequency sub-sequences and its length
[[右], [眼], [糖], [尿], [病], [视], [网], [膜], [病], [变]]; [[左], [眼], [糖], [尿], [病], [视], [网], [膜], [病], [变]]; [[双], [眼], [糖], [尿], [病], [视], [网], [膜], [病]] [[双], [眼], [糖], [网]]; [[双], [眼], [糖], [尿], [病], [视], [网], [膜], [病], [变], [（], [V], [期], [）]]; [[双], [眼], [增], [殖], [性], [糖], [尿], [病], [视], [网], [膜], [病], [变]]	眼糖尿病视网膜病变 (9) 眼尿病视网膜病变 (8) 眼糖病视网膜病变 (8) 眼糖尿病网膜病变 (8) 眼糖尿病视膜病变 (8) 眼糖尿病视网病变 (8) ...

Finally, the central concept of the above example cluster is obtained as " 眼糖尿病视网膜病变 (diabetic retinopathy)".

The similarity distributions of internal cluster and external clusters are changed once the central concepts are extracted. It is need to merge the clusters with same central concept and move the members to new cluster with more similar central concept. The empty cluster should be deleted until the number of clusters is not changed. The clusters merging process, MergingCluster, is defined as Algorithm 2.

Algorithm 2: MergingCluster. *mc*

```
Data: C-clusters set of D.
Result: The new clusters C' after merging.
begin
    Let m = |C|
    while size of C' is not changing do
        for every cluster c in C do
            for every s in c do
                if s is more likely element of c' except c
                then
                    Remove s from c.
                    Add s into c'.
        Delete all the empty clusters.
        Computing the new central concept for every
        cluster.
    Return C' = C
end
```

The experiment shows that the MergingCluster needs three to five iterations to obtain a stable state in most of the data sets. Fig. 3 (a) and (b) show the process of SplittingCluster and MergingCluster algorithms.

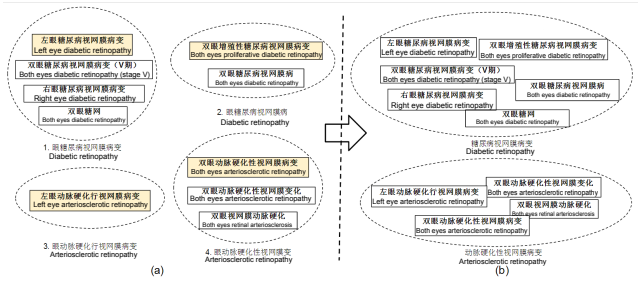

Fig. 3. The result of MergingCluster procedure on SplittingCluster output

There are some similar points between MergingCluster algorithm and K-means/K-medoids clustering algorithm. Firstly, they are all based on the similarity function to determine whether a data object belongs to a cluster. Secondly, a global optimization constrains is need to control the clustering procedure to a stable state. The main differences between them are described as following aspects.

- The size of clusters: Our algorithm does not require a pre-defined number of clusters, and we can get all the clusters using the MergingCluster. However, K is defined before the K-means or K-medoids algorithm running.
- The central point of cluster: The algorithm in this paper extracts the central concept of the cluster as central point. But The central point in the K-means or the K-medoids algorithm is usually the logic data center or centroid.
- Clusters number changing dynamically: The size is dynamically changing in the merging procedure until the number of cluster is no longer changed. The K-means or the K-medoids algorithm will not change all the time.

It is easy to know that the maximum complexity of MergingCluster algorithm is $O(c^2n)$ where c is the number of clusters and n is the data size.

Proof: The worst scenario is that there are n type clusters, each cluster only one member, that c is equal to n, then the need to calculate $c*[1*(c-1)+...+1*(c-1)] = c*[n(c-1)] \sim O(c^2n)$. The best case is that there is only one cluster, that has no merging procedure. ■

V. Hierarchical Concept Clustering

The first step of HCCST algorithm just uses the lexical features other than the semantic features of the objects. This section presents the semantic based hierarchical concept clustering procedure. semantic term is the basic unit that contains the human subjective opinions. E.g. the two term " 动脉 (Artery) " and "硬化 (Sclerosis) " stand different meanings, but the combination term " 动脉硬化 (Arteriosclerosis)" expresses a new definitely concept which stands more human subjective cognitive knowledge. Therefore, all the central concept texts are divided into semantic terms before the hierarchical disease concept construction.

A. Semantic Term Extraction

We know that there are some rules in the disease name text, such as the disease name generally contains the organ/tissue term. e.g. the disease name "动脉硬化 (Arteriosclerosis)" is composed by Anatomy(动脉 -Artery) and Morphology(硬化 -Sclerosis), " 动脉硬化性视网膜病变 (Arteriosclerotic retinopathy)" is built up by Cause(动脉硬化性 -Arteriosclerotic), Anatomy(视网膜 -Retina) and Morphology(病变 -Pathological changes)". Semantic term extraction from the disease names is the key step to create the disease concept graph.

The semantic term extraction generally uses POS parsing (part-of-speech, POS)[6] technology which make a sentence decomposed into a tree structure and each node labeled by POS tag. This method can be used to obtain the noun or noun phrase. In order to obtain the valid semantic phrases, some studies also use a variety of semantic phrase filtering method[14]. In this study, the texts in the data set are short and many semantic terms and relationship between them in each text are relatively fixed. Dictionary based semantic term extraction is applied in this study. Each text in the data set will be broken into a number of independent semantic terms/phrases which are treated as basic semantic units. An example is shown in Table IV in which five different terms are obtained finally.

TABLE IV
Semantic term extraction

ID	Central Concept	Sementic terms
1	糖尿病视网膜病变 (Diabetic retinopathy)	[糖尿病 (Diabetes), 视网膜 (Retina), 病变 (Pathological changes)]
2	动脉硬化性视网膜病变 (Arteriosclerotic retinopathy)	[动脉 (Artery), 硬化 (Sclerosis), 视网膜 (Retina), 病变 (Pathological changes)]

B. Potential Hierarchy Concept Pair Recognition

The next task is to identify and filter the upper concept in the hierarchy concept pair from the sematic terms which are extracted from the central concept text set. There are some previous works described in first section to build a hierarchical concept graph, e.g. the formal concept analysis (FCA)[13], hierarchical clustering [3][4], subsumption [7][8]. It is hard to process the sparse matrix problem in feature space. Therefore, an improved semantic induction technique is used in this study, which can find the initial hierarchy concept pairs. It is called potential hierarchy concept pair which is defined as following.

Definition 4: (Potential Hierarchy Concept Pair) Given two concepts, c_1 and c_1, (c_1, c_2) is called the Potential Hierarchy Concept Pair if it meets equation (2).

$$P(c_1|c_2) \geq t, P(c_2|c_1) < t, \tag{2}$$

where t is the threshold of c_1 and c_1 co-occurrence. If the concept c_2 appears in the case, the frequency of concept c_1 occurrence is at least greater than t in the data set. while concept c_1 appears in the case, the frequency of c_2 occurrence is less than t. This means that the concept c_1 is the potential parent concept of the concept c_2.

We can find a lot of potential concepts according principle in definition 4, and some concepts may have multiple parent concepts. We need to filter out useless parent concepts. This article uses the mutual information method to select the potential hierarchy concept pair and the mutual information of concept is defined as following.

Definition 5: (Mutual Information of Concepts) Given term set T, the concept set X and concept set Y, (x, y) is the potential hierarchy concept pair where $x \in X$, $y \in Y$, then the mutual information of the concept set X and Y , $CMI(X, Y)$, is defined as equation (3).

$$CMI(X,Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \qquad (3)$$

where $p(x, y)$ stands the co-occurrence frequency of concept x and y in the data sets in T, $p(x)$ and $p(y)$ is the occurrence frequency of concept x and y in the data set T respectively.

Definition 6: (Principal of Potential Hierarchy Concept Pair Selection) Given the concept y and the concept set X, the concept y is parent concept of X when $CMI(y, X) > tc$ and $|X| > 1$ are satisfied where tc is a pre-defined threshold value.

Definition 7: (Concept Merging) Given two concepts x, y, z and the concept set Z where $z \in Z$. if concept x and concepts y are the parent concept of z, then the concept x and y are merged into a new concept denoted as $x \wedge y$ (or $y \wedge x$).

Definition 8: (Reconstruction of Hierarchy Concept) Given the concepts x, y and concept sets Z_x, Z_y, x is parent concept Z_x, y is parent concept of Z_y. If the $Z_x \subset Z_y$ (or $Z_y \subset Z_x$), We define y is the parent concept of x and $Z_y = Z_y - Z_x$.

Definition 9: (Reconstruction of Potential Hierarchy Concept Pair) Given the concepts x, y and concept sets Z_x, Z_y, concept $z_x \in Z_x$, $z_y \in Z_y$, x is the parent concept of Z_x, y is parent concept of Z_y. If the concept of set $Z = Z_x \cap Z_y$ exist, and $2 * |Z|/(|ZX| + |ZY|) > tp$, then we merge the potential concept x, y, and reconstruct the conceptual level. x and y are the parent concept of $x \wedge y$ (or $y \wedge x$), $Z_x = Z_x - Z, Z_y = Z_y - Z$.

A hierarchical concept graph can be built up based on those potential hierarchy concept pair discovery and filtering principle in term set T. The process of hierarchical concept graph construction is defined as Algorithm 3.

The ConceptCluser algorithm makes the T as a bipartite graph structure firstly, then creates a multi-level graph on the bipartite graph. For example, the data is shown as in Fig. 4(a), and Fig. 4(b) shows the extraction of

Algorithm 3: ConceptCluser. *cc*

Data: T-Semantic term set， $tv = (t, tc, tp)$- threshold vector.
Result: HC - set of *sc*.

```
begin
    Initialize HC = ∅
    for (t1, t2) in T do
        if t1 and t2 meet the principal in definition 4 (Let t1 is parent concept of t2) then
            Add t2 into the child concept set of t1.
    for sc = (t, st1, st2, ..., stm) in HC do
        m is size of child concept of t
        if sc not meets the principal in definition 6 then
            Remove sc from HC.
    for (sc1,sc2) in HC do
        Computing the hierarchical relationships on sc1 and sc2 in HC using definition 7, 8 and 9.
    Return HC.
end
```

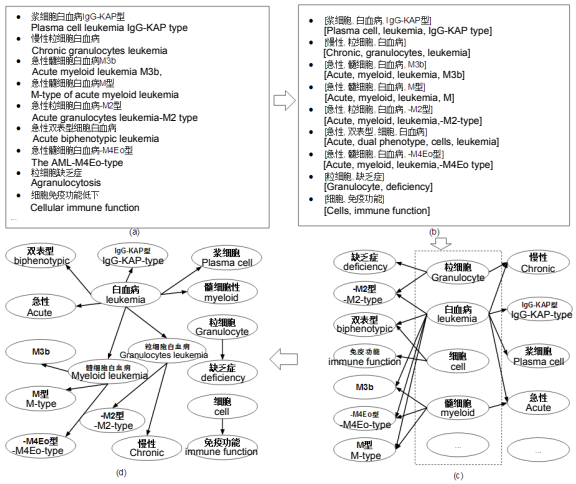

Fig. 4. The result of ConceptCluser procedure

the semantic term. The potential hierarchy concept pairs are found in Fig. 4(c) which can be seen as a bipartite graph. The Hierarchical term graph is obtained following the above definitions finally which is shown in Fig. 4 (d).

C. Hierarchical Disease Taxonomy Construction

The graph created by Algorithm 3 from term set T is not a hierarchical concept graph, because the terms in the lowest level should be mapped to original disease texts. We first define the upper concept of short text as following definition.

Definition 10: (Upper Concept of Short Text) Given a text s, concept x and the concept y, $y \rightarrow x$, if x and y appear in the text of s together. Concept x is defined as the upper concept of s.

In the potential hierarchy concept pair set created in previous sections, every concept has parent concepts or

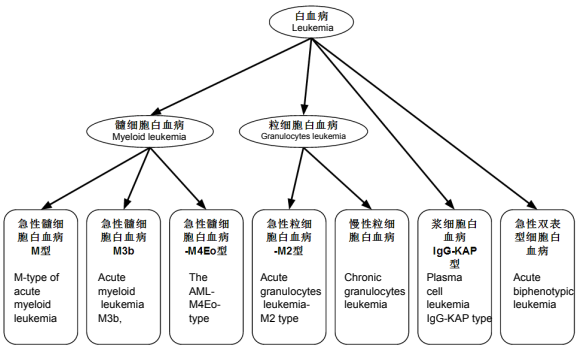

Fig. 5. An example of Hierarchical Disease Taxonomy

child concepts. It is easy to make a traversal on the hierarchical term graph from the root and the term sequence $[x_1, x_2, \cdots, x_p]$, and $x_1 \rightarrow x_2, x_2 \rightarrow x_3, \cdots, x_{p-1} \rightarrow x_p$ is obtained in term T. The text s in D is found that contains the above concept terms. Fig. 5 shows the final hierarchical Disease Taxonomy in which data is same as Fig. 4.

The hierarchical concept graph which is created by the method proposed in this paper is more actionable and operability for it is from actual data. The data set used in this paper only a subset of the complete disease, so the hierarchical disease concept graph is not cover all the diseases, and it is the subset of disease set in ICD-10 standard.

VI. Evaluation

The data set used in this paper is obtained from the Electronic Medical Records system in a hospital. We use data cleaning program to extract the disease diagnosis text as the target short text data set. The characters of the data set are shown as table V.

TABLE V
Characters of the short text corpus

Character	Value
Departments	62
Disease names	52957
Mixed texts	8.695%
Avg. diseases per case	3.143
Avg length of short text	6.310
Max length of short text	27
Min Length of short text	2

The disease name text often contains some extra and special characters which should be removed in the clustering process, such as brackets, special symbols. Meanwhile, in order to calculate the text similarity accuracy, the high-frequency ($\geq 7\%$) and low-frequency($\leq 8.36e{-}03\%$) words also need to be filter out which is shown as table VI.

In the medical data set, azimuth words and grade words which are used to describe the position or level of the human organs are also removed, such as "前 (front), 后 (back), 左 (left), 右 (right), 上 (top), 下 (bottom), 内 (inside), 外 (outside), 升 (up), 降 (down), 一级 (1st level), 二级 (2nd level), 1 级 (1st level), 2 级 (2nd level), $\cdots$". In actual process, the data set is divided into some small data sets according the organ/tissue. The statistical result is shown as Fig. 6 where the number of each disease name is greater than 30. There are 33 disease classes in total which are used for performance evaluation.

TABLE VI
The high and low frequency words in dataset

ID	High frequent words	Frequency	Low frequent words	Frequency
1	性 (type)	0.29	烟 (smoke)	8.36e-05
2	右 (right)	0.12	疲 (tired)	8.36e-05
3	左 (left)	0.12	绀 (cyanogen)	8.36e-05
4	病 (illness)	0.11	氢 (hydrogen)	8.36e-05
5	后 (afterwards)	0.11	茨 (heights)	8.36e-05
6	炎 (inflammation)	0.10	皱 (wrinkle)	8.36e-05
7	术 (surgery)	0.10	存 (retain)	8.36e-05
8	肿 (swollen)	0.10	兴 (eagerness)	8.36e-05
9	血 (blood)	0.09	欣 (Glad)	8.36e-05
10	骨 (bone)	0.08	许 (allow)	8.36e-05
11	心 (cardiac)	0.07	梢 (tip)	8.36e-05

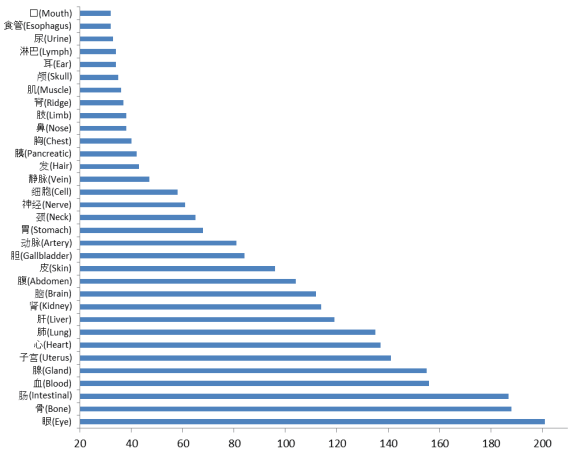

Fig. 6. The disease distribution related with human organ/tissue

The HCCST algorithm is implemented in Python v2.7. All experiments are conducted on a desktop computer with Intel Core 2 CPU of 2.80GHz, 4GB memory and Windows 7. There are rarely hierarchical concepts clustering algorithms on short text data to be comparison, so the contrast performance evaluation is not conducted. The t is set to 0.8 and the tc is set 0.03 The created concept numbers in the 33 classes with most disease name text are listed in the Table VII as well as the time performance. The average accuracy of the clustering on the dataset is 96.4% which achieves satisfactory performance according to the clinical standard for all disease names.

VII. Conclusions

This paper proposes a new hierarchical concept clustering method on short text data which takes advantage of

TABLE VII
HIERARCHY CONCEPTS EXTRACTED IN SECOND AND THREE LEVELS

ID	Class	short texts	2nd level	3rd level	Time (s)
1	眼 (Eye)	201	23	77	1.87
2	骨 (Bone)	188	26	114	1.61
3	肠 (Intestinal)	187	24	106	1.31
4	血 (Blood)	156	22	104	1.10
5	腺 (Gland)	155	27	85	0.88
6	子宫 (Uterus)	141	16	42	0.05
7	心 (Heart)	137	33	84	0.96
8	肺 (Lung)	135	24	53	0.73
9	肝 (Liver)	119	20	44	0.39
10	肾 (Kidney)	114	17	60	0.41
11	脑 (Brain)	112	27	57	0.36
12	腹 (Abdomen)	104	20	60	0.26
13	皮 (Skin)	96	26	59	0.44
14	胆 (Gallbladder)	84	20	46	0.14
15	动脉 (Artery)	81	14	48	0.17
16	胃 (Stomach)	68	15	36	0.14
17	颈 (Neck)	65	18	49	0.01
18	神经 (Nerve)	61	13	31	0.11
19	细胞 (Cell)	58	23	44	0.12
20	静脉 (Vein)	47	17	34	0.05
21	发 (Hair)	43	7	19	0.05
22	胰 (Pancreatic)	42	9	27	0.02
23	胸 (Chest)	40	10	25	0.02
24	鼻 (Nose)	38	7	23	0.02
25	肢 (Limb)	38	10	23	0.02
26	脊 (Ridge)	37	9	24	0.02
27	肌 (Muscle)	36	10	16	0.02
28	颅 (Skull)	35	8	26	0.07
29	耳 (Ear)	34	9	20	0.016
30	淋巴 (Lymph)	34	9	28	0.02
31	尿 (Urine)	33	7	19	0.01
32	食管 (Esophagus)	32	6	26	0.01
33	口 (Mouth)	32	9	20	0.015

hierarchical clustering and concept clustering approaches. The evaluation is conducted on Chinese medical disease name text data set and the result shows that HCCST achieves high accuracy and efficiency, which can be credited to the features of HCCST. Firstly, HCCST uses a new similarity measure method which covers all the problems in medical short text distance computing. Secondly, a adaptive clustering method is proposed for synonymous disease names without interaction. Thirdly, this paper uses a mutual information based potential hierarchy concept pair recognition method which improve the accuracy of the result.

HCCST introduces a new approach towards efficient and high quality clustering model on short text data. Our further work will focus on the following two tasks. The text is selected randomly in the splitting clustering algorithm, and we will do further experiments on the distribution of similarity to get unified order of the text selection. The final hierarchical disease network is obtained from a fixed test database, and how to update the present graph will a challenging useful research work in the next especially when new data are obtained.

ACKNOWLEDGMENT

This research is supported in part by Fundamental Research Funds for the Central Universities under grant (N120518001 and N110718001) and Open Research Fund of Beijing Key Laboratory of Magnetic Resonance Imaging and Brain Informatics.

REFERENCES

[1] E. T. Rhodes, M. B. Laffel, T. V. Gonzalez, et al. Accuracy of administrative coding for type 2 diabetes in children, adolescents, and young adults. Diabetes Care, 2007, 30(1): 141-143.

[2] N. Ramakrishnan, D. Hanauer and B. Keller, Mining electronic health records. Computer, 2010, 43(10): 77-81.

[3] A. K. Jain, M. N. Murty and P. J. Flynn. Data clustering: a review. ACM computing surveys (CSUR), 1999, 31(3): 264-323.

[4] P. A. Berkhin, survey of clustering data mining techniques//Grouping multidimensional data. Springer Berlin Heidelberg, 2006: 25-71.

[5] G. Chowdhury, Introduction to modern information retrieval. Facet publishing, 2010.

[6] K. T. Frantzi, S. Ananiadou and J. ichi Tsujii, The C-value/NC-value Method of Automatic Recognition for Multi-Word Terms, in: 2nd European Conference on Research and Advanced Technology for Digital Libraries (ECDL 1998), Springer, 1998, pp. 585–604.

[7] M. Sanderson and B. Croft, Deriving Concept Hierarchies from Text, in: 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1999), ACM, 1999, pp. 206–213.

[8] P. Schmitz, Inducing Ontology from Flickr Tags, in: Workshop on Collaborative Web Tagging (CWT 2006) collocated with the 15th World Wide Web Conference 2006 (WWW 2006), pp. 206–209.

[9] P. Nakov, A. Popova and P. Mateev, Weight functions impact on LSA performance. EuroConference RANLP, 2001, pp. 187-193.

[10] P. Shrestha, C. Jacquin and B. Daille, Reduction of search space to annotate monolingual corpora//Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011). 2011.

[11] D. Pinto, J. M. Benedí and P. Rosso, Clustering narrow-domain short texts by using the Kullback-Leibler distance//Computational Linguistics and Intelligent Text Processing. Springer Berlin Heidelberg, 2007, pp. 611-622.

[12] D. H. Fisher, Knowledge acquisition via incremental conceptual clustering. Machine learning, 1987, 2(2): 139-172.

[13] P. Cimiano, A. Hotho and S. Staab, Learning concept hierarchies from text corpora using formal concept analysis. Journal of Artificial Intelligence Research, 2005, 24(1): 305-339.

[14] F. Sclano and P. Velardi, TermExtractor: a Web Application to Learn the Shared Terminology of Emergent Web Communities//Enterprise Interoperability II. Springer London, 2007: 287-290.

[15] H. V. Nguyen and L. Bai, Cosine similarity metric learning for face verification//Computer Vision–ACCV 2010. Springer Berlin Heidelberg, 2011: 709-720.

[16] R. D'hulst and G. J. Rodgers, The hamming distance in the minority game. Physica A: Statistical Mechanics and its Applications, 1999, 270(3): 514-525.

[17] W. J. Heeringa, Measuring dialect pronunciation differences using Levenshtein distance. University Library Groningen, 2004.

[18] L. Hamers, Y. Hemeryck, G. Herweyers, et al. Similarity measures in scientometric research: the Jaccard index versus Salton's cosine formula. Information Processing & Management, 1989, 25(3): 315-318.

[19] Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In Proc. 5th Int. Conf. Extending Database Technology(EDBT' 96), Avignon, France,Mar. 1996: 3–17.

[20] J. Han, J. Pei, B. Mortazavi-Asl, et al. PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth//Proceedings of the 17th International Conference on Data Engineering. 2001: 215-224.

[21] The Challenges of ICD10 Implementation. 2011. From: http://geekdoctor.blogspot.com.au/2011/09/challenges-of-icd10-implementation.html
[22] Top Documentation Issues for ICD-10. 2011. From: http://journal.ahima.org/2011/04/18/top-documentation-issues-for-icd-10/