Research on Standardization Algorithms for Examination Indicators in Regional Healthcare Platforms

Authors: Zhang Jiaying, Wang Qi, Zhang Zhixing, Ruan Tong, Zhang Huanhuan, He Ping, Zhang Huanhuan

Date: 2018-11-13T00:00:00+00:00

Abstract

Due to the lack of a complete and available indicator synonym database for conducting indicator mapping, the different terminologies used by various hospitals for the same laboratory test indicator have severely impacted the interconnection and sharing of medical information across regions, thus necessitating the standardization of laboratory test indicators. This can be viewed as an entity alignment problem; however, indicators only have corresponding values and value ranges, making it difficult to utilize attribute information as in knowledge base instance matching, nor do they possess contextual information as in entity linking, and there exists no standard knowledge base that provides standard names for all indicators. To address the aforementioned issues, this paper proposes an indicator standardization algorithm that first performs clustering based on the literal features of indicators, and then iteratively conducts binary classification mapping using similarity features and chunk scoring features. Experimental results demonstrate that the final binary classification mapping achieves an F1-score of 85.27%, thereby proving the effectiveness of the method.

Full Text

Preamble

Lab Indicator Standardization in a Regional Medical Health Platform

Zhang Jiaying[1], Wang Qi[1], Zhang Zhixing[1], Ruan Tong[1], Zhang Huanhuan[1], and He Ping[2]
[1]East China University of Science and Technology, Shanghai 200237
[2]Shanghai Hospital Development Center, Shanghai 200041
[1]zhangjy_{ecust}@163.com

Abstract
Due to the lack of a complete synonym list for indicator mapping, different hospitals may use different names for the same lab indicator. Lab indicator name discrepancy has greatly affected the medical information sharing and exchange among hospitals. It is becoming increasingly important to standardize the lab indicators. Such a problem can be seen as an entity alignment task to map different indicators into standard ones. However, a lab indicator only involves its name and value, not including any extra properties or contexts which is needed by existing knowledge base (KB) alignment or entity linking methods. More importantly, there exists no available standard KBs to provide standard indicator terms. Therefore, we cannot implement these existing methods directly. To solve the problem, in this paper, we present the first effort to work on lab indicator standardization. We propose a novel standardization method, which firstly cluster the indicators based on their names and abbreviations, and then iteratively employ a binary classification algorithm based on similarity features and partition score features for indicator mapping. Experimental results on the real-world medical data show that the final classification achieves a F1-score of 85.27%, which indicates that our method improves the quality and outperforms state-of-the-art approaches.

Keywords: regional medical health platform; lab indicator; standardization; clustering; classification

Funding: This work is supported by the National Natural Science Foundation of China (61772201), the National Key R&D Program of China for "Precision medical research" (2018YFC0910500), and the National Major Scientific and Technological Special Project for "Significant New Drugs Development" (2018ZX09201008).

Corresponding Author: Zhang Huanhuan (hzhang@ecust.edu.cn)

Introduction

As medical informatization continues to deepen, regional medical health platforms have been established on top of existing healthcare systems both domestically and internationally. Taking Shanghai as an example, with the official launch of the Shanghai Medical Linkage Project in March 2008, the city built a clinical diagnosis and treatment information sharing platform encompassing 38 tertiary hospitals, enabling the exchange and sharing of patients' basic information, medical records, inpatient case files, medical orders, medical expenses, laboratory test reports, and medical imaging examination reports, while strengthening collaborative diagnosis and treatment among hospitals through websites and other auxiliary systems. However, due to historical reasons, different hospitals use varying names for the same lab indicator. For instance, "serum sodium"alone has over 10 different expressions such as "sodium ion concentration," "NA+," "arterial blood sodium," and "blood sodium (Na)." Since no complete and usable indicator synonym database currently exists for indicator mapping, this issue

has severely impacted the interconnection and sharing of medical information across regions. Consequently, standardizing lab indicators in regional medical health platforms—mapping different names of the same indicator from various hospitals to a unified standard name—has become critically important. Nevertheless, because lab indicators involve substantial medical knowledge and each hospital' s indicator system is complex and heterogeneous, manual standardization by medical professionals would consume considerable time and effort. Therefore, designing an algorithm for lab indicator standardization has become essential.

The lab indicator standardization problem can be viewed as an entity alignment task, which maps candidate indicators in the medical health platform to standard indicators. Current entity alignment tasks mainly fall into two categories: instance matching between entities in different knowledge bases [1][2], and entity linking between entities in text and those in knowledge bases [3][4]. The former typically utilizes attribute information of entities in knowledge bases for instance matching, while the latter leverages contextual information of entities in text and attribute information of entities in knowledge bases for entity linking. However, our task differs from both: lab indicators exist in electronic medical records with only corresponding values and value ranges, lacking attribute information; simultaneously, they do not possess contextual information like entities in text; more importantly, no standard knowledge base exists in our task to provide standard indicator names. In other words, existing methods cannot be directly applied to this task.

In light of this, we propose an indicator standardization algorithm framework for lab indicators in regional medical health platforms. The framework first preprocesses indicator data, then utilizes literal features of indicators to cluster different indicators into clusters via a density-based clustering algorithm, thereby narrowing the scope for indicator alignment. Subsequently, it determines a standard name for each cluster and employs a binary classification algorithm to identify synonymous indicators of the standard name within the cluster. For the remaining non-synonymous indicators, it selects a new standard name and continues searching for synonymous indicators using the binary classification algorithm iteratively[1], repeating this process until all indicators within each cluster are synonymous or only one indicator remains in the cluster. Finally, medical professionals review and correct the indicator alignment results. Experimental results on a dataset from eight tertiary hospitals in Shanghai demonstrate that the final binary classification mapping algorithm achieves an F1-score of 85.27%.

[1]Of course, one could also re-cluster all non-synonymous indicators and iterate in this manner. However, considering that there are too many different indicators across 38 hospitals in practical applications, the time cost of clustering would be very high. As a preliminary attempt at a standardization algorithm for lab indicators in regional medical health platforms, this paper temporarily uses iterative binary classification for standardization.

1. Related Work

The lab indicator standardization problem in regional medical health platforms can be viewed as an entity alignment task, which maps different indicator names from various hospitals to unified standard indicators. Current entity alignment tasks can be broadly divided into two categories: instance matching between entities in different knowledge bases, and entity linking between entities in text and those in knowledge bases.

Many studies have focused on instance matching between knowledge base entities. These studies leverage attribute information of entities in knowledge bases for matching and can be categorized into two types: pairwise entity matching methods and collective entity matching methods. Pairwise entity matching methods mainly include traditional probability model-based approaches, supervised learning methods, clustering methods, and active learning methods. Traditional probability model methods perform pairwise entity comparison based on attribute similarity [5][6]. Supervised learning methods commonly use decision trees [1][7][8], support vector machines [2][9], and ensemble learning [10][11] for binary classification. Clustering methods utilize attribute similarity for entity clustering [12][13][14]. Active learning methods train classification models iteratively through human-computer interaction [15][16][17]. Collective entity matching methods also consider associated entities of entities, with common approaches including LDA methods [18][19], CRF models [13][20], and Markov Logic Networks [21][22].

Regarding entity linking between text entities and knowledge base entities, major approaches include probability generative model-based methods [3][23], topic model-based methods [4][24], graph-based methods [25][26][27][28], and deep neural network-based methods [29][30][31][32].

It should be noted that our research differs from both aforementioned studies: lab indicators exist in electronic medical records with only corresponding values and value ranges, making it difficult to utilize attribute information as in knowledge base instance matching; simultaneously, they lack contextual information like text entities, making entity linking methods inapplicable; more importantly, no standard knowledge base exists in our task to provide standard indicator names.

2. Indicator Standardization Algorithm

The overall process of the indicator standardization algorithm is illustrated in Figure 1 [Figure 1: see original paper]. First, indicator data undergoes preprocessing to achieve case unification, unit unification, and indicator reference value extraction. Next, utilizing literal features of indicators, a density-based clustering algorithm groups different indicators into clusters to narrow the scope for indicator alignment. Then, a standard name is determined for each cluster, and a binary classification algorithm identifies synonymous indicators of the standard name within the cluster for indicator mapping. For the remaining

non-synonymous indicators, a new standard name is selected, and the search for synonymous indicators continues using the binary classification algorithm iteratively until all indicators within each cluster are synonymous or only one indicator remains in the cluster. Finally, medical professionals review and correct the indicator alignment results.

2.1 Data Preprocessing

The indicator data in medical records includes fields such as indicator name, abbreviation, reference value, unit, affiliated test item, test indicator result, and abnormal indicator prompt, excluding optional items[2]. Among these, the affiliated test item loses significance as a feature for indicator standardization because standards vary across hospitals; the test indicator result loses significance because its values differ by patient; and the abnormal indicator prompt loses significance because it lacks discriminative power for indicators. Therefore, the usable fields are essentially limited to four items: indicator name, abbreviation, reference value, and unit. Data preprocessing of indicators primarily involves unifying indicator case, unifying indicator units, and extracting indicator reference values.

[2]In practice, optional fields generally contain no data. For example, the "LOINC code"field is an important feature for identifying whether indicators are the same; however, as an optional field, no data is actually entered.

2.2 Indicator Clustering

To narrow the scope for indicator alignment, this paper employs a density-based clustering algorithm to group different indicators into clusters. Density-based clustering algorithms partition clusters according to the compactness of sample distribution, primarily examining sample connectivity and continuously expanding clusters based on connectable samples to obtain final results.

Based on the DBSCAN [33] algorithm, this paper performs indicator clustering using indicator names and their abbreviations. Specifically, given an indicator set $D = \{x_1, x_2, \ldots, x_n\}$, where $x_i = (name_i, abbr_i)$, with $name_i$ representing the indicator name of the i-th indicator and $abbr_i$ representing the abbreviation of the i-th indicator, we define the -neighborhood and core objects as follows:

Definition 1 (-neighborhood) For $x_i \in D$, its -neighborhood consists of all samples in dataset D whose distance from x_i is not greater than ϵ, i.e., $N_\epsilon(x_i) = \{x_j \in D \mid dist(x_i, x_j) \leq \epsilon\}$.

Definition 2 (Core object) If x_i' s -neighborhood contains at least $minPts$ samples, i.e., $|N_\epsilon(x_i)| \geq minPts$, then x_i is a core object.

Specifically, when determining the -neighborhood, this paper defines a joint distance $dist_{joint}(x_i, x_j)$: the indicator data x_i, x_j is divided into two parts for calculation. First, the cosine distance between indicator names $name_i$ and $name_j$ in multi-hot form (where different dimensions in the 0-1 vector represent

different Chinese characters) is computed. Then, the edit distance between indicator abbreviations $abbr_i$ and $abbr_j$ is calculated, which represents the minimum number of operations required to transform $abbr_i$ into $abbr_j$ through insertion, replacement, and deletion operations. Finally, the harmonic mean integrates the two distances to obtain the joint distance.

The clustering algorithm starts from core objects and continuously expands outward to generate clusters. Its pseudocode is shown in Algorithm 1.

```
Algorithm 1: Density-based clustering algorithm
Input: (1) Indicators set D = {x_1, x_2, ⋯, x_n}
       (2) Neighborhood parameters ( , minPts)
Output: Cluster partition C = {C_1, C_2, ⋯, C_m}
1.  Initialize the Core Object collection: Ω =
2.  for i = 1 to n do
3.      Determine the Eps-neighborhood: N_ (x_i)
4.      if |N_ (x_i)| $\geq$ minPts then
5.          Add x_i to the Core Object set: Ω = Ω   {x_i}
6.      end
7.  end
8.  Initialize number of clusters: k = 0, cluster set: C =  , unvisited set: Γ = D
9.  while Ω $\neq$   do
10.     Record currently not visited collection: Γ_{old} = Γ
11.     Select a core object o randomly from Ω
12.     Initialize the queue Q = [o]
13.     Remove o from Γ, Ω: Γ = Γ \ {o}; Ω = Ω \ {o}
14.     while Q $\neq$   do
15.         Take the first sample q in queue Q
16.         if |N_ (q)| $\geq$ minPts then
17.             S = Γ   N_ (q)
18.             Q = Q + S
19.             Γ = Γ \ S
20.         end
21.     end
22.     k = k + 1
23.     Generate cluster: C_k = Γ_{old} \ Γ
24.     Ω = Ω \ C_k
25.  end
26.  return C
```

It should be noted that since clustering is an unsupervised learning process, two potential issues may arise: 1) Indicators clustered together may have different medical meanings but are grouped due to similar names or abbreviations; 2) Some outliers are neither core objects nor accessible through core objects, and thus remain unclustered. Therefore, post-processing of clustering results is necessary.

1. **Unit verification**. Assuming that synonymous indicators have the same

unit, unit verification can be performed on each cluster to separate indicators with different units into different clusters.

2. **Outlier handling**. For unclustered outliers, there are two processing options: first, assign outliers to the nearest cluster with matching units based on distance; second, considering that outliers are far from other clusters, they may represent entirely new indicators. This paper adopts the second option.

2.3 Intra-Cluster Binary Classification

Even after post-processing, unsupervised clustering algorithms cannot guarantee that all indicators within a cluster are synonymous. Therefore, this paper determines a standard name for each cluster and uses a binary classification algorithm to partition indicators within the cluster into two categories: synonymous indicators of the standard name and non-synonymous indicators. Specifically, to facilitate post-processing correction by medical professionals and considering that standard indicators should be the most commonly used ones, this paper selects the most frequently occurring indicator within the cluster as the standard indicator.

1) **Data augmentation**. Since it is difficult for medical professionals to enumerate all synonymous indicators from scratch, and some indicators may have synonyms completely unrelated to their names (e.g., "B-type natriuretic peptide" and "brain natriuretic peptide"), this paper augments the training dataset by manually annotating some synonymous indicators for classifier training and extracting synonyms of standard indicators from SNOMED CT knowledge base [34], LOINC knowledge base [35], and Baidu Encyclopedia[3]. Since the SNOMED CT knowledge base is entirely in English with no Chinese version currently available, translation tools such as Baidu Translate[4], Tencent Translate[5], and iCIBA Translate[6] are used to translate English indicators into Chinese. It should be noted that even for the same indicator, translation tools may produce different translation results, making translation itself another source for obtaining synonyms. Table 1 provides an example of synonymous indicators for "B-type natriuretic peptide" after data augmentation.

Table 1 Example of synonymous indicators

Indicator Name	Synonym	Synonym Sources
B 型钠尿肽	B 型利钠肽	Baidu Encyclopedia
	B-型利钠肽	Baidu Encyclopedia
	B 型钠尿肽	LOINC, Tencent Translation
	利钠肽 B 型	Tencent Translation
	脑促尿钠排泄肽	iCIBA Translation
	脑利钠肽（物质）	Tencent Translation, Baidu & iCIBA Translation

2) **Feature extraction**. This paper designs two types of features for indicator binary classification: similarity features and partition scoring features.

Similarity features. These features primarily consider the name similarity and abbreviation similarity between each candidate indicator in the cluster and the standard indicator and all its synonyms.

For convenience of description, taking name similarity as an example (abbreviation similarity follows the same principle), we designate the candidate indicator name in the cluster as $name_x$ and the standard indicator name set as $S = \{s_1, s_2, \ldots, s_n\}$, where subscript n represents the total number of standard and synonymous indicators. We measure similarity using the following four metrics:

—**Longest Common Subsequence Similarity**: $\text{sim}_{lcs}(name_x, S) = \max_{s_i \in S} \frac{|\text{lcs}(name_x, s_i)|}{\min(|name_x|, |s_i|)}$, where $|\text{lcs}(name_x, s_i)|$ represents the length of the longest common subsequence between the candidate indicator name and a standard indicator name. This similarity can identify indicators with hypernym-hyponym relationships, such as "blood glucose" and "blood glucose (emergency)," which have a longest common subsequence similarity of 1.

—**Jaccard Similarity**: $\text{sim}_{jaccard}(name_x, S) = \max_{s_i \in S} \frac{|name_x \cap s_i|}{|name_x \cup s_i|}$, where $name_x$ and s_i are treated as sets of characters. This can identify indicators with different name orders, such as "B-type natriuretic peptide" and "natriuretic peptide B-type," which have a Jaccard similarity of 1.

—**Cosine Similarity**: $\text{sim}_{cos}(name_x, S) = \max_{s_i \in S} \frac{name_x \cdot s_i}{\|name_x\| \|s_i\|}$, where both $name_x$ and s_i are in multi-hot form (different dimensions in the 0-1 vector represent different Chinese characters). This similarity measures the cosine angle between two multi-hot indicator names and is less affected by formatting issues such as inserting "-" in the middle.

—**Edit Similarity**: $\text{sim}_{edit}(name_x, S) = \max_{s_i \in S} \left(1 - \frac{\text{med}(name_x, s_i)}{\max(|name_x|, |s_i|)}\right)$, where $\text{med}(name_x, s_i)$ represents the minimum number of operations required to transform $name_x$ into s_i through insertion, replacement, and deletion operations. This similarity measures the edit distance between two indicator names.

Partition scoring features based on one-to-many fields. Partition scoring features are primarily designed for one-to-many fields such as indicator reference values. For indicator reference values, since different hospitals may set slightly different upper and lower bounds for the same indicator, in practice there exists a phenomenon where one indicator name corresponds to multiple reference values. To address this issue, this paper proposes a reference value-based indicator partition scoring algorithm, drawing inspiration from the knowledge base entity alignment partition algorithm in reference [36]. The indicator partition scoring algorithm is based on two assumptions: first, identical indicators have similar reference values; second, indicators with similar reference

values may be the same indicator. Therefore, the partition scoring algorithm consists of two parts: first, find the most similar candidate indicator reference value for each reference value of the standard indicator; then, construct matching partitions between candidate indicators and standard indicators from these most similar reference values. It should be noted that since the same indicator may have multiple reference values, the algorithm allows the same indicator to appear in different partitions. This paper calculates the weighted average score of candidate indicators based on the weights of different partitions, which serves as a classification feature.

Specifically, given a candidate indicator x in a cluster, its corresponding reference value set is $Ref_x = \{ref_{x1}, ref_{x2}, \ldots, ref_{xm}\}$, and the reference value set of the standard indicator (and its synonyms) is $Ref_s = \{ref_{s1}, ref_{s2}, \ldots, ref_{sn}\}$. This paper defines reference value similarity as follows:

Definition 3 (Reference Value Similarity) Given two indicator reference values ref_x and ref_s, the reference value similarity $\text{sim}_{ref}(ref_x, ref_s)$ is defined as the cosine similarity after converting the reference value intervals into one-hot vectors.

For each reference value ref_{si} in the standard indicator' s reference set, we find the most similar candidate indicator reference value ref_{xj} from the cluster, and these two indicators form a reference value pair (ref_{xj}, ref_{si}). Based on the reference value pair, an indicator set pair $p_i = (X_i, S_i)$ can be constructed, where X_i is the set of all candidate indicators with reference value ref_{xj}, and S_i is the set of all standard indicators and their synonyms with reference value ref_{si}. The reference value pair similarity is then defined as:

Definition 4 (Reference Value Pair Similarity) Given two reference value pairs $p_i = (X_i, S_i)$ and $p_j = (X_j, S_j)$, their similarity is defined as the cosine similarity between their indicator sets after converting them to one-hot form.

As shown in Figure 2 [Figure 2: see original paper], the standard reference value ref_{s1} is the interval [0, 100], and its most similar candidate reference value ref_{x1} is also the interval [0, 100], thus its corresponding indicator set pair is $p_1 = (X_1, S_1) = (\{A, B\}, \{a, b\})$. Similarly, the standard reference $ref_{s2} = [0, 125]$ corresponds to the indicator set pair $p_2 = (X_2, S_2) = (\{A, B, C\}, \{a, b\})$.

During partition matching, if the similarity between two reference value pairs exceeds threshold θ, i.e., $\text{sim}(p_i, p_j) > \theta$, then their candidate indicator sets X_i, X_j and standard indicator sets S_i, S_j will be included in the same partition. Intuitively, the more indicators two reference value pairs share, the more likely they are to be grouped into the same partition.

After partitioning, this paper performs scoring on each partition. Define the partition result as $\mathcal{B} = \{B_1, B_2, \ldots, B_n\}$, where n is the number of partitions. For any partition B_i, its score $\text{score}_i = \frac{|S_i|}{|S'|^\alpha}$, where S_i is the set of standard indicators in partition B_i, S' is the set of all standard indicators, and α is a weight parameter. All indicators in block B_i share the same score_i.

Since the algorithm allows an indicator to appear in different partitions, an indicator may have multiple different scores and requires calculation of a weighted average based on different partition weights: $\text{score}'(x) = \frac{\sum_{B_i \in \mathcal{B}} \mathbb{I}(x \in B_i) \cdot \text{score}_i}{\sum_{B_i \in \mathcal{B}} \mathbb{I}(x \in B_i)}$. As a preliminary attempt at an indicator standardization algorithm, this paper simply assumes all partitions have equal weights. Specifically, if an indicator is not assigned to any partition, its score is 0. This is why weighted smoothing is applied when calculating partition scores score_i: as long as an indicator can be assigned to a partition, it receives a base score.

2.4 Cluster Redefinition

Intra-cluster binary classification partitions indicators within a cluster into synonymous and non-synonymous indicators of the standard indicator. For non-synonymous indicators, this paper extracts them as a new cluster, selects a new standard name from them, and continues searching for synonymous indicators using the binary classification algorithm, iterating until all indicators within each cluster are synonymous or only one indicator remains in the cluster.

2.5 Indicator Mapping and Correction

At this stage, the indicator standardization algorithm is nearing completion. It only remains to uniformly map synonymous indicators within each cluster to their corresponding standard indicators and have medical professionals verify and correct the alignment results. Specifically, the clustering process may assign synonymous indicators to different clusters, and after the binary classification process removes non-synonymous indicators from clusters, manual verification must also merge synonymous clusters.

3. Experiments

3.1 Dataset

This paper extracts indicator datasets from the Shanghai Clinical Diagnosis and Treatment Information Sharing Platform for experiments. During the indicator data extraction process, this paper considers two factors: first, the variety of indicators must be rich to simulate real-world application scenarios; second, the names of synonymous indicators must be diverse, otherwise indicator standardization would be meaningless. Therefore, this paper extracts all indicators on a per-hospital basis to ensure richness, while selecting the top 8 hospitals with the most diverse indicator names to satisfy diversity requirements.

The number of different indicator names in these 8 hospitals are: 1404, 1243, 1098, 1010, 992, 958, 921, and 849, respectively. After merging and deduplication, there are 5211 different indicator names. After expanding the abbreviation fields for these indicator names, the number of distinct records becomes 7542; after further expanding both abbreviation and reference value fields, the number

of distinct records reaches 12750. In the clustering experiment section, this paper selects 236 data points for evaluation. In the binary classification experiment section, this paper samples positive and negative examples at a 1:1 ratio and splits the sampling results into training and test sets at a 7:3 ratio, ultimately obtaining 947 training samples and 406 test samples. This paper additionally selects 100 positive examples and 100 negative examples as a validation set.

3.2 Experimental Setup

This paper employs grid search on the validation set and uses parameters $minPts = 3$, $\epsilon = 0.35$, threshold $\theta = 0.7$, and $\alpha = 0.6$ for experiments. Gradient Boosting Decision Tree (GBDT) is selected as the final binary classification model, and Precision, Recall, and F1-score are used to evaluate the effectiveness of clustering and binary classification.

3.3 Experimental Results

3.3.1 Clustering Algorithm Comparison To investigate the effectiveness of the density-based clustering algorithm (DBSCAN) used in this paper, we select four common clustering algorithms for comparison: k-means clustering (K-means), mean shift algorithm (Meanshift), Gaussian Mixture Model (GMM), and Agglomerative Hierarchical Clustering (AHC). It should be noted that except for Gaussian Mixture Model, these four baseline algorithms require predefined cluster numbers (whereas our algorithm does not). During experiments, we set their cluster numbers to the true cluster numbers. Experimental results are shown in Table 2 .

Table 2 Comparisons of different clustering algorithms

Clustering Algorithm	Precision	Recall	F1-score
K-means			
Meanshift			
GMM			
AHC			
Our DBSCAN			

The results show that our density-based clustering algorithm' s F1-score is significantly higher than the other four clustering algorithms, with improvements exceeding 10%. However, although our method' s Recall reaches 91.36%, Precision is still not very high, demonstrating the necessity of further binary classification mapping after clustering.

3.3.2 Binary Classification Algorithm Comparison Comparison of different classification features and classifiers. To investigate the impact of different classification features and classifiers on classification performance,

this paper selects different feature combinations and compares their F1-scores under various classifiers including Logistic Regression (LR), Naive Bayes (NB), k-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest (RF), and Gradient Boosting Decision Tree (GBDT). Experimental results are shown in Table 3 , where feature fields Name (name), Abbreviation (Abbr.), and Reference Value (Ref.) represent name similarity features, abbreviation similarity features, and reference value partition scoring features, respectively.

Table 3 Comparisons of different classification algorithms

Features	Abbr.	Name+Abbr.	Name+Ref.	Abbr.+Ref.	Name+Abbr.+Ref.
LR					
NB					
KNN					
SVM					
RF					
GBDT					

The results show that classification performance is best when using name similarity features, abbreviation similarity features, and reference value partition scoring features together with the GBDT classifier, achieving an F1-score of 85.27%. Horizontally, regardless of the features used, GBDT generally performs best while NB performs worst. This is because GBDT uses Boosting for ensemble learning, which can effectively improve generalization performance, whereas the conditional independence assumption of the NB classifier hardly holds in this context. Vertically, regardless of the classifier, performance generally improves as the number of features increases, reaching optimal performance when using all three types of classification features.

Comparison with existing methods. Finally, this paper selects three state-of-the-art methods published in the last three years for comparison with our binary classification method using all three types of features with the GBDT classifier. These three baseline methods are:

- **Knowledge Graph Fusion (KG Fusion)**: Wang et al. [37] designed different types of attribute similarities and used machine learning methods for multi-source knowledge graph fusion.
- **Diagnosis Alignment (Diag. Alignment)**: Ning et al. [38] utilized hypernym-hyponym information and attribute similarity of diagnoses to map Chinese diagnoses to ICD codes.
- **Knowledge Base Alignment (KB Alignment)**: Wang Xuepeng et al. [39] leveraged web semantic tags for entity alignment in multi-source knowledge bases.

It should be noted that since our task has neither attribute information nor contextual information, some features of the three baseline methods cannot

be used in actual experiments, and we mainly utilized their entity name and abbreviation similarity calculation methods.

Comparison results with existing methods are shown in Table 4 . The results demonstrate that our method achieves the best classification performance among all methods, with Precision, Recall, and F1-score of 86.84%, 83.76%, and 85.27%, respectively. Notably, compared with the last column of Table 3, any two-feature combination of our method using the GBDT classifier outperforms existing methods. This is because our algorithm is specifically designed for lab test indicators, thus achieving better results.

Table 4 Performance comparison of entity alignment

Method	Precision	Recall	F1-score
KG Fusion			
Diag. Alignment			
KB Alignment			
Our Method	86.84%	83.76%	85.27%

Conclusion

This paper addresses lab indicator standardization in regional medical health platforms by first clustering based on indicator literal features, then iteratively performing binary classification mapping using similarity features and partition scoring features. Experiments demonstrate that the final binary classification mapping achieves an F1-score of 85.27%, outperforming existing methods. In the future, synonym information and reference value information of indicators can be incorporated into the clustering algorithm, and more similarity measurement features can be attempted to obtain better results.

Acknowledgments

We thank Zhang Haitao from Shanghai University of Traditional Chinese Medicine and Li Yang from Tongji University School of Medicine for their assistance in dataset annotation.

References

[1] Mining W I D. Data Mining: Concepts and Techniques[J]. Morgan Kaufinann, 2006.

[7] Cochinwala M, Kurien V, Lalk G, et al. Efficient data reconciliation[J]. Information Sciences, 2001, 137(1-4): 1-15.

[8] Elfeky M G, Verykios V S, Elmagarmid A K. TAILOR: A record linkage toolbox[C]//Data Engineering, 2002.

[9] Christen P. Automatic training example selection for scalable unsupervised record linkage[C]//Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Berlin, Heidelberg, 2008: 511-518.

[10] Kantardzic M. Data mining: concepts, models, methods, and algorithms[M]. John Wiley & Sons, 2011.

[11] Chen Z, Kalashnikov D V, Mehrotra S. Exploiting context analysis for combining multiple entity resolution systems[C]//Proceedings of the 2009 ACM SIGMOD International Conference on Management of data. ACM, 2009: 207-218.

[12] Cohen W W, Richman J. Learning to match and cluster large high-dimensional data sets for data integration[C]//Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002: 475-480.

[13] McCallum A, Wellner B. Conditional models of identity uncertainty with application to noun coreference[C]//Advances in neural information processing systems. 2005: 905-912.

[14] Pasula H, Marthi B, Milch B, et al. Identity uncertainty and citation matching[C]//Advances in neural information processing systems. 2003: 1425-1432.

[15] Sarawagi S, Bhamidipaty A. Interactive deduplication using active learning[C]//Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002: 269-278.

[16] Tejada S, Knoblock C A, Minton S. Learning domain-independent string transformation weights for high accuracy object identification[C]//Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002: 350-359.

[17] Arasu A, Götz M, Kaushik R. On active learning of record matching packages[C]//Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. ACM, 2010: 783-794.

[18] Bhattacharya I, Getoor L. A latent dirichlet model for unsupervised entity resolution[C]//Proceedings of the 2006 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2006: 47-58.

[19] Hall R, Sutton C, McCallum A. Unsupervised deduplication using cross-field dependencies[C]//Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008: 310-317.

[20] Domingos P. Multi-relational record linkage[C]//In Proceedings of the KDD-2004 Workshop on Multi-Relational Data Mining. 2004.

[21] Singla P, Domingos P. Entity resolution with markov logic[C]//Data Mining, 2006. ICDM' 06. Sixth International Conference on. IEEE, 2006: 572-582.

[22] Rastogi V, Dalvi N, Garofalakis M. Large-scale collective entity matching[J]. Proceedings of the VLDB Endowment, 2011, 4(4): 208-218.

[23] Blanco R, Ottaviano G, Meij E. Fast and space-efficient entity linking for queries[C]//Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. ACM, 2015: 179-188.

[24] Shen W, Wang J, Luo P, et al. Linking named entities in tweets with knowledge base via user interest modeling[C]//Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 68-76.

[25] Han X, Sun L, Zhao J. Collective entity linking in web text: a graph-based method[C]//Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, 2011: 765-774.

[26] Gentile A L, Zhang Z, Xia L, et al. Graph-based semantic relatedness for named entity disambiguation[J]. 2009.

[27] Alhelbawy A, Gaizauskas R. Graph ranking for collective named entity disambiguation[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2014, 2: 75-80.

[28] Hoffart J, Yosef M A, Bordino I, et al. Robust disambiguation of named entities in text[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011: 782-792.

[29] He Z, Liu S, Li M, et al. Learning entity representation for entity disambiguation[C]//Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2013, 2: 30-34.

[30] Huang H, Heck L, Ji H. Leveraging deep neural networks and knowledge graphs for entity disambiguation[J]. arXiv preprint arXiv:1504.07678, 2015.

[31] Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-621.

[32] Collobert R, Weston J. A unified architecture for natural language processing: Deep neural networks with multitask learning[C]//Proceedings of the 25th international conference on Machine learning. ACM, 2008: 160-167.

[33] Ester M, Kriegel H P, Xu X. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise[C]//International Conference on Knowledge Discovery and Data Mining. AAAI Press, 1996:226-231.

[34] Donnelly K. SNOMED-CT: The advanced terminology and coding system for eHealth.[J]. Studies in Health Technology & Informatics, 2006, 121(121):279.

[35] Mcdonald C J, Huff S M, Suico J G, et al. LOINC, a universal standard for identifying laboratory observations: a 5-year update.[J]. Clinical Chemistry,

2003, 49(4):624.

[36] Zhuang Y, Li G, Zhong Z, et al. Hike: A Hybrid Human-Machine Method for Entity Alignment in Large-Scale Knowledge Bases[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, 2017: 1917-1926.

[37] Wang H, Fang Z, Zhang L, et al. Effective online knowledge graph fusion[C]//International Semantic Web Conference. Springer, Cham, 2015: 1-16.

[38] Ning W, Yu M, Zhang R. A hierarchical method to automatically encode Chinese diagnoses through semantic similarity estimation[J]. BMC medical informatics and decision making, 2016, 16(1): 30.

[39] X.-P. Wang, K. Liu, S.-Z. He, S.-L. Liu, Y.-Z. Zhang, and J. Zhao, "Multi-source knowledge bases entity alignment by leveraging semantic tags," Jisuanji Xuebao/Chinese Journal of Computers, vol. 40, no. 3, pp. 701 –711, 2017.(in Chinese) (Wang Xuepeng, Liu Kang, He Shizhu, et al. Multi-source knowledge base entity alignment algorithm based on web semantic tags[J]. Chinese Journal of Computers, 2017, 40(3): 701-711.)

Note: Figure translations are in progress. See original paper for figures.