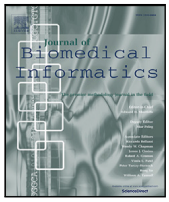
Journal of Biomedical Informatics 133 (2022) 104147

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin



Original Research

Multiview Incomplete Knowledge Graph Integration with application to cross-institutional EHR data harmonization

Doudou Zhou [a,1], Ziming Gan [b,1], Xu Shi [c], Alina Patwari [d], Everett Rush [e], Clara-Lea Bonzel [f,g], Vidul A. Panickan [f,g], Chuan Hong [g,h], Yuk-Lam Ho [g], Tianrun Cai [g,i], Lauren Costa [g], Xiaoou Li [j], Victor M. Castro [k], Shawn N. Murphy [k], Gabriel Brat [f,l], Griffin Weber [f], Paul Avillach [f], J. Michael Gaziano [f,g,i], Kelly Cho [f,g,i], Katherine P. Liao [g,i], Junwei Lu [g,m,1], Tianxi Cai [f,g,m,*,1]

[a] University of California, Davis, CA, USA
[b] University of Chicago, Chicago, IL, USA
[c] University of Michigan, MI, USA
[d] Cambridge Rindge & Latin School, Cambridge, MA, USA
[e] Department of Energy, Oak Ridge National Lab, Oak Ridge, TN, USA
[f] Harvard Medical School, Boston, MA, USA
[g] VA Boston Healthcare System, Boston, MA, USA
[h] Duke University, Durham, NC, USA
[i] Brigham and Women's Hospital, Boston, MA, USA
[j] University of Minnesota, MN, USA
[k] Mass General Brigham, Somerville, MA, USA
[l] Beth Israel Deaconess Medical Center, Boston, MA, USA
[m] Harvard T.H. Chan School of Public Health, Boston, MA, USA

ARTICLE INFO

Keywords:
Code mapping
PMI matrix
Word embedding
Transfer learning
Knowledge graph

ABSTRACT

Objective: The growing availability of electronic health records (EHR) data opens opportunities for integrative analysis of multi-institutional EHR to produce generalizable knowledge. A key barrier to such integrative analyses is the lack of semantic interoperability across different institutions due to coding differences. We propose a **M**ultiview **I**ncomplete **K**nowledge **G**raph **I**ntegration (MIKGI) algorithm to integrate information from multiple sources with partially overlapping EHR concept codes to enable translations between healthcare systems.

Methods: The MIKGI algorithm combines knowledge graph information from (i) embeddings trained from the co-occurrence patterns of medical codes within each EHR system and (ii) semantic embeddings of the textual strings of all medical codes obtained from the Self-Aligning Pretrained BERT (SAPBERT) algorithm. Due to the heterogeneity in the coding across healthcare systems, each EHR source provides partial coverage of the available codes. MIKGI synthesizes the incomplete knowledge graphs derived from these multi-source embeddings by minimizing a spherical loss function that combines the pairwise directional similarities of embeddings computed from all available sources. MIKGI outputs harmonized semantic embedding vectors for all EHR codes, which improves the quality of the embeddings and enables direct assessment of both similarity and relatedness between any pair of codes from multiple healthcare systems.

Results: With EHR co-occurrence data from Veteran Affairs (VA) healthcare and Mass General Brigham (MGB), MIKGI algorithm produces high quality embeddings for a variety of downstream tasks including detecting known similar or related entity pairs and mapping VA local codes to the relevant EHR codes used at MGB. Based on the cosine similarity of the MIKGI trained embeddings, the AUC was 0.918 for detecting similar entity pairs and 0.809 for detecting related pairs. For cross-institutional medical code mapping, the top 1 and top 5 accuracy were 91.0% and 97.5% when mapping medication codes at VA to RxNorm medication codes at MGB; 59.1% and 75.8% when mapping VA local laboratory codes to LOINC hierarchy. When trained with 500 labels, the lab code mapping attained top 1 and 5 accuracy at 77.7% and 87.9%. MIKGI also attained best performance in selecting VA local lab codes for desired laboratory tests and COVID-19 related features

* Corresponding author at: Harvard Medical School, Boston, MA, USA.
E-mail address: tcai@hsph.harvard.edu (T. Cai).
[1] Contributed equally.

https://doi.org/10.1016/j.jbi.2022.104147
Received 31 January 2022; Received in revised form 5 July 2022; Accepted 15 July 2022
Available online 21 July 2022

for COVID EHR studies. Compared to existing methods, MIKGI attained the most robust performance with accuracy the highest or near the highest across all tasks.

Conclusions: The proposed MIKGI algorithm can effectively integrate incomplete summary data from biomedical text and EHR data to generate harmonized embeddings for EHR codes for knowledge graph modeling and cross-institutional translation of EHR codes.

1. Introduction

The adoption of electronic health record (EHR) systems has not only changed clinical practice, but also expanded the breadth of biomedical research, providing a myriad of opportunities for clinical research such as predicting disease diagnosis [1–3], comparing treatment options [4,5], extracting medical features, and representing medical concepts [6–8]. A key barrier to EHR-based multi-institutional research is the lack of interoperability across healthcare systems [3,9,10]. EHR data harmonization, which is a process of standardizing definitions for core data elements from a variety of sources [11–13], has been recognized as a critical step to reduce heterogeneity in data elements and improve reproducibility of research [14,15].

To harmonize EHR data across healthcare systems, a standard practice is to employ a common data model, such as the Observational Medical Outcomes Partnership (OMOP) [16] and Patient-Centered Outcomes Research network (PCORnet) [17], which organizes data into a standard structure, maps data elements into a common format, and standardizes vocabularies into the same ontology [18,19]. Although the widespread adoption of common data models has improved interoperability, EHR data harmonization remains challenging for several reasons.

First, although the common data model can standardize coding vocabulary, heterogeneity remains due to differential coding practices and financial incentives between healthcare systems. With the increasing diversity and specificity of coding systems, there is more and more potential variation in the way a clinical concept can be coded even within the same vocabulary. In fact, it is often observed in practice that the same clinical feature might be represented by distinct codes at different healthcare systems [20,21].

Second, existing ontologies such as the International Classification of Diseases, Ninth Revision (ICD-9) [22] and Tenth Revision (ICD-10) [23], RxNorm medication codes [24], Current Procedural Terminology (CPT) [25], and Logical Observation Identifiers Names and Codes (LOINC) [26], are only partially useful since most healthcare systems adopted some but not all of the ontologies. Mappings from local codes to standardized ontologies tend to be incomplete, ambiguous [21] and sometimes inaccurate [27]. In addition, these ontologies are constantly updated over time. Mapping between coding systems, such as the Generalized Equivalence Mapping (GEM) for ICD-9 to ICD-10 mapping [28], comes with complicated relationships such as one-to-many and many-to-many mapping, which further complicates vocabulary standardization.

Third, the process of code mapping generally requires some level of manual effort and relies on clinical and informatics domain knowledge [29–31], which is time-consuming, resource-intensive, and particularly hard to scale for projects with large amounts of codes [29,32]. Manual curation is also susceptible to subjective bias and human errors. Recently, automated methods such as open access mapping tools [33–36] and corpus or lexical based algorithms [20,37–41] have been explored for mapping terminologies in diverse domains [20,37,38]. However, these methods may rely on domain knowledge with the requirement of golden labels; the methods are not data-driven in the sense that they do not fully utilize the rich information from routinely collected EHR data to learn the relationship among medical codes. In addition, a majority of the methods are limited to one type of medical codes, such as drug or lab codes.

In recent years, automated code mapping algorithms have evolved around a technique in natural language processing (NLP) termed word embedding [42–44]. Word embeddings are numeric vectors that capture the meaning of words, such that embeddings of synonyms have smaller distances. Such semantic representations have been widely used to translate words between two languages [45]. By analogy to language translation based on word embeddings, one can achieve code translation based on code embeddings that represent the similarity and relatedness of medical codes. That is, one can train embeddings of 'words', which are medical codes in the context of EHR data harmonization, then map them between two 'languages', which are the use of codes in two healthcare institutions with distinct coding practices. Existing methods include knowledge graph (KG) [46–49], neural network based methods such as the skip-gram algorithm [44,50–52], and matrix factorization [6,53–55]. KG embedding methods translate components of a KG, which are essentially entities and their relations, to lower dimensional embedding vectors [56]. The code mapping problem can then be viewed as a fundamental link prediction task [57–59] to infer whether two codes from different institutions have similar meanings. Neural network based methods typically train a shallow neural network to predict codes or context of codes, with embeddings being a byproduct extracted from a hidden layer. To improve downstream code translation, the embedding vector spaces are first aligned across institutions [60–63]. A dictionary that maps codes between two institutions is then developed by finding nearest neighbors of a code with distances typically measured via cosine similarity [28,45,64]. Matrix factorization based method takes as input a shifted positive pointwise mutual information (SPPMI) matrix computed from medical codes' co-occurrence patterns in patient health records, and computes code embeddings via a singular vector decomposition (SVD) [65]. This method has been shown to have similar performance as the skip-gram algorithm [53]. It is advantageous because embeddings are trained based on population-level summaries of EHR data which breaks data sharing barriers and offers scalability.

The above methods have several limitations. KG embedding methods require input data to be structured as triples of the form *(head entity, relation, tail entity)*. Thus, KG embeddings are typically trained from well structured knowledge databases such as the Unified Medical Language System (UMLS) [66], which contains relational information between entity pairs. On the other hand, free form unlabeled EHR data cannot be easily turned into such triplets for KG training. Neural-network-based embeddings are often trained on patient-level EHR data, which poses significant administrative challenges when data sharing across research groups and institutions is not feasible due to privacy concerns. Matrix factorization-based embedding algorithms trained with EHR data does not incorporate information from code textual descriptions. Since code descriptions provide important information about the interpretation and relationship of medical codes, the embeddings trained with only EHR data tend to have limited accuracy in cross-institutional code mapping as demonstrated in the validation studies below.

In this paper, we propose the **M**ultiview **I**ncomplete **K**nowledge **G**raph **I**ntegration (MIKGI) algorithm to co-train embeddings for EHR codes from multiple institutions by combining both EHR co-occurrence information and textual information from the code descriptions. We leverage the Self-Aligning Pretrained Bidirectional Encoder Representations from Transformers (SAPBERT) algorithm proposed by [67] to extract semantic information from code descriptions. By combining the pre-trained language model with contrastive representation learning using synonyms in the MRCONSO table of the UMLS, the SAPBERT algorithm achieves state-of-the-art performance for the synonymous medical entity linking task. For extracting information from

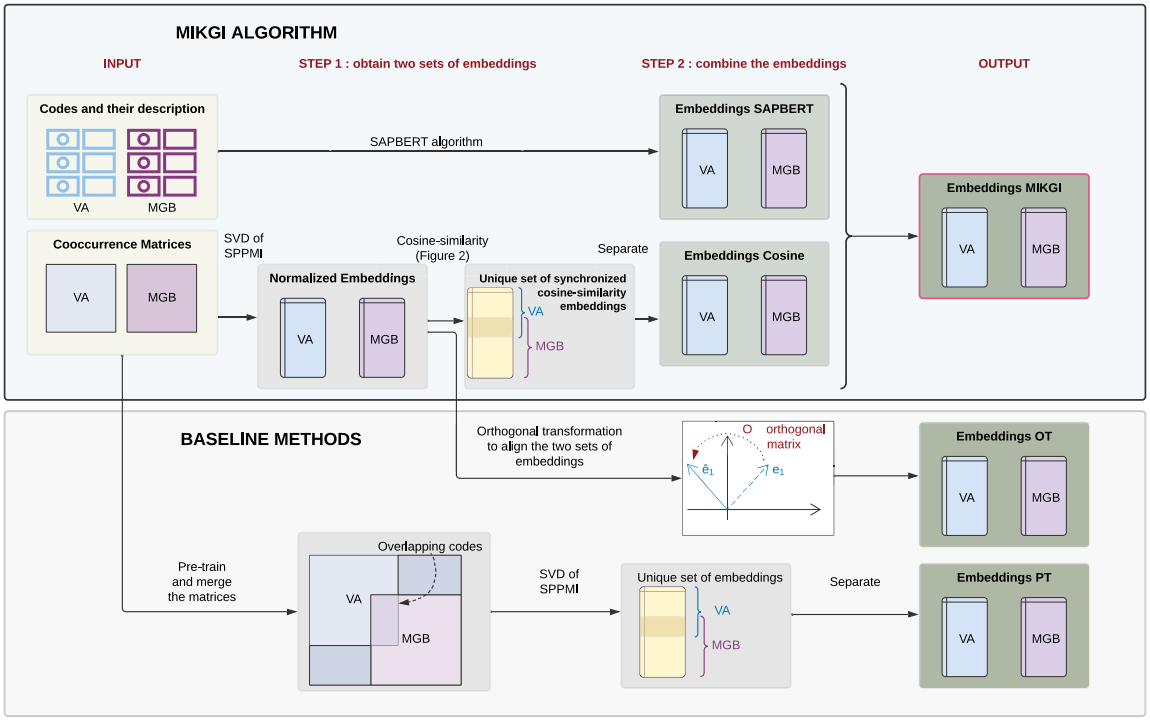


Fig. 1. Illustration of MIKGI and other relevant methods.

the co-occurrence matrix, we utilize the SPPMI-SVD algorithm. We hypothesize that integrating information from SAPBERT with SPPMI-SVD can improve the quality of embeddings significantly. Our rationale is that SAPBERT embeddings provide information on code similarity based on the textual description of codes, while SPPMI-SVD captures code relationships based on the co-occurrence of codes. The latter is an effective supplement since diseases and related treatments may have completely different descriptions not captured by SAPBERT. Our method combines the information in two types of embeddings properly to reveal the code relationships in the same or different coding systems. We choose hyperparameters to ensure the generalizability of the embeddings while allowing users to tune hyperparameters with a user-specified loss function for their specific tasks. Finally, we validate our method in four tasks in comparison to existing methods.

2. Materials and methods

The proposed MIKGI algorithm derives high quality embeddings for EHR codes from two institutions by integrating information from the text description of the codes and co-occurrence patterns of EHR codes within each institution, leveraging the shared codes as anchors. As illustrated in Fig. 1, the MIKGI algorithm consists of two key steps: (1) generating two initial sets of embeddings, one based on the text descriptions of all codes via the SAPBERT algorithm and the other one based on co-occurrence patterns of the EHR data, and (2) integrating the multi-source embeddings into the final MIKGI embeddings.

2.1. Step 1: Generating initial embeddings from multiple sources

2.1.1. Generating code embeddings based on text description of medical codes

We first generate one set of embeddings for all codes, denoted by $\mathcal{N}$, based on their textual descriptions via the SAPBERT algorithm.

Since some codes are shared between the two institutions and hence have the same descriptions, SAPBERT embeddings would generate distinct embeddings for codes that are unique to two institutions but the same embeddings for codes that shared by the institutions. We let $\mathbf{U} = (\mathbf{U}_{11}^{\mathsf{T}}, \mathbf{U}_{12}^{\mathsf{T}}, \mathbf{U}_{22}^{\mathsf{T}})^{\mathsf{T}} \in \mathbb{R}^{(n_{11}+n_{12}+n_{22})\times r_1}$ denote the r_1-dimensional code embeddings for all codes in $\mathcal{N} = \mathcal{N}_{11} \cup \mathcal{N}_{12} \cup \mathcal{N}_{22}$ trained from the SAPBERT algorithm, referred to as SAPBERT embeddings, where $\mathbf{U}_{11}$, $\mathbf{U}_{12}$, and $\mathbf{U}_{22}$ are the respective normalized unit-length embeddings for codes in $\mathcal{N}_{11}$ (codes unique to institution 1), $\mathcal{N}_{12}$ (codes shared by the institutions), and $\mathcal{N}_{22}$ (codes unique to institution 2), and $n_{mm'} = \text{Card}(\mathcal{N}_{mm'})$.

2.1.2. Generating embeddings based on EHR code co-occurrence patterns

To generate embeddings for EHR codes based on their co-occurrence patterns, we first obtain two initial sets of embeddings, one for $\mathcal{N}_1 = \mathcal{N}_{11} \cup \mathcal{N}_{12}$ and one for $\mathcal{N}_2 = \mathcal{N}_{12} \cup \mathcal{N}_{22}$, based on the pairwise co-occurrence counts of the codes within each institution. Second, we synchronize the two sets of embeddings by generating an alternative representation of all codes based on their distance from the n_{12} overlapping codes. Specifically, we construct the co-occurrence matrix of codes in $\mathcal{N}_m$, $C_m = [C_m(i,j)]$, following [6] such that $C_m(i,j)$ is the total co-occurrence the ith code and the jth code within 30-day moving windows of a patient's health record across all patients. For the mth institution, the (i,j)th entry of the SPPMI matrix is obtained as

$$\mathbb{SPPMI}_m(i,j) = \max\left\{0, \log\frac{C_m(i,j)}{C_m(i,\cdot)C_m(j,\cdot)} - \log(k)\right\},$$

where k is the negative sample which is often set as 1 (i.e., no shifting), and $C_m(i,\cdot)$ is the row sum of $C_m(i,j)$. The SPPMI-SVD algorithm generates embeddings for the mth institution by taking an SVD of the SPPMI matrix as $\mathbb{SPPMI}_m = [\mathbb{SPPMI}_m(i,j)] = \mathbb{U}_m \text{diag}(\Lambda_{m,1}, \ldots, \Lambda_{m,n_m})\mathbb{U}_m^{\mathsf{T}}$ and letting

$$\mathbf{X}_m = \mathscr{R}\left\{\mathbb{U}_m^{(r_2)} \text{diag}\left(\Lambda_{m,1}^{\frac{1}{2}}, \ldots, \Lambda_{m,r_2}^{\frac{1}{2}}\right)\right\},$$

where $\mathbb{U}_m^{(r_2)}$ is the first r_2 singular vectors with positive eigenvalues, $\mathscr{R}(\cdot) : \mathbb{M} \to \mathscr{R}(\mathbb{M})$ is the unit-length normalization operator that standardizes each row of $\mathbb{M}$ to have unit length, and the dimension r_2 can be selected to optimize embedding quality as detailed in the validation studies. This scalable variant of the skip-gram algorithm has been shown to be effective in generating high quality embeddings

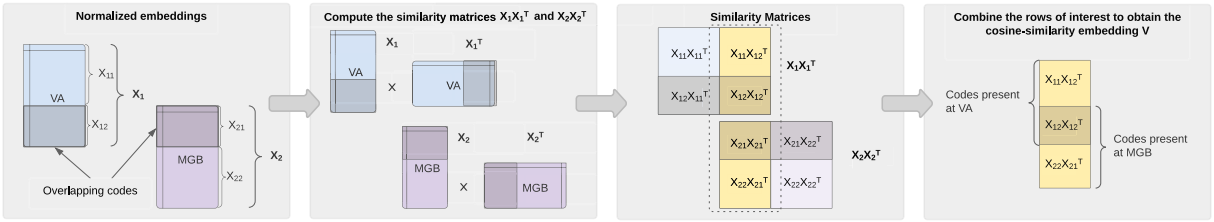


Fig. 2. Embedding obtained from the similarity matrices.

for EHR codes [6,55]. We write $\mathbf{X}_1 = (\mathbf{X}_{11}^\top, \mathbf{X}_{12}^\top)^\top \in \mathbb{R}^{(n_{11}+n_{12})\times r_2}$ and $\mathbf{X}_2 = (\mathbf{X}_{21}^\top, \mathbf{X}_{22}^\top)^\top \in \mathbb{R}^{(n_{12}+n_{22})\times r_2}$, where $\mathbf{X}_{12}$ and $\mathbf{X}_{21} \in \mathbb{R}^{n_{12}\times r_2}$ correspond to $\mathcal{N}_{12}$, the overlapping codes in the two institutions.

We next generate an alternative EHR embedding vector for each EHR code based on its cosine similarity with the n_{12} overlapping codes, which can be viewed as anchor codes. As illustrated in Fig. 2, we compute the cosine-similarity matrices $\widetilde{\mathbf{V}}_{11} = \mathbf{X}_{11}\mathbf{X}_{12}^\top$, $\widetilde{\mathbf{V}}_{12} = \mathbf{X}_{12}\mathbf{X}_{12}^\top$, and $\widetilde{\mathbf{V}}_{22} = \mathbf{X}_{22}\mathbf{X}_{21}^\top$, where each row of these matrices represents the distance between a candidate code and the n_{12} anchor codes. The final set of synchronized cosine-similarity (SynC) based embeddings is obtained as $\mathbf{V} = (\mathbf{V}_{11}^\top, \mathbf{V}_{12}^\top, \mathbf{V}_{22}^\top)^\top \in \mathbb{R}^{n\times n_{12}}$, where $\mathbf{V}_{mm'} = \mathscr{R}(\widetilde{\mathbf{V}}_{mm'})$. These SynC embeddings integrate information from the EHR co-occurrence patterns from two institutions and will be further combined with the SAPBERT embeddings as detailed in Section 2.2.

2.2. Step 2: Generating MIKGI embeddings

In the second step, we generate MIKGI embeddings $\mathbf{W} = (\mathbf{W}_{11}^\top, \mathbf{W}_{12}^\top, \mathbf{W}_{22}^\top)^\top \in \mathbb{R}^{n\times r}$ by further integrating information from the code descriptions and co-occurrence patterns. We achieve this via consensus learning with $\mathbf{W}\mathbf{W}^\top$ obtained as low-rank approximations to both the pairwise cosine similarity matrices defined by SAPBERT embeddings and by SynC embeddings. Specifically, we generate the final MIGKI embeddings $\mathbf{W} \in \mathbb{R}^{n\times r}$ by solving a constrained minimization problem

$$\min_{\mathbf{W}\in\mathbb{R}^{n\times r}, \mathrm{diag}(\mathbf{W}\mathbf{W}^\top)=\mathbf{1}} f(\mathbf{W}), \tag{1}$$

where

$$f(\mathbf{W}) = \left\|\mathbf{W}\mathbf{W}^\top - \mathbf{U}\mathbf{U}^\top\right\|_{\mathrm{F}}^2 + \lambda\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{V}\mathbf{V}^\top\right\|_{\mathrm{F}}^2. \tag{2}$$

The first component of the loss function, $\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{U}\mathbf{U}^\top\right\|_{\mathrm{F}}^2$, leverages information from textual descriptions, while the second component, $\lambda\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{V}\mathbf{V}^\top\right\|_{\mathrm{F}}^2$, incorporates information from the co-occurrence patterns. The tuning parameter λ allows the multiple sources of information to contribute differently, which can be tuned for specific down-stream tasks, as detailed in Section 2.2.1. The constraint $\mathrm{diag}(\mathbf{W}\mathbf{W}^\top) = \mathbf{1}$ is imposed to reflect the cosine similarity as the distance metric.

Since $f(\mathbf{W})$ is non-convex, we develop a projected gradient descent algorithm (PGD) to solve for $\mathbf{W}$ in (1). With a user-specified initial value, we iterate over the following two steps until a stopping condition is met: (1) update the $\mathbf{W}$ towards descent direction; (2) normalize each row of the $\mathbf{W}$. Details of the PGD algorithm are given in Appendix S.2.

2.2.1. Tuning parameters

There are four hyper-parameters: the SAPBERT embedding dimension r_1, the initial EHR embedding dimension r_2, the regularization coefficient λ, and the dimension r of $\mathbf{W}$ in the MIKGI algorithm loss $f(\mathbf{W})$. We follow [68] to select the dimension of the SAPBERT embedding r_1. For r_2, we rely on the eigen decay of the two SPPMI matrices and choose r_2 as the smallest L that explains 80% of variation in that $\{\sum_{m=1}^2\sum_{l=1}^L \Lambda_{m,l}\}/\{\{\sum_{m=1}^2\sum_{l=1}^{n_m}\Lambda_{m,l}\}\} \geq 0.8$ [55,69]. For simplicity, we use the same r_2 for both embeddings although different dimensions can be used for $\mathbf{X}_1$ and $\mathbf{X}_2$. Finally, for different down-stream tasks, we tune λ and r using a small set of golden labels. To be specific, for the task of detecting similar or related codes, we tune λ and r by maximizing the accuracy of the trained embeddings $\mathbf{W}$ in detecting known relationship pairs as detailed in Section 3.2.1. These known relationship pairs were curated from a range of knowledge sources and have been mapped to EHR codes as described in [55]. We randomly selected 20% of the relationship pairs for a training set to tune $\{\lambda, r\}$, and the remaining 80% were used to evaluate the quality of the embeddings. In addition to the relation pairs from the training set, we randomly selected an equal number of random pairs as controls. For $\mathbf{W}$ trained with a given $\{\lambda, r\}$, we calculate the area under the receiver operating characteristic curve (AUC) of the cosine similarity based on $\mathbf{W}$ in distinguishing relation pairs from random pairs. The final $\{\lambda, r\}$ are selected to maximize the AUC. For the task of mapping synonymous medication codes across the two institutions, we utilize the drug–drug hierarchy detailed in Section 3.2.1 to select $\{\lambda, r\}$ that maximize the AUC of detecting the drug–drug similar pairs. For the mapping of local lab codes to LOINC codes, we randomly sample 50 pairs of manually mapped local lab codes to LOINC codes and choose the parameters that maximize the acc@1 with details in Section 3.2.2.

3. Validation of MIKGI using real-world EHR data

3.1. EHR data sources and preprocessing

We validate the performance of the MIKGI algorithm using real world EHR data from two large hospital systems, the Veterans Affairs (VA) Healthcare System and the Mass General Brigham (MGB) [70]. The VA Corporate Data Warehouse (CDW) aggregates EHR data from 23 million unique individuals (1999 – 2019) over 150 VA facilities into a single data warehouse. A total of 12.6 million patients with inpatient and outpatient codified data from at least 1 visit were included for this analysis. The MGB EHR data contains codified information on diagnoses, medications, procedures, and laboratories from 2.5 million patients with at least 3 visits spanning more than 30 days. The analysis included coded data from all inpatient, outpatient and emergency department visits of these patients between 1998 and 2018.

We gathered four domains of codified data including ICD diagnosis codes, procedures, lab tests, and medications prescriptions. All ICD codes were aggregated into PheCodes using the ICD-to-PheCode mapping from PheWAS catalog (https://phewascatalog.org/phecodes). All procedure codes except for medications, including CPT-4, HCPCS, ICD-9-PCS, ICD-10-PCS, are grouped into clinical classification software (CCS) categories based on the CCS mapping (https://www.hcup-us.ahrq.gov/toolssoftware/ccs_svcsproc/ccssvcproc.jsp). For medication codes, all local medication codes at MGB have been aggregated and rolled up into ingredient level RxNorm codes. At VA, a majority of medication codes have been previously mapped to RxNorm codes but 199 medication codes are not mapped. We pre-processed the VA local medication codes by removing dose information and manually mapped those codes to RxNorm codes. These manual mappings are used only for validating the MIKGI algorithm, not as part of the overlapping codes. All MGB laboratory codes have been mapped to LOINC codes while only 16.7% of VA laboratory codes have been mapped to LOINC codes or higher level lab concepts such as leukocytes and platelets. In addition,

Table 1
Number of PheCodes, CCS codes, RxNorm codes, LOINC codes, LP codes ShortName codes and local lab codes in the two institutions.

	PheCodes	CCS	RxNorm	LOINC	LOINC:LP	ShortName	Local lab
MGB only	74	20	511	2068	3624	0	0
VA only	78	1	745	204	171	94	2367
overlap	1698	223	724	178	481	0	0

mapping of local laboratory codes to LOINC codes can be ambiguous since multiple LOINC codes can be appropriate. To reduce ambiguity, we further leverage the LOINC Multiaxial Hierarchy, which has a tree structure with higher level LOINC Parts codes (LP codes) that reflect broader laboratory code concepts [71]. For instance, as shown in Fig. 3, code 'LOINC:5821-4' (leukocytes in urine sediment by microscopy high power field) is rolled up to the first level LP code 'LP402498-2' (leukocytes | urine sediment | urinalysis) and the second level code 'LP14419-3' (leukocytes). We include these LP codes that are either parent or grandparent nodes of existing base LOINC codes as additional EHR entities when creating the EHR co-occurrence matrices and the SAPBERT embeddings. The co-occurrence of an LP code x with any EHR code c is calculated by summing over the co-occurrence between any leaf LOINC codes under x and c. SAPBERT embeddings are also generated for LP codes based on their descriptions. All PheCodes, CCS codes, RxNorm codes, LOINC codes, local medication and laboratory codes with frequency lower than 5000 in VA or 1000 in MGB were removed to reduce the noise from the rare codes following [55]. All local laboratory codes with identical descriptions were merged and treated as the same code. This results in a total of 9601 codes at MGB and 6964 codes at VA, with 3304 overlapping codes with details in Table 1.

The code descriptions, used as the input to SAPBERT, were obtained from either existing ontologies for codes have been mapped or local institutions for local codes. For instance, LOINC:10335-8, has the description 'color of cerebral spinal fluid' and the local VA lab code '1200087498' has the description 'calcium'. An ideal code description should fully describe the meaning of the code, thus containing enough information, to generate high-quality embeddings from SAPBERT. However, many local codes have very short and ambiguous descriptions, for example, the description of the local lab code '1000019610' is 'ig #', which cannot describe the code fully. Commonly used codes with less clear descriptions may have lower quality SAPBERT embeddings and can be better represented by MIKGI embeddings.

3.2. Validation analyses

The performance of the MIKGI algorithm was validated in four tasks: (1) detecting known similar or related clinical concepts; (2) mapping synonymous medication and laboratory codes across MGB and VA; (3) identifying relevant VA local lab codes for 21 laboratory tests for a COVID-19 study by the 4CE international Consortium for Clinical Characterization of COVID-19 by EHR [72]; and (4) identifying EHR features important for COVID-19. Most evaluations are based on unsupervised classifications using the cosine similarities between the embedding vectors associated with relevant concept pairs. For cross-institutional lab code mapping, we also examined the performance based on supervised training with learned embeddings as input features.

As benchmark comparisons, we compare MIKGI embeddings with (i) embeddings from SAPBERT [67], BioBERT [73], PubmedBert [74] and CODER [49] based on textual descriptions of the codes; (ii) embeddings derived from the VA and MGB SPPMI matrices based on the orthogonal transformation (OT) [64] and the pre-training (PT). OT aligns the SPPMI-SVD embeddings derived from the two institutions separately by an orthogonal rotation matrix using the overlapping codes. PT merges the co-occurrence matrices to obtain the complete SPPMI matrix, which combined the information of two institutions with details in Appendix S.1 . The hyper-parameters of the two methods (e.g., the embedding dimension) are selected using the same procedure in Section 2.2.1.

Table 2
Number of curated relationship pairs categorized by seven categories.

Relation type	Relation category	# Code pairs
Similarity	RxNorm-RxNorm (Drug-Drug)	2258
	PheCode Hierarchy	3506
	Lab-Lab	2177
Relatedness	PheCode-PheCode (Disease-Disease)	1996
	PheCode-RxNorm (Disease-Drug)	4646
	PheCode-CCS (Disease-Procedure)	2093
	PheCode-Lab (Disease-Lab)	595

3.2.1. Detecting similar or related concepts

We first evaluate the quality of the learned embeddings based on their ability to detect known related or similar pairs of clinical entities. The entities (e.g., drugs) have been mapped to EHR codes such that related pairs are represented by EHR code pairs. Similar code pairs refer to pairs of codes that represent highly similar clinical concepts according to existing ontologies; while related code pairs refer to pairs of codes that have more complex relation like 'may cause' or 'may treat'. For similarity evaluation, we leverage several ontologies's hierarchy to define similar pairs, including PheCode for disease–disease pairs, the LOINC Multiaxial Hierarchy for lab–lab pairs, as well as drug–drug pairs from the [2]ATC classification system in the UMLS. In addition, we have 916 similar lab–lab pairs manually curated by domain experts. For relatedness, we curated known relationship pairs from online knowledge sources including related disease–disease pairs from Wikipedia, disease–drug pairs from *https://www.drugs.com/* [75] and MEDRT [76], and disease–lab pairs from the UMLS. As shown in Table 2, we stratify these relationship pairs into seven categories, with 7941 similar pairs of codes and 9330 related pairs. For each type of relationship, we calculate the cosine similarities of the embedding vectors of known pairs and those of randomly selected pairs. Finally, we calculate the AUC of the cosine similarities in distinguishing known pairs from random pairs. We summarize the overall performance by averaging over different similarity and relatedness categories .

3.2.2. Code mapping across two institutions

We next evaluate the quality of learned embeddings for the task of mapping synonymous medication and laboratory codes across MGB and VA, including mapping (i) local VA medication codes to RxNorm codes; and (ii) local VA lab codes to MGB LOINC and/or LP codes. For medications, at VA there are a list of medications that are not harmonized and coded in the format of local medications, while at MGB, all the medications have been mapped to RxNorm. For laboratory tests, there are a total of 2246 lab codes at MGB, all mapped to LOINC codes, and 2843 lab codes at VA, out of which only 16.7% have been mapped to LOINC codes or higher level lab concepts manually curated by VA (e.g., 'CRP' for all C-reactive Protein measures). To map each VA local code x using embeddings, we identify the codes from the same category (e.g., medications) with highest cosine similarities with x as potential matches. We then evaluate the accuracy against manually curated labels: (i) 199 local VA medication codes to RxNorm codes, and (ii) 1897 local VA lab codes to LOINC or LP codes. Since a local lab code can potentially be mapped to multiple LOINC codes and most clinical studies only concern broader lab concepts (e.g., leukocytes), we assess the matching accuracy based on whether the mapped LOINC

[2] https://www.nlm.nih.gov/research/umls/rxnorm/sourcereleasedocs/atc.html.

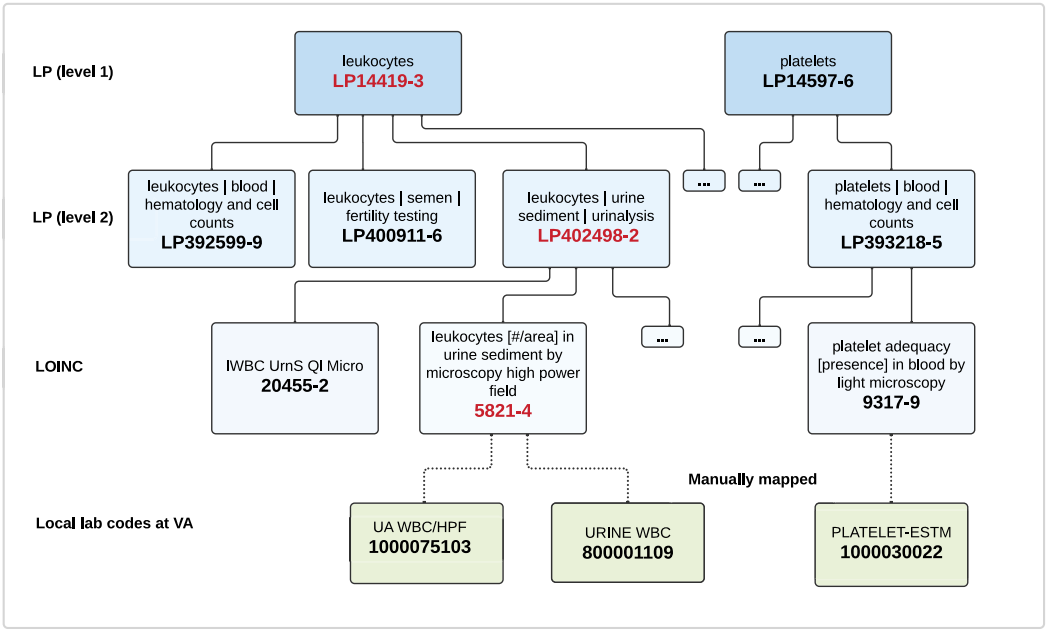


Fig. 3. Lab codes rolling up hierarchy.

and the gold standard labels share the same LP code. We evaluate the code mapping accuracy based on the top k accuracy (acc@k). Denote the top k codes from LOINC and LP code set that are closest in cosine similarity distance to the given local lab code x as $\mathbf{y}_x$. All the LOINC and LP codes that share the same LP code with the gold standard labels are denoted as $\mathbf{z}_x$. The acc@k is defined as the ratio of the number of codes x that $\mathbf{y}_x \cap \mathbf{z}_x \neq \emptyset$ and the total number of codes.

For VA→MGB lab code mapping, we additionally trained supervised algorithms based on the standard orthogonal transform algorithm [60] with the learned embeddings of 450 labeled pairs as input along with 50 labeled pairs for the tuning of hyper-parameters (see Section 2.2.1). We also compared to a maximum entropy (Max) based supervised algorithm for code mapping [39]. The Max algorithm uses the normalized local lab descriptions as input features by generating a set of tokens using lexical rules and the corresponding LOINC codes of each local codes as labels. For example, the local term description 'CSF CELL COUNT/DIFF' is tokenized to generate four tokens 'csf', 'cell', 'count', 'diff', which are then treated as one-hot vectors. Max can only map lab codes to LOINC codes that already exist in the training set and hence has limited generalizability. For supervised algorithms, acc@k is calculated using the curated pairs excluding the 500 training pairs.

3.2.3. Case study of selecting VA local lab codes for a COVID-19 EHR study

We further validated the performance of MIKGI in an EHR study of predicting COVID-19 mortality by the Consortium for Clinical Characterization of COVID-19 by EHR (4CE) international COVID-19 consortium for which VA is a contributing site [72]. We selected relevant VA local lab codes for 21 laboratory tests required in the study, including C-reactive protein (CRP), Albumin, white blood cell count, and D-dimer. We manually curated all VA lab codes and MGB LOINC and LP codes that should be mapped to these 21 laboratory tests, denoted by $\{G_i, i = 1, \ldots, 21\}$. VA lab codes include manually curated lab concepts, local lab codes, and LOINC codes. The manually curated lab concepts, such as CRP, are well annotated with accurate descriptions and can be easily identified even via simple string match. We thus only consider the task of identifying local VA lab codes and LOINC codes for each G_i. Let $\mathbf{x}_i = \{x_{ij}, j = 1, \ldots, n_i\}$ denote all such lab codes including curated lab concepts for the ith test, let $\mathbf{x}*_i = \{x *_{ij}, j = 1, \ldots, n *_i\}$ denote all such lab codes except curated lab concepts for the ith test and let $y_i = \{y_{il}, l = 1, \ldots, m_i\}$ denote all LOINC and LP codes in MGB that should be mapped to G_i. We perform mapping for lab G_i by identifying all VA codes whose top k LOINC or LP code matches (Match@k) include any code in $\mathbf{x}_i$, with $k = 1, 5, 10$ and 20. For each VA lab code x, we let $\hat{y}_x$ denote the top k LOINC or LP codes that x is mapped to. Then we evaluate the accuracy of the MIKGI based mapping by calculating sensitivity and specificity of the mapping defined as

$$\text{specificity} = 1 - \left(\sum_{i=1}^{21} \frac{\sum_{x \notin \mathbf{x}_i} I(\hat{y}_x \cap \mathbf{y}_i \neq \emptyset)}{\#\{x | x \notin \mathbf{x}_i\}} \right) / 21 ,$$

$$\text{sensitivity} = \left(\sum_{i=1}^{21} \frac{\sum_{j=1}^{n*_i} I(\hat{y}_{x*_{ij}} \cap \mathbf{y}_i \neq \emptyset)}{n_i} \right) / 21 .$$

3.2.4. Identifying features important for COVID-19

Finally, we evaluate the quality of embeddings based on their ability in identifying important features for selecting important features for studying a disease of interest. Specifically, we examine whether MIGKI can effectively identify relevant signs/symptoms, medications, laboratory tests to study COVID-19 as a novel disease. These features can be used as downstream predictive modeling such as predicting mortality risk for patients hospitalized with COVID-19 or the risk of experiencing post acute sequelae COVID-19 infection. To identify COVID-19 related features, we created the SPPMI matrix for MGB and VA based on co-occurrence matrices from the EHR data of 14,885 and 100K COVID-19 positive patients up to December, 2020, respectively. We trained MIKGI embeddings for COVID-19 by integrating the COVID-specific SPPMI matrices together with the SAPBERT embeddings. We set the U07.1 ICD code as the target code and identify all EHR codes with cosine similarity higher than a threshold value c_0, where c_0 is chosen as the 99% percentile of the cosine similarity of randomly selected code pairs.

4. Results

Based on the strategy of tuning parameters discussed in Section 2.2.1, we chose the dimension of the SPPMI-SVD embedding for building SynC embedding as 1500.

Table 3
Weighted average AUCs of between-vector cosine similarity in detecting known similar pairs and related pairs with embeddings trained via MIKGI, SAPBERT, BioBERT, PubmedBert, CODER, Orthogonal transformation (OT) of SPPMI-SVD, and Pre-training (PT) of SPPMI-SVD.

Method	MIKGI	SAPBERT	BioBERT	PubmedBERT	CODER	OT	PT
Similarity	0.918	0.832	0.691	0.643	**0.940**	0.877	0.863
Relatedness	**0.809**	0.698	0.581	0.556	0.733	0.805	0.792

4.1. Performance in detecting known similar or related pairs

For detecting similar pairs, the dimension r of MIKGI was chosen as 200 and the coefficient λ is set as 0.55; for detecting related pairs, $r = 200$ and $\lambda = 0.95$. Table 3 presents the average AUCs of the between vector cosine similarity in detecting known similar pairs and related pairs. The AUCs for detecting different types of relation pairs are shown in Tables 6 and 7 of Appendix S.3. MIKGI is the only algorithm that attained the highest or near the highest performance for both similarity and relatedness detection. It is not surprising that the CODER embeddings attained a high AUC of 0.940 in detecting similar pairs since they are trained leveraging similarity and relatedness information stored in the UMLS. SAPBERT, which only used synonymous information in the UMLS, attained a lower AUC of 0.832. Nevertheless, MIKGI has a comparable result to CODER with an AUC of 0.918 in detecting similar pairs. For detecting the relatedness relationship, SAPBERT, BioBERT, PubmedBert and CODER perform substantially worse with an AUCs lower than 0.74 while OT, leveraging EHR co-occurrence patterns within each of the two institutions, attained an AUC of 0.805. On the other hand, by combining both textual descriptions and co-occurrence information, MIGKI attained the highest AUC of 0.809. The PT embeddings based on EHR co-occurrence and the SAPBERT, PubmedBERT and BioBERT embeddings did not perform well in general with relatively low AUC for both similarity and relatedness. The CODER embeddings can detect the similar pairs but are less useful in detecting the related pairs. MIKGI is the only algorithm that can accurately detect both similarity and relatedness relationships. It is not supervising as MIKGI utilizes two kind of information and integrates them properly.

4.2. Cross-institutional code mapping

For code mapping tasks, we tuned the MIKGI to optimize lab mapping based on 50 pairs of labels and detection of known similar drugs which yielded $r = 200$ and $\lambda = 0.75$ for lab mapping and $r = 200$ and $\lambda = 0.25$ for medication mapping. After supervised training with 500 pairs of labels, we obtain $r = 200$ and $\lambda = 0.45$ for lab mapping. Table 4 summarizes the acc@k of mapping VA local laboratory codes to LOINC or LP codes and VA local medication codes to RxNorm, for $k = 1, 5, 10$, and 20.

For both lab and medication code mapping, OT, PT, BioBERT and PubMedBERT performed poorly. Compared to SAPBERT and CODER, MIKGI attained a higher accuracy with acc@1 of 0.59 vs. 0.54 and 0.46 for lab code mapping and 0.91 vs. 0.76 and 0.78 for medication to RxNorm mapping. Using the supervised training with 500 labels, the lab code mapping accuracy improved substantially for MIKGI, SAPBERT and CODER, although the improvement of MIKGI is the most substantial with acc@1 of supervised MIKGI vs SAPBERT and CODER was 0.77 vs. 0.66 and 0.69.

4.3. Case study of selecting VA local lab codes for a COVID EHR study

Table 5 shows the accuracy of mapping VA local lab codes to desired LOINC codes for 21 laboratory tests used in a COVID study. Some examples of the Match@3 of MIKGI are given in Table 8 of Supplementary Materials S.3. MIKGI achieved the highest sensitivity among all methods while maintaining a high specificity. For Match@1, MIKGI and SAPBERT attained comparable accuracy with a sensitivity of 80% and 71% and a specificity of 99%. All other approaches led to much lower sensitivity, ranging from 2% to 55%. These results suggest that MIKGI can be effectively used as a screening tool to identify potential local codes for a set of target lab tests.

4.4. Performance in identifying COVID-19 related codes

Based on MIKGI trained embeddings, a total of 167 original EHR codes and 25 LP codes were identified as potentially relevant for COVID-19 according to their cosine similarity with the U07.1 ICD code being higher than the threshold c_0. On the other hand, SAPBERT and CODER failed to identify any code whose cosine with U07.1 is higher than the critical value. The top 140 selected codes by MIKGI are shown in Fig. 4 and we present top 140 selected codes by SAPBERT and by CODER in Figure 5 and Figure 6 of Supplementary Materials. MIKGI is able to identify key symptoms including shortness of breath and respiratory failure, highly important laboratory tests including C-reactive Protein, D-dimer, ferritin, cardiac troponin, and various oxygen level related measures; medications including dexamethasone and remdesivir, as well as procedures including respiratory intubation and mechanical ventilation. These selected symptoms, laboratory tests, medications, and procedures are consistent with recent literature on the diagnosis and management of COVID-19 [77,78], can serve as candidate features for deriving risk prediction models for COVID-19.

5. Discussion

In this paper, the proposed MIKGI algorithm generates high quality embeddings to simultaneously represent EHR codes from multiple institutions by integrating information from code descriptions and co-occurrence patterns. As demonstrated via our experiments, the proposed loss function is simple yet effective in combining such information efficiently. The algorithm only requires sharing of summary level EHR data, overcoming data privacy challenges. By providing a unified set of embeddings for all EHR codes from multiple institutions, MIKGI enables cross institutional code mapping for data harmonization.

MIKGI attains the most robust performance across multiple downstream tasks compared to the commonly used embedding methods including SAPBERT, BioBERT, PubmedBERT, CODER, OT, and PT. For detecting known relationship tasks, MIKGI is more effective than the existing embedding methods in detecting relatedness relationships since such relationships can be well captured in EHR data from healthcare systems, and the co-occurrence patterns reflect physician's decision processes in managing diseases.

For cross-institutional code mapping tasks, OT, PT, BioBERT, and PubmedBert performed poorly since these methods do not comprehensively leverage existing knowledge on biomedical entities from the UMLS. Both OT and PT only leverage code co-occurrence patterns in EHR data and do not incorporate code textual description information. In addition, since the code pairs requiring cross-institutional mapping do not co-occur within either institution, OT and PT are expected to have difficulty in inferring their relationships. BioBERT and PubMedBert also fail to accurately map the codes since these methods heavily rely on contextual information while the code descriptions are short phrases with insufficient contextual information. SAPBERT and CODER substantially outperform BioBERT and PubMedBERT in code mapping since both algorithms leverage a large amount of relational information between biomedical entities in the UMLS via contrastive learning. Nevertheless, MIKGI attains more robust performance than SAPBERT and CODER by further leveraging code co-occurrence patterns in EHR data, which essentially encode the meaning of codes in clinical practice based on their relationship with other codes.

In addition to its robust performance, the MIKGI algorithm has several major advantages in practice. First, MIKGI provides a statistically and computationally efficient framework for automated data

Table 4
Accuracy of mapping codes from other local lab codes in VA to LOINC codes in MGB and accuracy of mapping codes from MedProc codes in VA to RxNorm codes in MGB with embeddings trained via MIKGI, SAPBERT, BioBERT, PubmedBert, CODER, OT and PT. An corpus-based supervised learning algorithm (Max) is also included for the supervised learning analysis with 500 training labels.

Mapping type	Method	acc@1	acc@5	acc@10	acc@20
VA LAB Code → LOINC/LP	MIKGI	**0.591**	**0.758**	**0.818**	**0.871**
	SAPBERT	0.541	0.671	0.702	0.743
	BioBERT	0.113	0.151	0.174	0.196
	PubmedBERT	0.108	0.165	0.197	0.237
	CODER	0.459	0.645	0.698	0.755
	OT	0.017	0.060	0.101	0.178
	PT	0.008	0.025	0.044	0.078
VA Lab Code → LOINC/LP supervised learning with 500 labels	MIKGI	**0.777**	**0.879**	**0.904**	**0.929**
	SAPBERT	0.657	0.792	0.833	0.872
	BioBERT	0.067	0.126	0.164	0.210
	PubmedBERT	0.097	0.181	0.219	0.284
	CODER	0.693	0.811	0.853	0.887
	OT	0.123	0.211	0.284	0.369
	PT	0.098	0.177	0.232	0.306
	Max	0.654	0.677	0.711	0.736
VA Medication Code → RxNorm	MIKGI	**0.910**	**0.975**	**0.995**	**0.995**
	SAPBERT	0.759	0.945	0.960	0.975
	BioBERT	0.000	0.005	0.005	0.005
	PubmedBERT	0.000	0.000	0.000	0.000
	CODER	0.784	0.925	0.970	0.985
	OT	0.241	0.497	0.608	0.724
	PT	0.000	0.010	0.020	0.030

Table 5
The sensitivity and specificity of mapping 21 COVID related laboratory tests.

Method	Match@1		Match@5		Match@10		Match@20	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
MIKGI	**0.796**	0.988	**0.911**	0.976	**0.929**	0.961	**0.960**	0.933
SAPBERT	0.714	0.989	0.812	0.977	0.844	0.962	0.906	0.936
BioBERT	0.140	0.993	0.235	0.979	0.251	0.964	0.282	0.935
PubmedBERT	0.169	0.994	0.236	0.979	0.308	0.960	0.396	0.929
CODER	0.549	0.991	0.740	0.975	0.798	0.960	0.873	0.936
OT	0.119	0.997	0.167	0.988	0.217	0.976	0.270	0.955
PT	0.016	**0.998**	0.016	**0.992**	0.125	**0.985**	0.146	**0.974**

Fig. 4. Top 140 Codes related with U07.1 ICD code detected by MIKGI based on cosine similarity. We omit LP codes and integer level parent phecodes if child codes are present for conciseness. Since many oxygen related local codes were selected, we manually merged and annotated in the figure for ease of presentation.

harmonization across multiple institutions compared to the state-of-art NLP models and KG embedding models which require extensive training. As illustrated by the COVID-19 study, inconsistent coding of the key laboratories measures impose a great challenge for federated predictive modeling analyses. Therefore, the MIKGI framework is critical to facilitating federated learning. Second, besides code mapping, the MIKGI-based code embeddings can be used to directly project patient features to an embedding space. Because MIKGI integrates data from multiple institutions, such patient level embeddings are harmonized features with low heterogeneity across institutions, which can facilitate multi-institutional integrative analysis and improve cross-institutional transportability. By incorporating EHR co-occurrence information, MIKGI also allows the same code (e.g., U07.1 for COVID-19) to be represented differently at different healthcare centers to capture between-healthcare heterogeneity. Furthermore, since MIKGI only relies on the simple co-occurrence matrix of code pairs from the EHR along with code descriptions, it can be updated over time to incorporate new diseases or treatments such as COVID-19. Once these new clinical concepts appear in the EHR, we may generate MIKGI embeddings for these concepts for downstream analysis of a relevant disease, either serving as feature selection tools or directly representing selected features.

Although the current implementation of MIKGI only includes two institutions, it can be easily adapted to include three or more EHR systems. This can be achieved by formulating the problem as a multi-task matrix completion, which can be solved via a gradient descent algorithm. An potential alternative approach is to first employ the **B**lockwise **O**verlapping **N**oisy **M**atrix **I**ntegration (BONMI) algorithm [79] to generate a EHR SPPMI based embeddings to replace the embeddings **V** proposed in Section 2.1.2. The BONMI algorithm allows any number of institutions and hence can be combined with SAPBERT through the proposed consensus learning algorithm in (1).

The main goal of MIKGI is to provide high quality embeddings for multi-institutional EHR codes to enable downstream tasks such as code mapping or predictive modeling. We use the code mapping task as one approach to evaluate the quality of the MIKGI embedding and the MIKGI embeddings do not rely on cross-institutional code mapping labels. The unsupervised mapping of VA medication to RxNorm attained a high acc@1 of 91% while the unsupervised VA lab code mapping only attained acc@1 of 59%. This is in part due to (i) the challenge of lab code descriptions having varying degrees of ambiguity with the use of acronyms; (ii) not fully utilizing the EHR information on the laboratory tests such as their findings (e.g., high vs normal); (iii) certain laboratory tests are always ordered together as part of a panel, which results identical EHR embeddings for their associated EHR codes. It is possible to further improve the code mapping accuracy via supervised methods as demonstrated in the validation study. The acc@10 of MIKGI based supervised algorithm can reach 90%, suggesting that one may use the MIKGI algorithm as a smart search tool to semi-automatically identify the correct mapping. This can substantially reduce human effort needed to map local codes to common ontology. Other supervised or semi-supervised machine learning methods can be used such as transductive vector support machine [80], set covering machine [81], and freetext matching algorithm [82]. Such supervised approaches can also be used to further enrich the knowledge network by predicting specific associations between a pair of EHR concepts such as "may treat" or "may cause".

CRediT authorship contribution statement

Doudou Zhou: Methodology, Software, Writing – original draft. **Ziming Gan:** Methodology, Software, Writing – original draft. **Xu Shi:** Writing – review & editing. **Alina Patwari:** Validation, Data curation. **Everett Rush:** Data curation, Resources. **Clara-Lea Bonzel:** Visualization. **Vidul A. Panickan:** Data curation. **Chuan Hong:** Data curation. **Yuk-Lam Ho:** Data curation. **Tianrun Cai:** Data curation. **Lauren Costa:** Writing – review & editing, Project administration. **Xiaoou Li:** Writing – review & editing. **Victor M. Castro:** Writing – review & editing. **Shawn N. Murphy:** Writing – review & editing. **Gabriel Brat:** Writing – review & editing. **Griffin Weber:** Writing – review & editing. **Paul Avillach:** Writing – review & editing. **J. Michael Gaziano:** Writing – review & editing. **Kelly Cho:** Writing – review & editing. **Katherine P. Liao:** Writing – review & editing. **Junwei Lu:** Methodology, Conceptualization, Writing – review & editing, Supervision. **Tianxi Cai:** Conceptualization, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project was supported by the NIH 1OT2OD032581 grant (Biomedical Informatics and AI/ML Pipelines to Advance Health Equity and Research Diversity - Infrastructure core) and the Million Veteran Program, Department of Veteran Affairs, Office of Research and Development, Veterans Health Administration, supported by award (#MVP000). This research used resources of the Knowledge Discovery Infrastructure at Oak Ridge National Laboratory, which is supported by the Office of Science of the US Department of Energy under Contract No. DE-AC05-00OR22725. This publication does not represent the views of the Department of Veterans Affairs or the U.S. government.

Appendix A. This document contains the pre-training and projected gradient descent algorithms, and additional numeric results

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jbi.2022.104147.

References

[1] C.Z. Lipton, C.D. Kale, C. Elkan, R. Wetzell, Learning to diagnose with LSTM recurrent neural networks, in: International Conference on Learning Representations, 2016.
[2] E. Choi, M.T. Bahadori, A. Schuetz, W.F. Stewart, J. Sun, Doctor AI: Predicting clinical events via recurrent neural networks, in: Machine Learning for Healthcare Conference, 2016, pp. 301–318.
[3] A. Rajkomar, E. Oren, K. Chen, A.M. Dai, N. Hajaj, M. Hardt, P.J. Liu, X. Liu, J. Marcus, M. Sun, et al., Scalable and accurate deep learning with electronic health records, NPJ Digit. Med. 1 (1) (2018) 18.
[4] P. Federico, J. Unger, A. Amor-Amorós, L. Sacchi, D. Klimov, S. Miksch, Gnaeus: Utilizing clinical guidelines for knowledge-assisted visualisation of EHR cohorts., in: EuroVA@ EuroVis, 2015, pp. 79–83.
[5] K. Chunchu, L. Mauksch, C. Charles, V. Ross, J. Pauwels, A patient centered care plan in the EHR: improving collaboration and engagement, Fam. Syst. Health 30 (3) (2012) 199.
[6] A.L. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, I.S. Kohane, Clinical concept embeddings learned from massive sources of multimodal medical data, in: Biocomputing 2020, WORLD SCIENTIFIC, 2019, http://dx.doi.org/10.1142/9789811215636_0027.
[7] Y. Choi, C.Y.-I. Chiu, D. Sontag, Learning low-dimensional representations of medical concepts, AMIA Summits Transl. Sci. Proc. 2016 (2016) 41.
[8] R. Miotto, L. Li, B.A. Kidd, J.T. Dudley, Deep patient: an unsupervised representation to predict the future of patients from the electronic health records, Sci. Rep. 6 (1) (2016) 1–10.
[9] R. Belenkaya, A. Watson, S. Bethusamy, M. Patel, T. Sandler, J. Schwartz, J. Park, M. Dobbins, M. Maloy, M. Lam, et al., Abstract PO-061: Data harmonization for COVID-19 and cancer research registries, 2020.
[10] J.G. Klann, M.A. Joss, K. Embree, S.N. Murphy, Data model harmonization for the all of us research program: Transforming i2b2 data into the OMOP common data model, PLoS One 14 (2) (2019) e0212463.
[11] S. Beer-Borst, S. Hercberg, A. Morabia, M. Bernstein, P. Galan, R. Galasso, S. Giampaoli, E. McCrum, S. Panico, P. Preziosi, et al., Dietary patterns in six European populations: results from EURALIM, a collaborative European data harmonization and information campaign, Eur. J. Clin. Nutr. 54 (3) (2000) 253–262.

[12] J. Kalter, M.G. Sweegers, I.M. Verdonck-de Leeuw, J. Brug, L.M. Buffart, Development and use of a flexible data harmonization platform to facilitate the harmonization of individual patient data for meta-analyses, BMC Res. Notes 12 (1) (2019) 1–6.

[13] D. Doiron, P. Raina, F. L'Heureux, I. Fortier, Facilitating collaborative research: Implementing a platform supporting data harmonization and pooling, Norsk Epidemiol. 21 (2) (2012).

[14] R.V. Burkhauser, D.R. Lillard, The contribution and potential of data harmonization for cross-national comparative research, J. Comp. Policy Anal. 7 (4) (2005) 313–330.

[15] D. Liu, X. Wang, F. Pan, P. Yang, Y. Xu, X. Tang, J. Hu, K. Rao, Harmonization of health data at national level: a pilot study in China, Int. J. Med. Inform. 79 (6) (2010) 450–458.

[16] OMOP, Omop, 2021, https://ohdsi.org/omop/, Accessed June, 2021.

[17] R.L. Fleurence, L.H. Curtis, R.M. Califf, R. Platt, J.V. Selby, J.S. Brown, Launching PCORnet, a national patient-centered clinical research network, J. Amer. Med. Inform. Assoc. 21 (4) (2014) 578–582.

[18] J. Weeks, R. Pardee, Learning to share health care data: a brief timeline of influential common data models and distributed health data networks in US health care research, EGEMs 7 (1) (2019).

[19] G. Hripcsak, J.D. Duke, N.H. Shah, C.G. Reich, V. Huser, M.J. Schuemie, M.A. Suchard, R.W. Park, I.C.K. Wong, P.R. Rijnbeek, et al., Observational health data sciences and informatics (OHDSI): opportunities for observational researchers, Stud. Health Technol. Inform. 216 (2015) 574.

[20] P. Hernandez, T. Podchiyska, S. Weber, T. Ferris, H. Lowe, Automated mapping of pharmacy orders from two electronic health record systems to RxNorm within the STRIDE clinical data warehouse, in: AMIA Annual Symposium Proceedings, Vol. 2009, American Medical Informatics Association, 2009, p. 244.

[21] S. Abhyankar, D. Demner-Fushman, C.J. McDonald, Standardizing clinical laboratory data for secondary use, J. Biomed. Inform. 45 (4) (2012) 642–650.

[22] W.H. Organization, et al., International classification of diseases—Ninth revision (ICD-9), Wkly. Epidemiol. Rec.=Relev. Épidém'iol. Hebd. 63 (45) (1988) 343–344.

[23] G.R. Brämer, International Statistical Classification of Diseases and Related Health Problems. Tenth Revision 41, World Health Statistics Quarterly. Rapport Trimestriel de Statistiques Sanitaires Mondiales, 1988, pp. 32–36.

[24] S. Liu, W. Ma, R. Moore, V. Ganesan, S. Nelson, Rxnorm: prescription for electronic drug information exchange, IT Prof. 7 (5) (2005) 17–23.

[25] J.A. Hirsch, T.M. Leslie-Mazwi, G.N. Nicola, R.M. Barr, J.A. Bello, W.D. Donovan, R. Tu, M.D. Alson, L. Manchikanti, Current procedural terminology; a primer, J. Neurointerventional Surg. 7 (4) (2015) 309–312.

[26] C.J. McDonald, S.M. Huff, J.G. Suico, G. Hill, D. Leavelle, R. Aller, A. Forrey, K. Mercer, G. DeMoor, J. Hook, et al., Loinc, a universal standard for identifying laboratory observations: a 5-year update, Clin. Chem. 49 (4) (2003) 624–633.

[27] M.-C. Lin, D.J. Vreeman, C.J. McDonald, S.M. Huff, Correctness of voluntary LOINC mapping for laboratory tests in three large institutions, in: AMIA Annual Symposium Proceedings, Vol. 2010, American Medical Informatics Association, 2010, pp. 447–451.

[28] X. Shi, X. Li, T. Cai, Spherical regression under mismatch corruption with application to automated knowledge translation, J. Amer. Statist. Assoc. (2020) 1–12.

[29] N. Kume, K. Suzuki, S. Kobayashi, H. Yoshihara, K. Araki, Original laboratory test code mapping system using test result data on electronic health record, in: MEDINFO 2019: Health and Wellbeing E-Networks for All, IOS Press, 2019, pp. 1518–1519.

[30] G. Tournavitis, Z. Wang, B. Franke, M.F. O'Boyle, Towards a holistic approach to auto-parallelization: integrating profile-driven parallelism detection and machine-learning based mapping, ACM Sigplan Not. 44 (6) (2009) 177–187.

[31] C. Baloukas, L. Papadopoulos, D. Soudris, S. Stuijk, O. Jovanovic, F. Schmoll, D. Cordes, R. Pyka, A. Mallik, S. Mamagkakis, et al., Mapping embedded applications on mpsocs: the MNEMEE approach, in: 2010 IEEE Computer Society Annual Symposium on VLSI, IEEE, 2010, pp. 512–517.

[32] D.M. Baorto, J.J. Cimino, C.A. Parvin, M.G. Kahn, Combining laboratory data sets from multiple institutions using the logical observation identifier names and codes (LOINC), Int. J. Med. Inform. 51 (1) (1998) 29–37.

[33] L.M. Lau, K. Johnson, K. Monson, S.H. Lam, S.M. Huff, A method for the automated mapping of laboratory results to LOINC, in: Proceedings of the AMIA Symposium, American Medical Informatics Association, 2000, p. 472.

[34] H. Kim, R. El-Kareh, A. Goel, F. Vineet, W.W. Chapman, An approach to improve LOINC mapping through augmentation of local test names, J. Biomed. Inform. 45 (4) (2012) 651–657.

[35] G. Kopanitsa, Application of a regenstrief RELMA V. 6.6 to map Russian laboratory terms to LOINC, Methods Inf. Med. 55 (02) (2016) 177–181.

[36] C. Zunner, T. Bürkle, H.-U. Prokosch, T. Ganslandt, Mapping local laboratory interface terms to LOINC at a German university hospital using RELMA v. 5: a semi-automated approach, J. Amer. Med. Inform. Assoc. 20 (2) (2013) 293–297.

[37] L. Peters, J.E. Kapusnik-Uner, O. Bodenreider, Methods for managing variation in clinical drug names, in: AMIA Annual Symposium Proceedings, Vol. 2010, American Medical Informatics Association, 2010, p. 637.

[38] L. Zhou, J.M. Plasek, L.M. Mahoney, F.Y. Chang, D. DiMaggio, R.A. Rocha, Mapping partners master drug dictionary to RxNorm using an NLP-based approach, J. Biomed. Inform. 45 (4) (2012) 626–633.

[39] M. Fidahussein, D.J. Vreeman, A corpus-based approach for automated LOINC mapping, J. Amer. Med. Inform. Assoc. 21 (1) (2014) 64–72.

[40] A.N. Khan, S.P. Griffith, C. Moore, D. Russell, A.C. Rosario Jr., J. Bertolli, Standardizing laboratory data by mapping to LOINC, J. Amer. Med. Inform. Assoc. 13 (3) (2006) 353–355.

[41] J.Y. Sun, Y. Sun, A system for automated lexical mapping, J. Amer. Med. Inform. Assoc. 13 (3) (2006) 334–343.

[42] K.J. Holyoak, Parallel distributed processing: explorations in the microstructure of cognition, Science 236 (1987) 992–997.

[43] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 533–536.

[44] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: Proceedings of Workshop At ICLR, 2013, pp. 2013.

[45] T. Mikolov, Q.V. Le, I. Sutskever, Exploiting similarities among languages for machine translation, 2013, arXiv preprint arXiv:1309.4168.

[46] I. Balažević, C. Allen, T.M. Hospedales, Tucker: Tensor factorization for knowledge graph completion, 2019, arXiv preprint arXiv:1901.09590.

[47] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 28, 2014.

[48] L. Yao, C. Mao, Y. Luo, KG-BERT: BERT For knowledge graph completion, 2019, arXiv preprint arXiv:1909.03193.

[49] Z. Yuan, Z. Zhao, H. Sun, J. Li, F. Wang, S. Yu, CODER: KNowledge-infused cross-lingual medical term embedding for term normalization, J. Biomed. Inform. (2022) 103983.

[50] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Adv. Neural Inf. Process. Syst. 26 (2013) 3111–3119.

[51] C. Lin, Y.-S. Lou, D.-J. Tsai, C.-C. Lee, C.-J. Hsu, D.-C. Wu, M.-C. Wang, W.-H. Fang, Projection word embedding model with hybrid sampling training for classifying ICD-10-CM codes: Longitudinal observational study, JMIR Med. Inform. 7 (3) (2019) e14499.

[52] W. Boag, H. Kané, Awe-cm vectors: Augmenting word embeddings with a clinical metathesaurus, 2017, arXiv preprint arXiv:1712.01460.

[53] O. Levy, Y. Goldberg, Neural word embedding as implicit matrix factorization, in: Advances in Neural Information Processing Systems, Vol. 27, 2014.

[54] J. Pennington, R. Socher, C.D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.

[55] C. Hong, E. Rush, M. Liu, D. Zhou, J. Sun, A. Sonabend, V.M. Castro, P. Schubert, V.A. Panickan, T. Cai, et al., Clinical knowledge extraction via sparse embedding regression (KESER) with multi-center large scale electronic health record data, NPJ Digit. Med. 4 (1) (2021) 1–11.

[56] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: Neural Information Processing Systems (NIPS), 2013, pp. 1–9.

[57] S.M. Kazemi, D. Poole, Simple embedding for link prediction in knowledge graphs, Adv. Neural Inf. Process. Syst. 31 (2018).

[58] Y. Peng, J. Zhang, Lineare: Simple but powerful knowledge graph embedding for link prediction, in: 2020 IEEE International Conference on Data Mining (ICDM), IEEE, 2020, pp. 422–431.

[59] M. Wang, L. Qiu, X. Wang, A survey on knowledge graph embeddings for link prediction, Symmetry 13 (3) (2021) 485.

[60] S.L. Smith, D.H. Turban, S. Hamblin, N.Y. Hammerla, Offline bilingual word vectors, orthogonal transformations and the inverted softmax, ICLR (2017).

[61] Y. Kementchedjhieva, S. Ruder, R. Cotterell, A. Søgaard, Generalizing procrustes analysis for better bilingual dictionary induction, in: Proceedings of the 22nd Conference on Computational Natural Language Learning, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 211–220.

[62] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, H. Jégou, Word translation without parallel data, in: Proceedings of ICLR, 2018.

[63] D. Zhou, T. Cai, J. Lu, Multi-source learning via completion of block-wise overlapping noisy matrices, 2021, arXiv:2105.10360.

[64] C. Xing, D. Wang, C. Liu, Y. Lin, Normalized word embedding and orthogonal transform for bilingual word translation, in: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Denver, Colorado, 2015, pp. 1006–1011.

[65] A.L. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, I.S. Kohane, Clinical concept embeddings learned from massive sources of multimodal medical data, in: Pacific Symposium on Biocomputing 2020, World Scientific, 2019, pp. 295–306.

[66] O. Bodenreider, The unified medical language system (UMLS): integrating biomedical terminology, Nucleic Acids Res. 32 (suppl_1) (2004) D267–D270.

[67] F. Liu, E. Shareghi, Z. Meng, M. Basaldella, N. Collier, Self-alignment pretraining for biomedical entity representations, 2020, arXiv preprint arXiv:2010.11784.

[68] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.
[69] I. Jolliffe, Principal component analysis, Encyclopedia Statist. Behav. Sci. (2005).
[70] S.N. Murphy, G. Weber, M. Mendis, V. Gainer, H.C. Chueh, S. Churchill, I. Kohane, Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2), J. Amer. Med. Inform. Assoc. 17 (2) (2010) 124–130.
[71] C. McDonald, S. Huff, J. Suico, K. Mercer, Logical Observation Identifiers Names and Codes (LOINC®) Users' Guide, Regenstrief Institute, Indianapolis, 2004.
[72] G.A. Brat, G.M. Weber, N. Gehlenborg, P. Avillach, N.P. Palmer, L. Chiovato, J. Cimino, L.R. Waitman, G.S. Omenn, A. Malovini, et al., International electronic health record-derived COVID-19 clinical course profiles: the 4CE consortium, Npj Digit. Med. 3 (1) (2020) 1–9.
[73] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (4) (2020) 1234–1240.
[74] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, 2020, arXiv:arXiv:2007.15779.
[75] Drugs, Drugs.com, 2021, https://www.drugs.com/, Accessed June, 2021.
[76] V. VHA, National Drug File Reference Terminology (NDF-RT) Documentation, US Department of Veterans Affairs, 2012.
[77] M.S. Mughal, I.P. Kaur, A.R. Jaffery, D.L. Dalmacion, C. Wang, S. Koyoda, V.E. Kramer, C.D. Patton, S. Weiner, M.H. Eng, et al., COVID-19 Patients in a tertiary US hospital: Assessment of clinical course and predictors of the disease severity, Respir. Med. 172 (2020) 106130.
[78] P. Zhai, Y. Ding, X. Wu, J. Long, Y. Zhong, Y. Li, The epidemiology, diagnosis and treatment of COVID-19, Int. J. Antimicrob. Ag. 55 (5) (2020) 105955.
[79] D. Zhou, T. Cai, J. Lu, Multi-source learning via completion of block-wise overlapping noisy matrices, 2021, arXiv preprint arXiv:2105.10360.
[80] V.N. Vapnik, Statistical Learning Theory, Wiley-Interscience, 1998.
[81] R. Rosales, P. Krishnamurthy, R.B. Rao, Semi-supervised active learning for modeling medical concepts from free text, in: Sixth International Conference on Machine Learning and Applications (ICMLA 2007), IEEE, 2007, pp. 530–536.
[82] Z. Wang, A.D. Shah, A.R. Tate, S. Denaxas, J. Shawe-Taylor, H. Hemingway, Extracting diagnoses and investigation results from unstructured text in electronic health records by semi-supervised machine learning, PLoS One 7 (1) (2012) e30412.