A Dataset sizes for training and validation

		CD	RA	CHF
MGB	Train	516	984	2213
	Valid (#pos)	136 (73)	153 (56)	113 (28)
VA	Train	NA	27424	120067
	Valid (#pos)		98 (64)	190 (144)
BCH	Train	1009	NA	4516
	Valid (#pos)	25 (21)		20 (5)

(a) Phenotypes shared across sites

		CAD	UC	T1DM	T2DM	Depression
MGB	Train	3278	473	1651	2227	9236
	Valid (#pos)	151 (69)	126 (61)	113 (21)	174 (94)	235 (131)
		MS	AD			Asthma
UPMC	Train	16972	80526	**BCH**	Train	21924
	Valid (#pos)	449 (346)	210 (98)		Valid (#pos)	20 (11)

(b) Site-unique phenotypes

Table 2: Summary of the sample sizes of training and validation label data for each phenotype across sites.

B Technical details of the KOMAP algorithm

B.1 Training steps

Recall that c is the p-dimensional EHR feature vector of the i-th subject as defined in Section 2.2. The KOMAP online training algorithm, as outlined in Figure 2, starts with the derived covariance matrix $\mathbb{C} = N^{-1}\sum_{i=1}^{N}(\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^\intercal$, where $\bar{\mathbf{X}}$ is the mean vector of $\mathbf{X}_1, ..., \mathbf{X}_N$. Firstly, we perform Cholesky decomposition on $\mathbb{C}$ to obtain the embedding matrix $\mathbb{U}$ such that $\mathbb{C} = \mathbb{U}^\intercal\mathbb{U}$, where $\mathbb{U} = [\mathbb{U}_1, ..., \mathbb{U}_K, ..., \mathbb{U}_p] \in \mathbb{R}^{p\times p}$ stores the vector of the corresponding original features in $\mathbf{X}$. Secondly, we regress out the effect of healthcare utilization $\mathcal{H}$ from each main surrogate k by linear regression: $\tilde{\alpha}_k = \text{argmin}_{\alpha_k}\|\mathbb{U}_k - \alpha_k\mathbb{U}_p\|_2^2$, and denote $\tilde{\mathbb{U}} = [\tilde{\mathbb{U}}_1, ..., \tilde{\mathbb{U}}_K, \mathbb{U}_{K+1}, ..., \mathbb{U}_p]$ as the embedding matrix with the main surrogate columns replaced by their utilization-adjusted version (residuals):

$$\tilde{\mathbb{U}}_k = \mathbb{U}_k - \tilde{\alpha}_k\mathbb{U}_p. \tag{A.1}$$

As the third step, we adopt a similar denoising procedure as used in (12) that performs elastic net (24) on $\tilde{\mathbb{U}}_k$ against all features in $\tilde{\mathbb{U}}$ to derive the denoising coefficient vector:

$$\tilde{\boldsymbol{\beta}}_k = \underset{\boldsymbol{\beta}}{\text{argmin}}\,\|\tilde{\mathbb{U}}_k - \tilde{\mathbb{U}}\boldsymbol{\beta}\|_2^2 + \lambda_k\{\zeta_k\|\boldsymbol{\beta}\|_1 + (1-\zeta_k)\|\boldsymbol{\beta}\|_2^2\}, \quad \text{for} \quad k = 1, 2, ..., K. \tag{A.2}$$

Tuning strategy of the penalty parameters ζ_k and λ_k is described in Supplement B.2. Then we are able to derive the denoised embedding score $\mathbb{W}_k = \tilde{\mathbb{U}}\tilde{\boldsymbol{\beta}}_k$ for the main surrogates and denote them by $\mathbb{W} = (\mathbb{W}_1, \mathbb{W}_2, \ldots, \mathbb{W}_K) = \tilde{\mathbb{U}}\tilde{\mathbf{B}} = \mathbb{U}\tilde{\mathbf{A}}\tilde{\mathbf{B}}$, where $\tilde{\mathbf{B}} = (\tilde{\boldsymbol{\beta}}_1, ..., \tilde{\boldsymbol{\beta}}_K) \in \mathbb{R}^{p\times K}$,

$$\tilde{\mathbf{A}} = \begin{pmatrix} \mathbb{I}_{(p-1)\times(p-1)} & \mathbf{0}_{(p-1)\times 1} \\ -\tilde{\boldsymbol{\alpha}}^\intercal & 1 \end{pmatrix},$$

and $\tilde{\boldsymbol{\alpha}} = \{\tilde{\alpha}_1, ..., \tilde{\alpha}_K, \mathbf{0}_{1\times(p-K-1)}\}^\intercal$. Also, with $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$, we may reconstruct the normalizing and denoising coefficients for $\mathbf{X}$ as $\tilde{\mathbf{A}}\tilde{\mathbf{B}}$, which gives a K-dimensional vector of the denoising scores corresponding to the K main surrogates.

If we only have one surrogate, then the final phenotyping score of the original embedding matrix $\mathbb{U}$ is $\mathbb{W} = \mathbb{W}_1$ and the transferable phenotyping coefficient vector is $\tilde{\mathbf{A}}\tilde{\boldsymbol{\beta}}_1$. If multiple surrogates are available (i.e., $K > 1$), we will take one more step to combine the denoised scores in $\mathbb{W}$ by finding the loading corresponding to the first principle component:

$$\tilde{\boldsymbol{\gamma}} = \text{argmax}_{\boldsymbol{\gamma}} \boldsymbol{\gamma}^\intercal \mathbb{W}^\intercal \mathbb{W} \boldsymbol{\gamma}, \quad \text{s.t.} \quad \|\boldsymbol{\gamma}\|_2 = 1. \tag{A.3}$$

This procedure could provide the most representative direction of the scores from multiple main surrogates. The final phenotyping score of the original embedding matrix $\mathbb{U}$ becomes $\mathbb{W}\tilde{\boldsymbol{\gamma}} = \mathbb{U}\tilde{\mathbf{A}}\tilde{\mathbf{B}}\tilde{\boldsymbol{\gamma}}$ and the transferable coefficient is $\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}} = \tilde{\mathbf{A}}\tilde{\mathbf{B}}\tilde{\boldsymbol{\gamma}} \in \mathbb{R}^{p\times 1}$.

Finally, we demonstrate the equivalence between the phenotyping coefficient vector $\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}} = \tilde{\mathbf{A}}\tilde{\mathbf{B}}\tilde{\boldsymbol{\gamma}}$ derived with $\mathbb{U}$ and that obtained with the centralized patient-level data matrix $\mathbb{X} = (\mathbf{X}_1 - \bar{\mathbf{X}}, ..., \mathbf{X}_N - \bar{\mathbf{X}})^\intercal$ using the same procedures, leveraging the fact that the embedding matrix and the original individual data matrix share the same covariance structure. We inspect (A.1), (A.2), and (A.3) to justify the equivalence results for $\tilde{\alpha}_k$, $\tilde{\boldsymbol{\beta}}_k$, and $\tilde{\boldsymbol{\gamma}}$ respectively. First, we note that

$$\text{argmin}_{\alpha_k} \|\mathbb{U}_k - \alpha_k \mathbb{U}_p\|_2^2 = \text{argmin}_{\alpha_k} (\mathbb{U}_p^\intercal \mathbb{U}_p \alpha_k^2 - 2\mathbb{U}_p^\intercal \mathbb{U}_k \alpha_k) = \text{argmin}_{\alpha_k} (\mathbb{X}_p^\intercal \mathbb{X}_p \alpha_k^2 - 2\mathbb{X}_p^\intercal \mathbb{X}_k \alpha_k),$$

where $\mathbb{X}_j$ represents the j-th column of $\mathbb{X}$. This implies that $\tilde{\alpha}_k$'s obtained in this way are equal to those we would obtain using $\mathbb{X}$. Consequently, we have $\tilde{\mathbb{U}}^\intercal\tilde{\mathbb{U}} = \tilde{\mathbb{X}}^\intercal\tilde{\mathbb{X}}$ where $\tilde{\mathbb{X}} = (\tilde{\mathbb{X}}_1, ..., \tilde{\mathbb{X}}_K, \mathbb{X}_{K+1}, ..., \mathbb{X}_p)$ and $\tilde{\mathbb{X}}_k = \mathbb{X}_k - \tilde{\alpha}_k \mathbb{X}_p$ are defined in the same way as $\tilde{\mathbb{U}}$. Then, similarly for the denoising step in (A.2), we have that the quadratic loss:

$$\|\tilde{\mathbb{U}}_k - \tilde{\mathbb{U}}\boldsymbol{\beta}\|_2^2 = \|\tilde{\mathbb{X}}_k - \tilde{\mathbb{X}}\boldsymbol{\beta}\|_2^2,$$

indicating that the denoising coefficient vector $\tilde{\boldsymbol{\beta}}_k$ derived with $\tilde{\mathbb{U}}$ is also identical to that obtained by performing the same elastic net regression on the individual data $\tilde{\mathbf{X}}$. As a result,

$$\mathbb{W}^\intercal\mathbb{W} = [\tilde{\mathbb{U}}\tilde{\mathbf{B}}]^\intercal\tilde{\mathbb{U}}\tilde{\mathbf{B}} = [\tilde{\mathbb{X}}\tilde{\mathbf{B}}]^\intercal\tilde{\mathbb{X}}\tilde{\mathbf{B}}.$$

So problem (A.3) that depends on $\mathbb{W}$ only through $\mathbb{W}^\intercal\mathbb{W}$ is equivalent to finding the first PC on the individual-level denoised scores $\mathbb{X}\tilde{\mathbf{B}}$. This indicates that $\tilde{\boldsymbol{\gamma}}$ derived with the embeddings $\mathbb{W}$ is also equal to the solution obtained with $\tilde{\mathbf{X}}\tilde{\mathbf{B}}$. Consequently, the derived coefficient $\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}} = \tilde{\mathbf{A}}\tilde{\mathbf{B}}\tilde{\boldsymbol{\gamma}}$ is exactly the same as the one derived by performing the same regression steps on the individual-level data $\mathbb{X}$.

B.2 Tuning strategy

To fine tune the penalty parameter λ, traditional cross-validation strategies cannot be applied here. Unlike the individual-level data $\mathbb{X}$ that are independent from each other, $\tilde{\mathbb{U}}$ derived from Cholesky decomposition has correlated rows and we cannot hold out part of it as a validation dataset. Another issue is when the predictors $\tilde{\mathbb{U}}$ in (A.2) include the response variable $\tilde{\mathbb{U}}_k$, even if we construct an independent validation data set to perform parameter tuning, the final λ would favor $\tilde{\beta}_{k,k}$ to be 1 while assign other coefficients close to be 0. Such a resulted coefficient vector is unwanted since it does not leverage information from the EHR features besides the main surrogate k. To avoid this problem, we randomly split the original N subjects into N_t tuning samples denoted as $\mathbb{X}^{\mathsf{tune}}$ and $N - N_t$ training samples $\mathbb{X}^{\mathsf{train}}$. Then we construct

$$\mathbb{X}^{\mathsf{tune}} = [\mathbb{X}_1^{\mathsf{tune}}, \mathbb{X}_1^{\mathsf{corrupt}}, ..., \mathbb{X}_K^{\mathsf{tune}}, \mathbb{X}_K^{\mathsf{corrupt}}, \mathbb{X}_{K+1}^{\mathsf{tune}}, ..., \mathbb{X}_p^{\mathsf{tune}}] \in \mathbb{R}^{N_t \times (p+K)},$$

where $\mathbb{X}_k^{\mathsf{corrupt}} = (X_{1k}^{\mathsf{corrupt}}, ..., X_{N_t k}^{\mathsf{corrupt}})$ and $X_{ik}^{\mathsf{corrupt}} = O_{ik} X_{ik}^{\mathsf{tune}} + (1 - O_{ik}) \bar{X}_k^{\mathsf{tune}}$ denotes the corrupted main surrogate, X_{ik}^{tune} is the subject i-th entry of $\mathbb{X}_k^{\mathsf{tune}}$, $\bar{X}_k^{\mathsf{tune}} = N_t^{-1} \sum_{i=1}^{N_t} X_{ik}^{\mathsf{tune}}$, and O_{ik}'s are independent and identically distributed Bernoulli random variables, with the dropout rate $P(O_{ik} = 0)$ fixed as 20%.

Similar to the training steps described in Supplement B.1, we derive $\mathbb{C}^{\mathsf{tune}}$ as the empirical covariance matrix of $\mathbb{X}^{\mathsf{tune}}$, and perform Cholesky decomposition on $\mathbb{C}^{\mathsf{tune}}$ to obtain the validation embedding matrix as

$$\mathbb{U}^{\mathsf{tune}} = [\mathbb{U}_1^{\mathsf{tune}}, \mathbb{U}_1^{\mathsf{corrupt}}, ..., \mathbb{U}_K^{\mathsf{tune}}, \mathbb{U}_K^{\mathsf{corrupt}}, \mathbb{U}_{K+1}^{\mathsf{tune}}, ..., \mathbb{U}_p^{\mathsf{tune}}] \in \mathbb{R}^{(p+K) \times (p+K)}.$$

Then, we directly borrow the regression coefficients for healthcare utilization $\mathcal{H}$ in (A.1) and adjust $\mathbb{U}_k^{\mathsf{tune}}, \mathbb{U}_k^{\mathsf{corrupt}}$ for $k = 1, ..., K$ accordingly:

$$\tilde{\mathbb{U}}_k^{\mathsf{tune}} = \mathbb{U}_k^{\mathsf{tune}} - \tilde{\alpha}_k \mathbb{U}_p^{\mathsf{tune}}; \quad \tilde{\mathbb{U}}_k^{\mathsf{corrupt}} = \mathbb{U}_k^{\mathsf{corrupt}} - \tilde{\alpha}_k \mathbb{U}_p^{\mathsf{tune}},$$

to obtain a partially corrupted and utilization-adjusted tuning embedding matrix:

$$\tilde{\mathbb{U}}^{\mathsf{tune}} = [\tilde{\mathbb{U}}_1^{\mathsf{tune}}, \tilde{\mathbb{U}}_1^{\mathsf{corrupt}}, ..., \tilde{\mathbb{U}}_K^{\mathsf{tune}}, \tilde{\mathbb{U}}_K^{\mathsf{corrupt}}, \mathbb{U}_{K+1}^{\mathsf{tune}}, ..., \mathbb{U}_p^{\mathsf{tune}}],$$

To find the optimal ζ_k and λ_k in (A.2) for each k, one could prespecify a candidate list $\mathcal{T}_k = \{(\lambda_k^\ell, \zeta_k^\ell) : \ell = 1, ..., L\}$, obtain the denoising vector $\tilde{\boldsymbol{\beta}}_k(\lambda_k^\ell, \zeta_k^\ell)$ using the embedding of the training data $\mathbb{X}^{\mathsf{train}}$, and evaluate the mean square error (MSE) with the tuning matrix:

$$\mathrm{MSE}\{\tilde{\boldsymbol{\beta}}_k(\lambda_k^\ell, \zeta_k^\ell)\} = N_t^{-1} \left\| \tilde{\mathbb{U}}_k^{\mathsf{tune}} - [\tilde{\mathbb{U}}_1^{\mathsf{tune}}, ..., \tilde{\mathbb{U}}_k^{\mathsf{corrupt}}, \tilde{\mathbb{U}}_{k+1}^{\mathsf{tune}}, ..., \tilde{\mathbb{U}}_K^{\mathsf{tune}}, ..., \mathbb{U}_p^{\mathsf{tune}}] \tilde{\boldsymbol{\beta}}_k(\lambda_k^\ell, \zeta_k^\ell) \right\|_2^2,$$

where only the k-th column in the covariate matrix is replaced with the corrupted feature embedding $\tilde{\mathbb{U}}_k^{\mathsf{corrupt}}$. Then we choose the candidate $(\lambda_k, \zeta_k) \in \mathcal{T}_k$ to minimize $\mathrm{MSE}\{\tilde{\boldsymbol{\beta}}_k(\lambda_k^\ell, \zeta_k^\ell)\}$. In our numerical studies, we find that the performance of KOMAP is not that sensitive to choice on ζ_k. Thus, we fix $\zeta_k = 0.15$ in the candidate set $\mathcal{T}_k$ for all k and only vary λ_k in a large enough range for tuning. This turns out to significantly speed up the tuning procedure while maintaining good prediction performances.

B.3 Validation steps

Suppose there is a small validation set of subjects: $\{(\mathbf{X}_i^v, Y_i^v) : i = 1, 2, \ldots, n\}$ with gold standard labels $Y_i^v \in \{0, 1\}$ obtained through chart reviewing by experts. Instead of referring to the individual phenotyping score $\hat{Y}^v = \hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \mathbf{X}^v$ directly used to evaluate model performance with Y^v, we further introduce an online model evaluation procedure that only relies on summary statistics derived from the validation set. This procedure aims at estimating the receiver operating characteristic (ROC) and the area under the ROC curve (AUC) of the KOMAP phenotyping score $\hat{Y}^v$.

Specifically, the user needs to extract the summary data by calculating the prevalence $\hat{p}_Y$ of Y^v, the (stratified) sample means $(\hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\mu}}_1)$ and covariance matrices $(\hat{\boldsymbol{\Sigma}}_0, \hat{\boldsymbol{\Sigma}}_1)$ of $\mathbf{X}_i^v$ conditional on $Y_i^v = 0$ and $Y_i^v = 1$ separately. The ROC evaluation procedure in our online system relies on the *working* gaussian mixture assumption that:

$$\mathbf{X} \mid Y = y \sim N(\mu_y, \boldsymbol{\Sigma}_y), \quad y \in \{0, 1\}, \tag{A.4}$$

with the mean and variance parameters estimated by $\hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\mu}}_1, \hat{\boldsymbol{\Sigma}}_0, \hat{\boldsymbol{\Sigma}}_1$. These indicate that $\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \mathbf{X} \mid Y = y$ is approximately $N(\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \boldsymbol{\mu}_y, \hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \boldsymbol{\Sigma}_y \hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}})$. Based on these, we generate labels $Y_1^{\mathsf{sim}}, Y_2^{\mathsf{sim}}, \ldots, Y_M^{\mathsf{sim}}$ with a large M (e.g., $M = 10,000$) from Bernoulli distribution with $\mathrm{P}(Y^{\mathsf{sim}} = 1) = \hat{p}_Y$, and scores $R_1^{\mathsf{sim}}, R_2^{\mathsf{sim}}, \ldots, R_M^{\mathsf{sim}}$ from

$$R^{\mathsf{sim}} \mid Y^{\mathsf{sim}} = y \sim N(\hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \boldsymbol{\mu}_y, \hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}^{\mathsf{T}} \boldsymbol{\Sigma}_y \hat{\boldsymbol{\beta}}_{\mathsf{KOMAP}}).$$

The ROC properties and AUC of the KOMAP score $\hat{Y}^v$ are then estimated by calculating them on $Y_1^{\mathsf{sim}}, Y_2^{\mathsf{sim}}, \ldots, Y_M^{\mathsf{sim}}$ against the generated predictors $R_1^{\mathsf{sim}}, R_2^{\mathsf{sim}}, \ldots, R_M^{\mathsf{sim}}$.

C Article corpus of ONCE

The article corpus comprises data from seven online sources: Wikipedia, the Centers for Disease Control and prevention (CDC), Mayo Clinic, Medscape, MedlinePlus, UpToDate, and the Merck Manual (Merck). All resources are freely available online with the exception of UpToDate, which requires a subscription to view articles in their entirety.

Articles were collected automatically using Python to collect page text for each available article in the index. For multilingual sources (Wikipedia and UpToDate), we select English articles only. For Wikipedia, we process article titles with Named Entity Recognition (NER) [REF] and only include articles whose titles include a term that can be mapped to a UMLS CUI.

In some cases, there are multiple articles per source available for a disease or condition. For example, "Wilson disease: Clinical manifestations, diagnosis, and natural history", "Wilson disease: Diagnostic tests", "Wilson disease: Epidemiology and pathogenesis", and "Wilson disease: Treatment and prognosis". Where there is a discernible pattern and multiple articles share the same main topic, they are concatenated into a single article. In the

Wilson disease example, the article titles are split on the character ":" and concatenated into one "Wilson disease" article.

To ensure that articles in the corpus are maximally relevant, article titles were processed using NILE. Articles in the corpus must have a title which maps to a CUI belonging to the UMLS semantic group *Disorders* (20).

We process articles and their titles using NILE and a curated version of the 2021AB UMLS dictionary containing a subset of the available semantic types. For each article the results of NILE are aggregated and populate i) a list of CUIs for concepts found in the article title and ii) a list of CUIs for concepts found in the article text.

D Detailed validation results for embeddings

D.1 AUC of detecting known pairs of features from the union of two EHR data sources

Pairs	Type	Group	MultiReL	MGB	VA	BioBert	PubmedBert	SAPBert	Num
Code-Code	Similar	PheCode Hierachy	**0.974**	0.901	0.969	0.568	0.613	0.766	4103
		Local Lab Mapping	**0.904**	0.823	0.823	0.654	0.653	0.810	1981
		summary	**0.951**	0.876	0.921	0.596	0.627	0.780	6084
	Related	ddx	**0.803**	0.762	0.780	0.568	0.603	0.629	5986
		Classifies	**0.927**	0.888	0.905	0.625	0.665	0.785	4839
		May Treat (Prevent)	0.767	0.751	**0.788**	0.556	0.629	0.595	4839
		Causative	**0.780**	0.774	0.746	0.553	0.564	0.638	3134
		summary	**0.820**	0.793	0.808	0.576	0.619	0.660	19371
CUI-CUI	Similar	Parent	**0.943**	0.804	0.830	0.656	0.767	0.852	65779
		Sibling	**0.911**	0.811	0.848	0.640	0.784	0.800	36165
		summary	**0.931**	0.806	0.836	0.651	0.773	0.834	101944
	Related	May Treat (Prevent)	0.744	0.741	**0.796**	0.599	0.665	0.608	11523
		Classifies	**0.968**	0.876	0.914	0.653	0.771	0.878	8790
		ddx	**0.919**	0.760	0.792	0.661	0.754	0.672	6806
		Method_of	**0.866**	0.642	0.714	0.460	0.37	0.703	4347
		Causative	**0.889**	0.791	0.839	0.563	0.702	0.809	1476
		summary	**0.862**	0.770	0.818	0.606	0.710	0.716	33212
CUI-Code	Similar	CUI-PheCode	**0.942**	0.874	0.900	0.551	0.611	0.795	16396
		CUI-RXNORM	**0.999**	0.943	0.987	0.690	0.801	0.981	1139
		CUI-LOINC	**0.965**	0.859	0.898	0.540	0.593	0.849	611
		CUI-CCS	**1.000**	0.955	0.986	0.842	0.917	0.990	66
		summary	**0.947**	0.878	0.906	0.560	0.623	0.809	15026

Table 3: Detailed AUCs of between-vector cosine similarity in detecting known similar or related pairs of codified concepts (COD), NLP CUI concepts, and COD vs NLP CUI concepts for embeddings trained by MultiReL, MGB alone with VA pre-training, VA alone with MGB pre-training, BioBert, PubmedBert, and SAPBert. Concepts of each pair lie in the union of MGB and VA embedding spaces.

D.2 AUC of detecting known pairs of site-unique features

Pairs	Type	Group	MultiReL	MGB VA	BioBert	PubmedBert	SAPBert	Num
Code-Code	Similar_s	PheCode Hierachy	**0.911**	0.829	0.568	0.653	0.816	1981
		Local Lab Mapping	0.915	**0.938**	0.506	0.400	0.615	48
		summary	**0.911**	0.832	0.653	0.647	0.811	2029
	Related_s	ddx	**0.904**	0.881	0.614	0.730	0.802	30
		Classifies	**0.988**	0.978	0.818	0.830	0.906	50
		May Treat (Prevent)	0.614	0.525	0.499	**0.619**	0.560	32
		Causative	0.741	**0.762**	0.670	0.582	0.619	57
		summary	**0.819**	0.802	0.671	0.689	0.725	169
CUI-CUI	Similar_s	Parent	**0.930**	0.780	0.653	0.768	0.829	15971
		Sibling	**0.918**	0.811	0.683	0.801	0.826	1606
		summary	**0.929**	0.783	0.655	0.771	0.828	17577
	Related_s	May Treat (Prevent)	0.681	0.704	0.586	0.663	**0.713**	40
		Classifies	**0.955**	0.814	0.711	0.818	0.868	129
		ddx	**0.928**	0.714	0.631	0.833	0.726	48
		Method_of	**0.791**	0.536	0.412	0.642	0.640	709
		Causative	**0.844**	0.766	0.620	0.743	0.817	162
		summary	**0.820**	0.617	0.495	0.687	0.700	1088
CUI-Code	Similar_s	CUI-PheCode	**0.919**	0.898	0.421	0.263	0.525	216
		CUI-RXNORM	**1.000**	0.771	0.698	0.875	0.963	22
		CUI-LOINC	**0.917**	0.786	0.554	0.548	0.833	201
		CUI-CCS						0
		summary	**0.922**	0.8041	0.496	0.423	0.688	439

Table 4: Detailed AUCs of between-vector cosine similarity in detecting known similar or related pairs of codified concepts (COD), NLP CUI concepts, and COD vs NLP CUI concepts for embeddings trained by MultiReL, MGB alone with VA pre-training, VA alone with MGB pre-training, BioBert, PubmedBert, and SAPBert. Concepts of each pair only exist in either MGB or VA site.

E Detailed phenotyping results for separate diseases

E.1 AUC per disease across sites

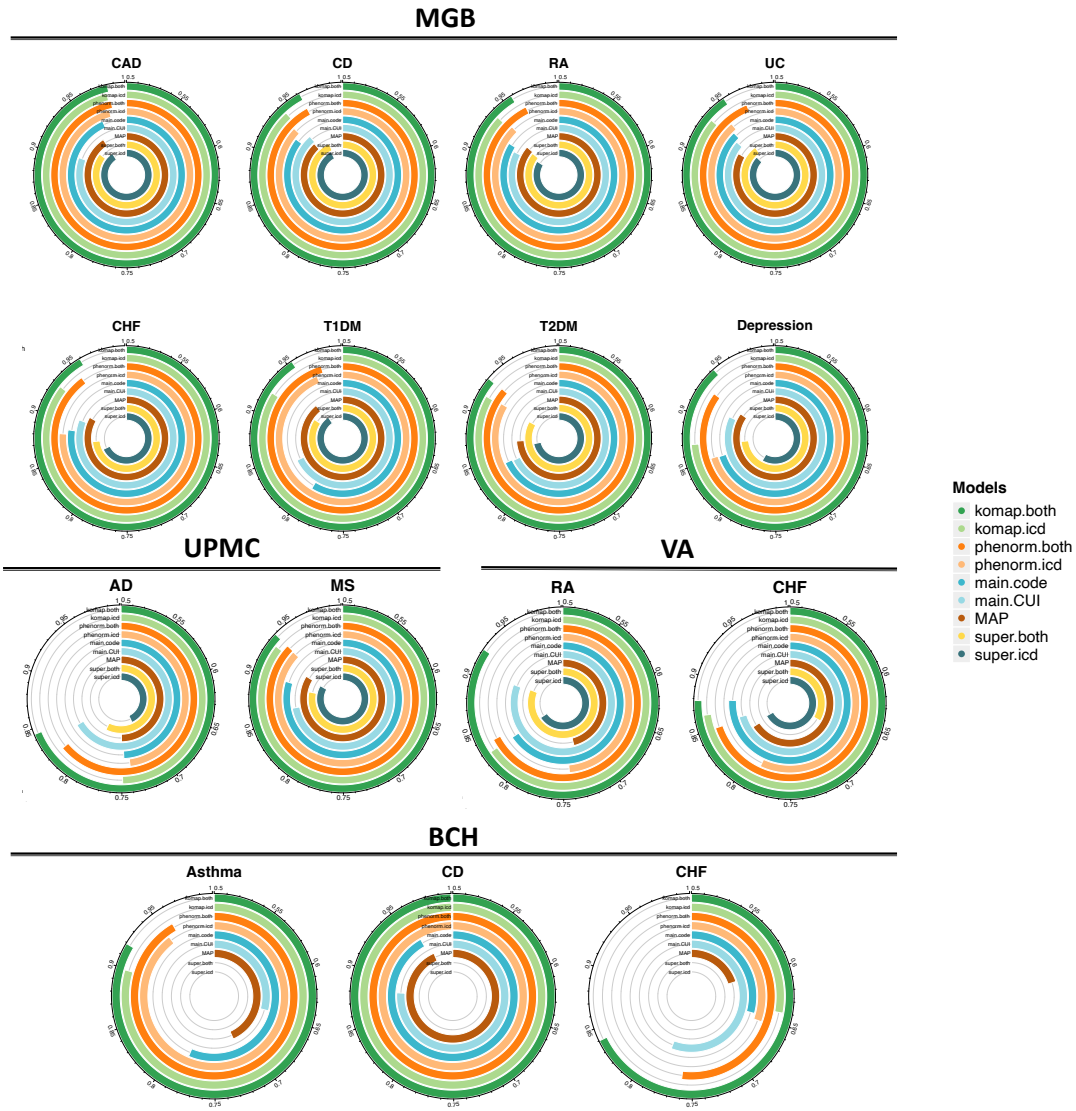

Figure 10: AUC for the 11 phenotypes generated from KOMAP with both codified data and CUIs; PheNorm with both codified data and CUIs; supervised model with both codified data and CUIs; KOMAP with only codified data; PheNorm with only codified data; supervised model with only codified data; main ICD; main CUI and MAP.

F Visualization of the coefficients

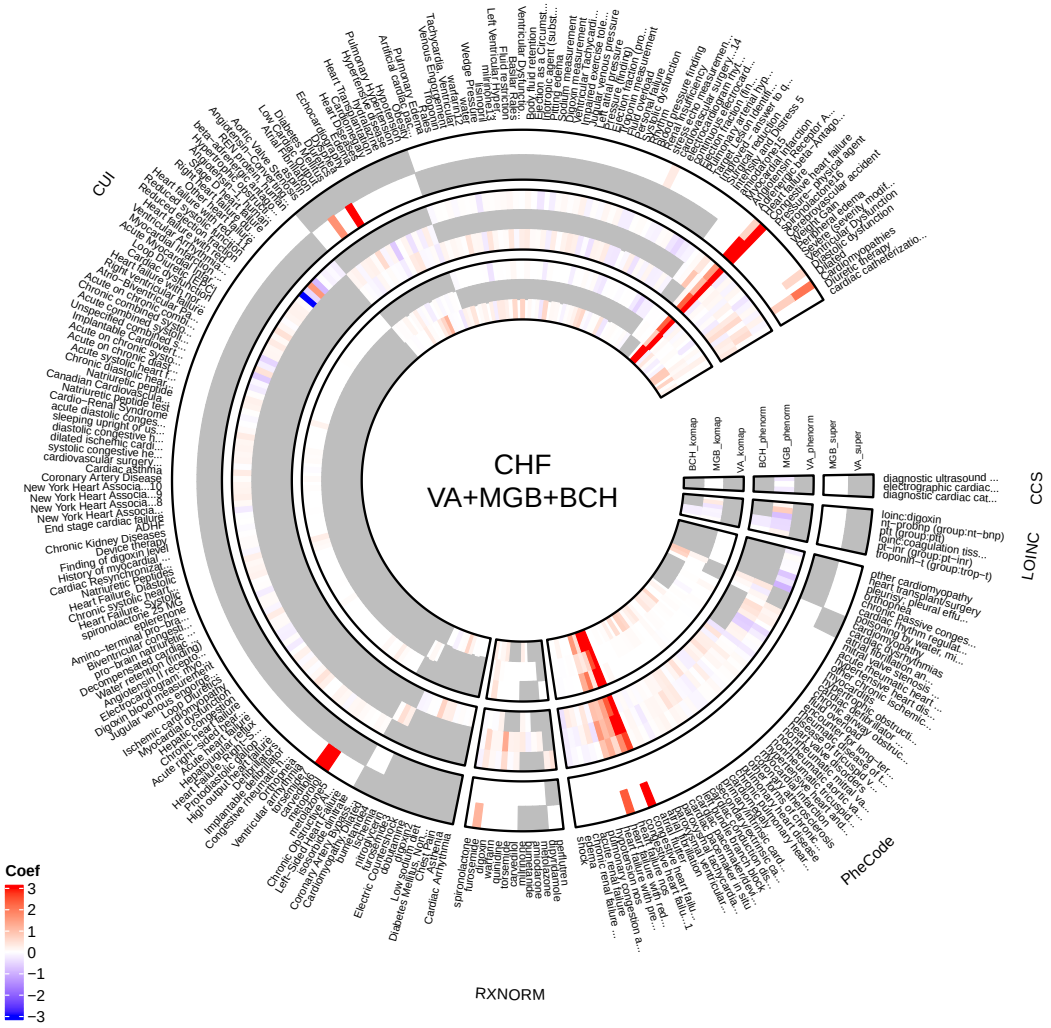

Figure 11: Coefficients of features for HF estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data from VA, MGB and BCH sites.

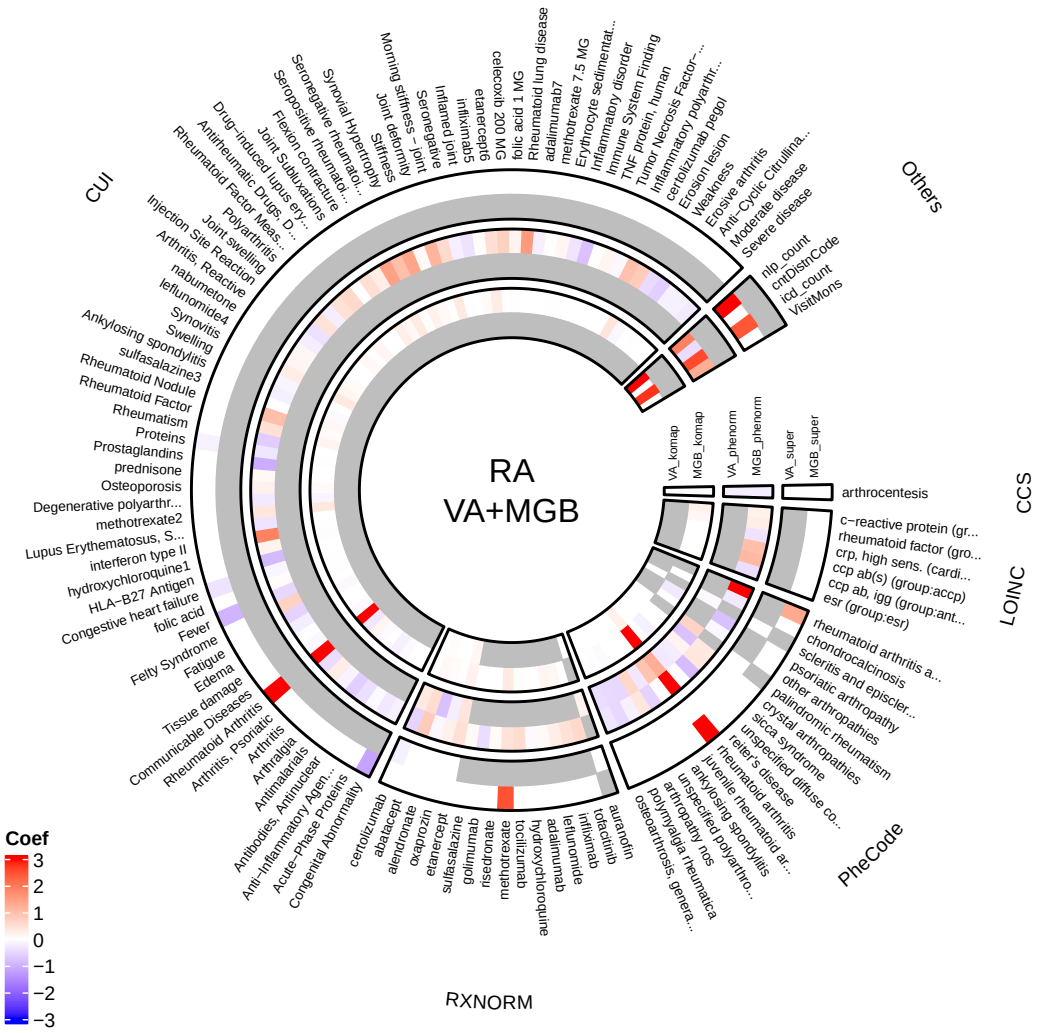

Figure 12: Coefficients of features for RA estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data from VA and MGB.

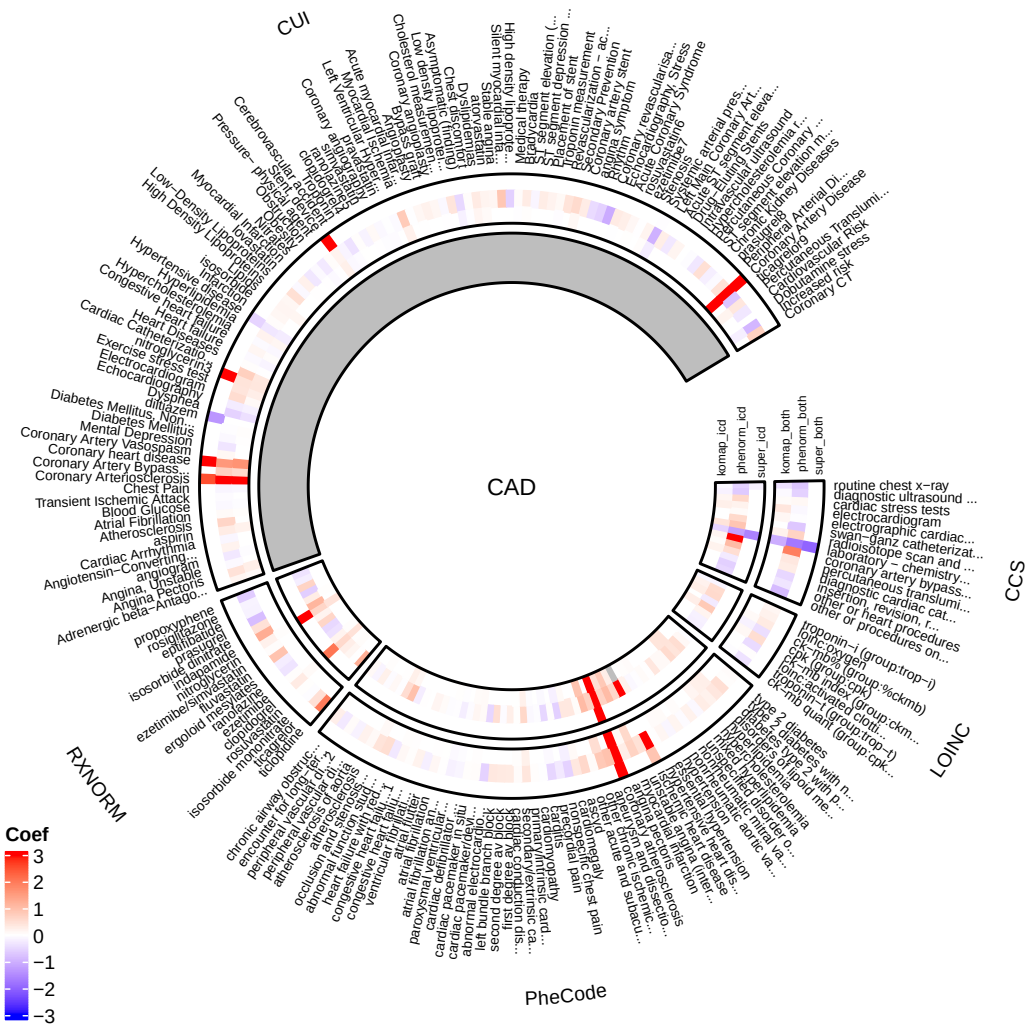

Figure 13: Coefficients of features for CAD estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

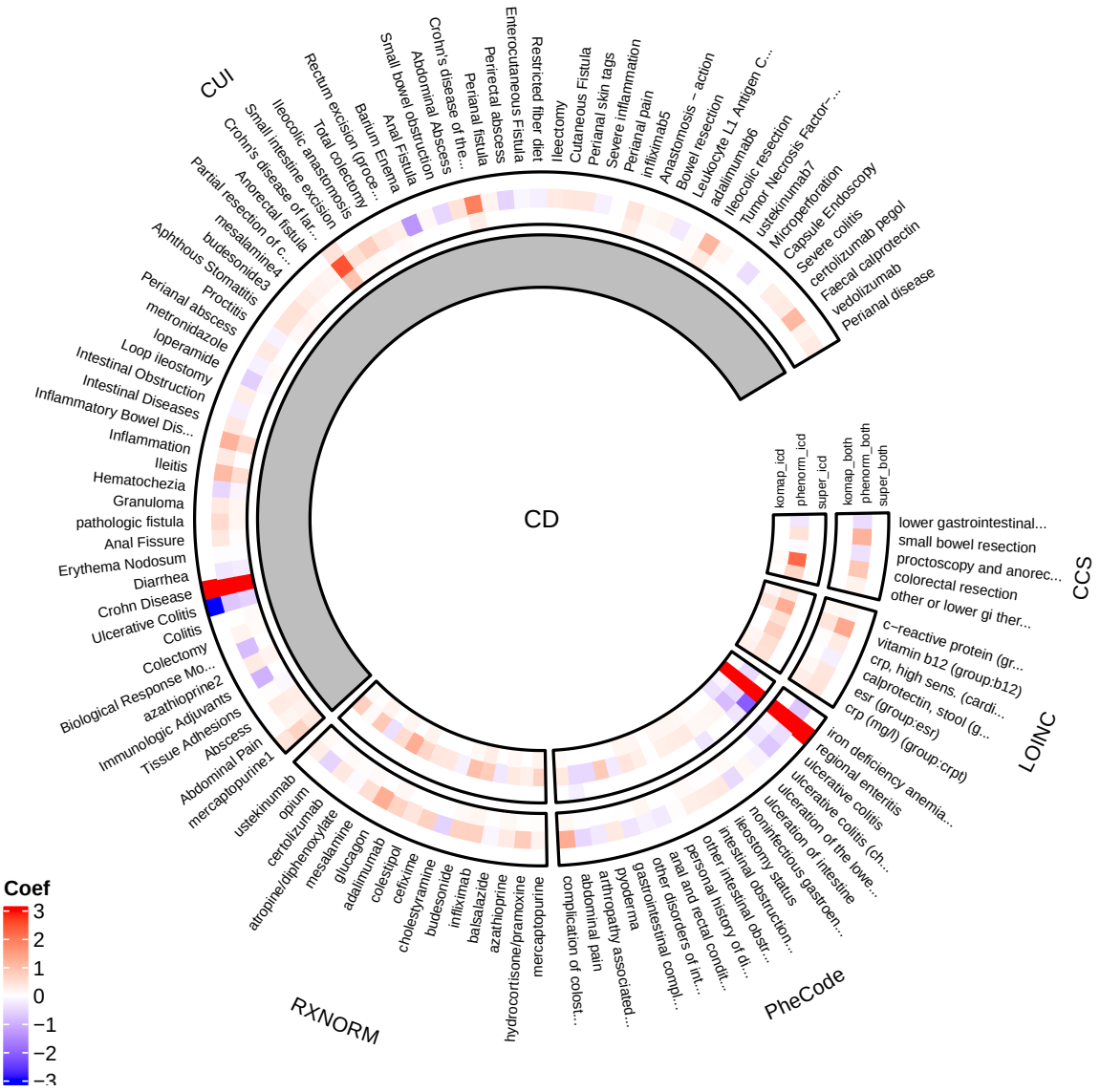

Figure 14: Coefficients of features for CD estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

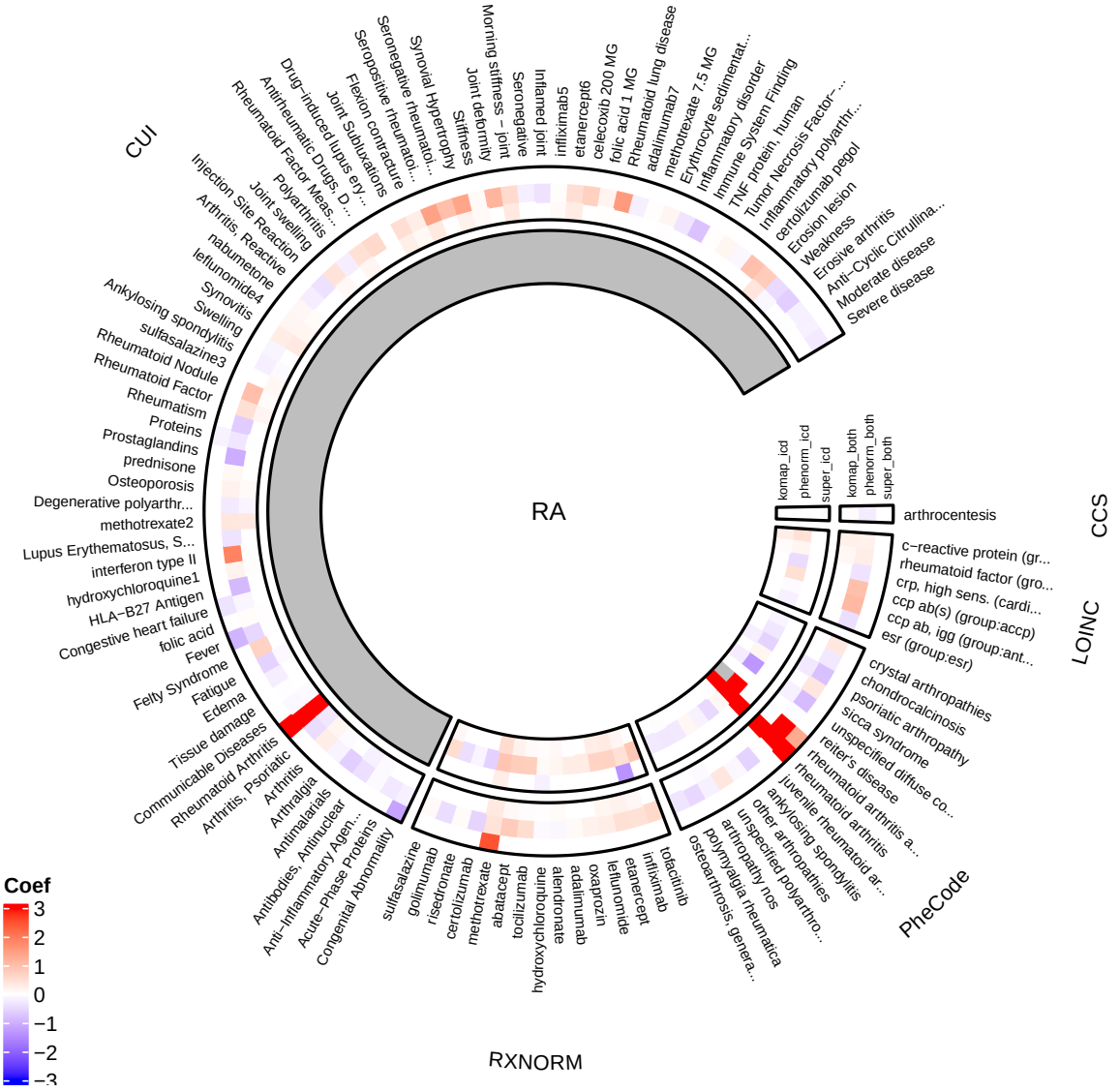

Figure 15: Coefficients of features for RA estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

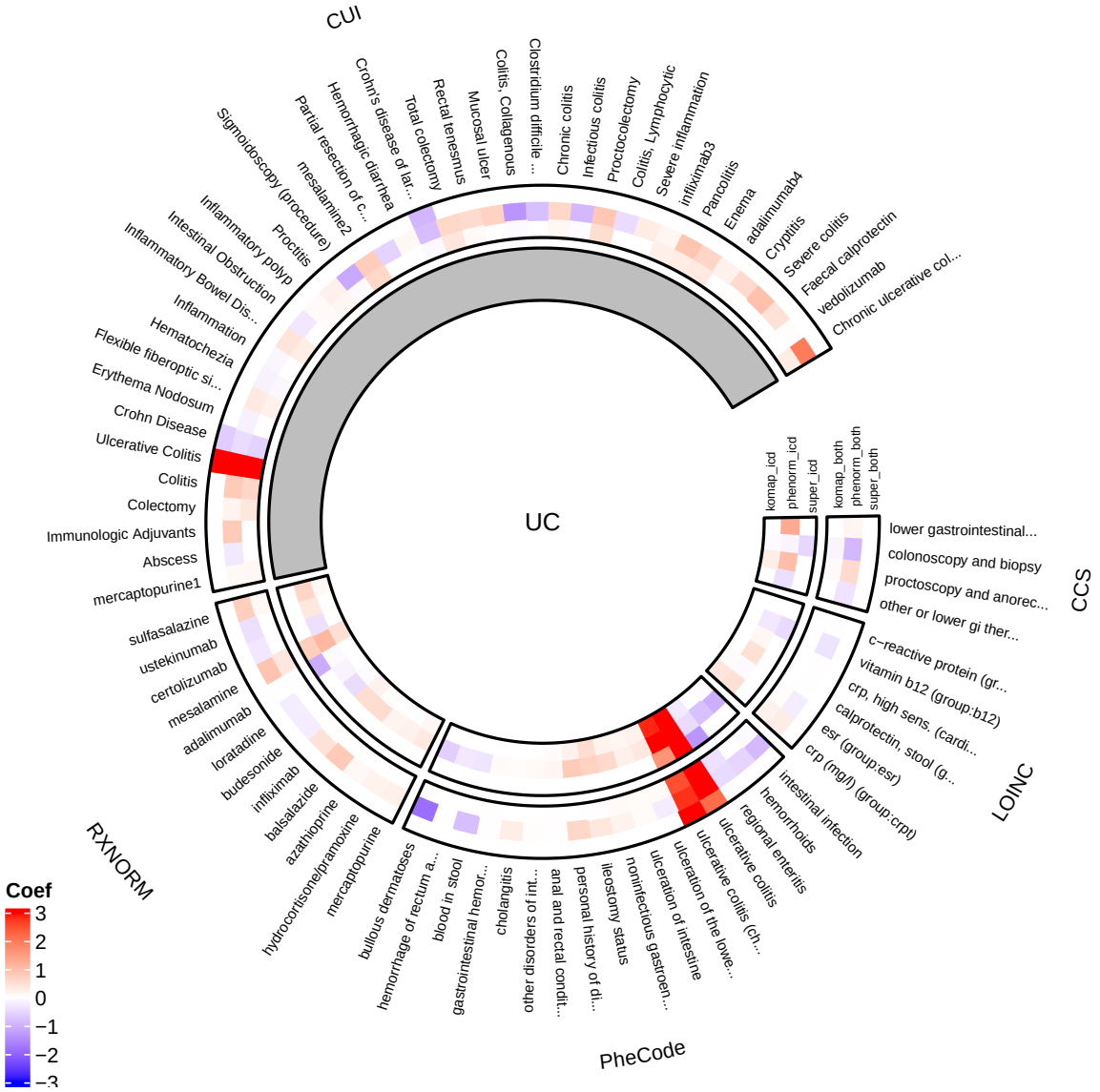

Figure 16: Coefficients of features for UC estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

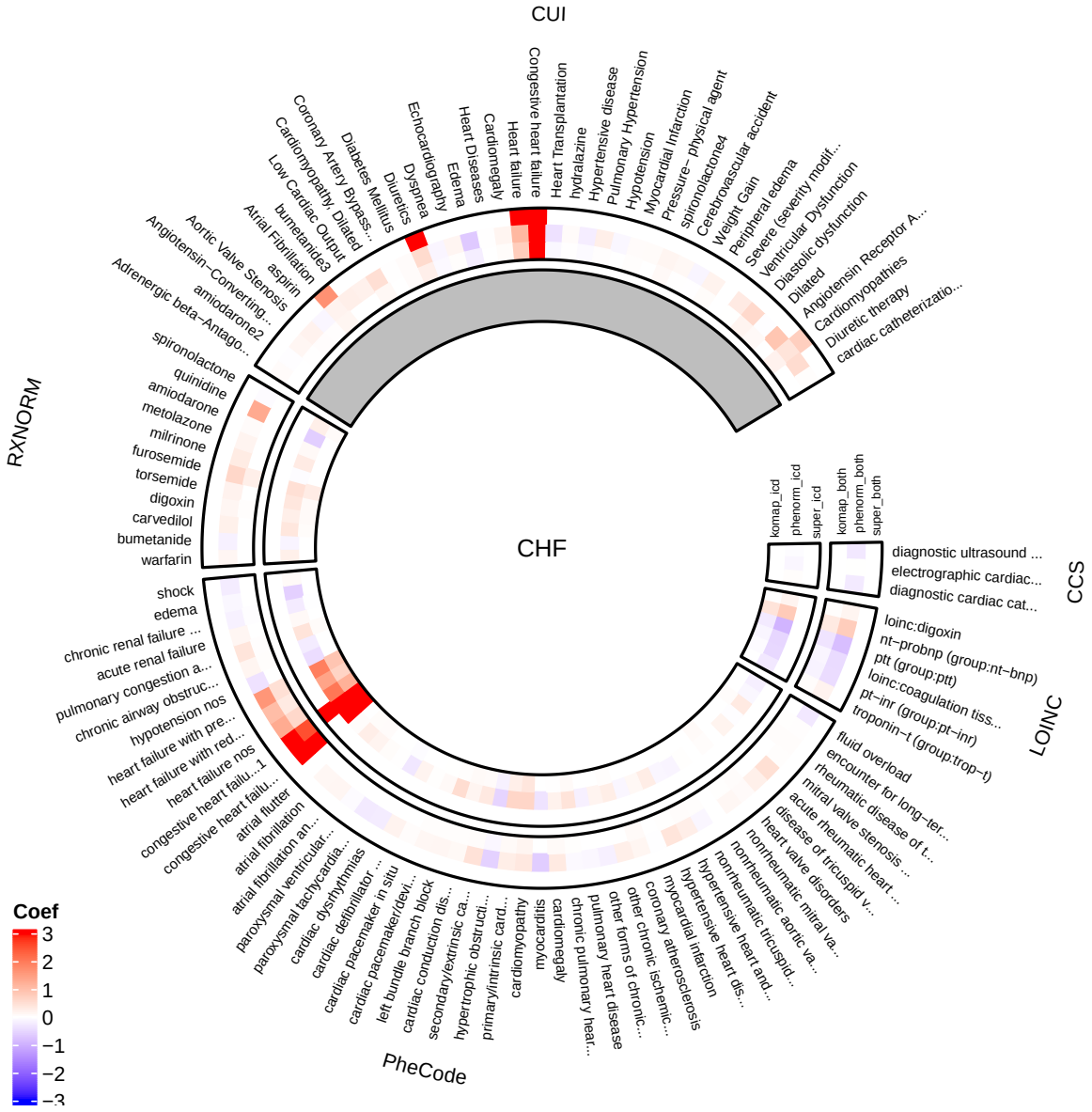

Figure 17: Coefficients of features for CHF estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

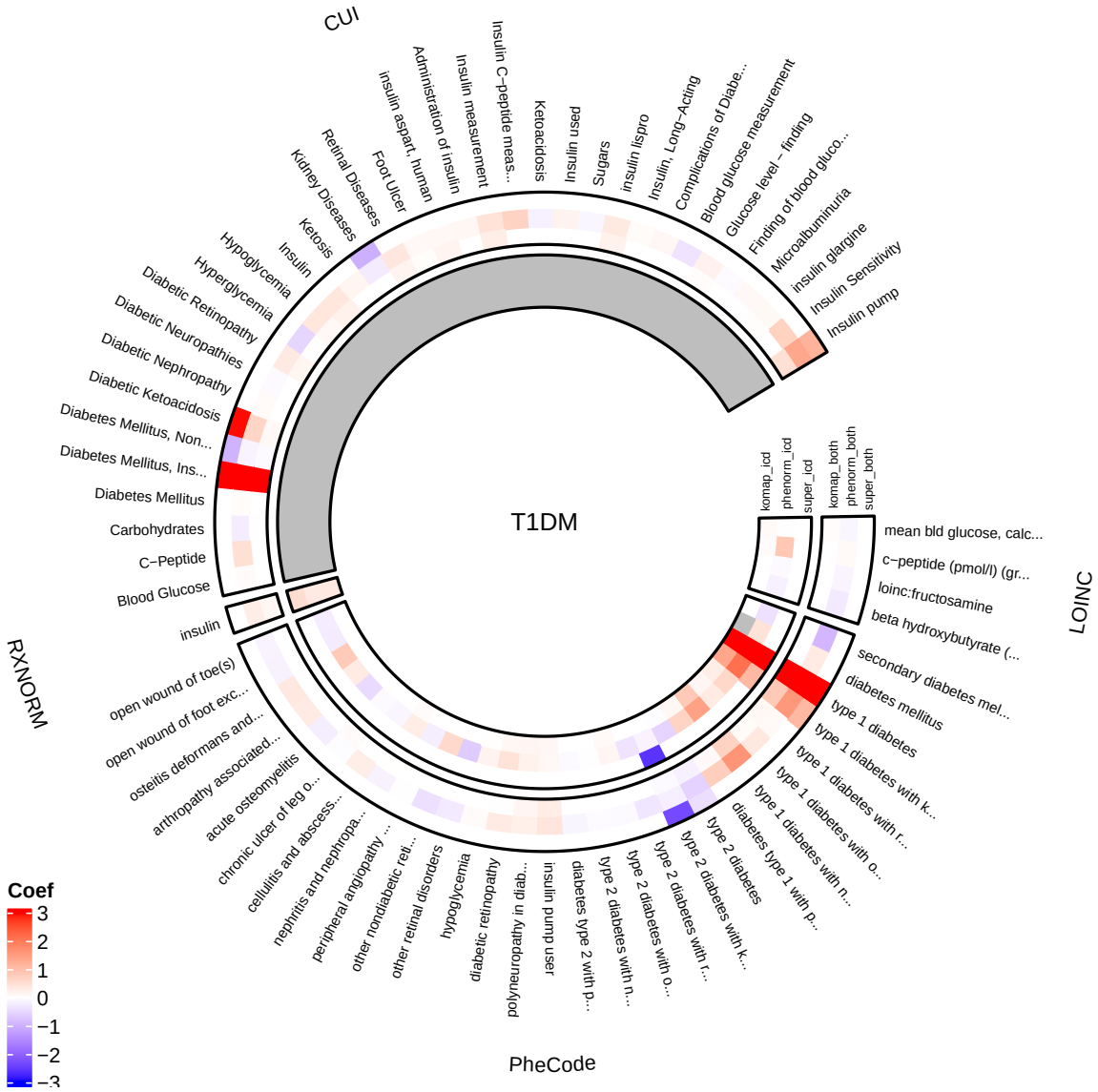

Figure 18: Coefficients of features for T1D estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).

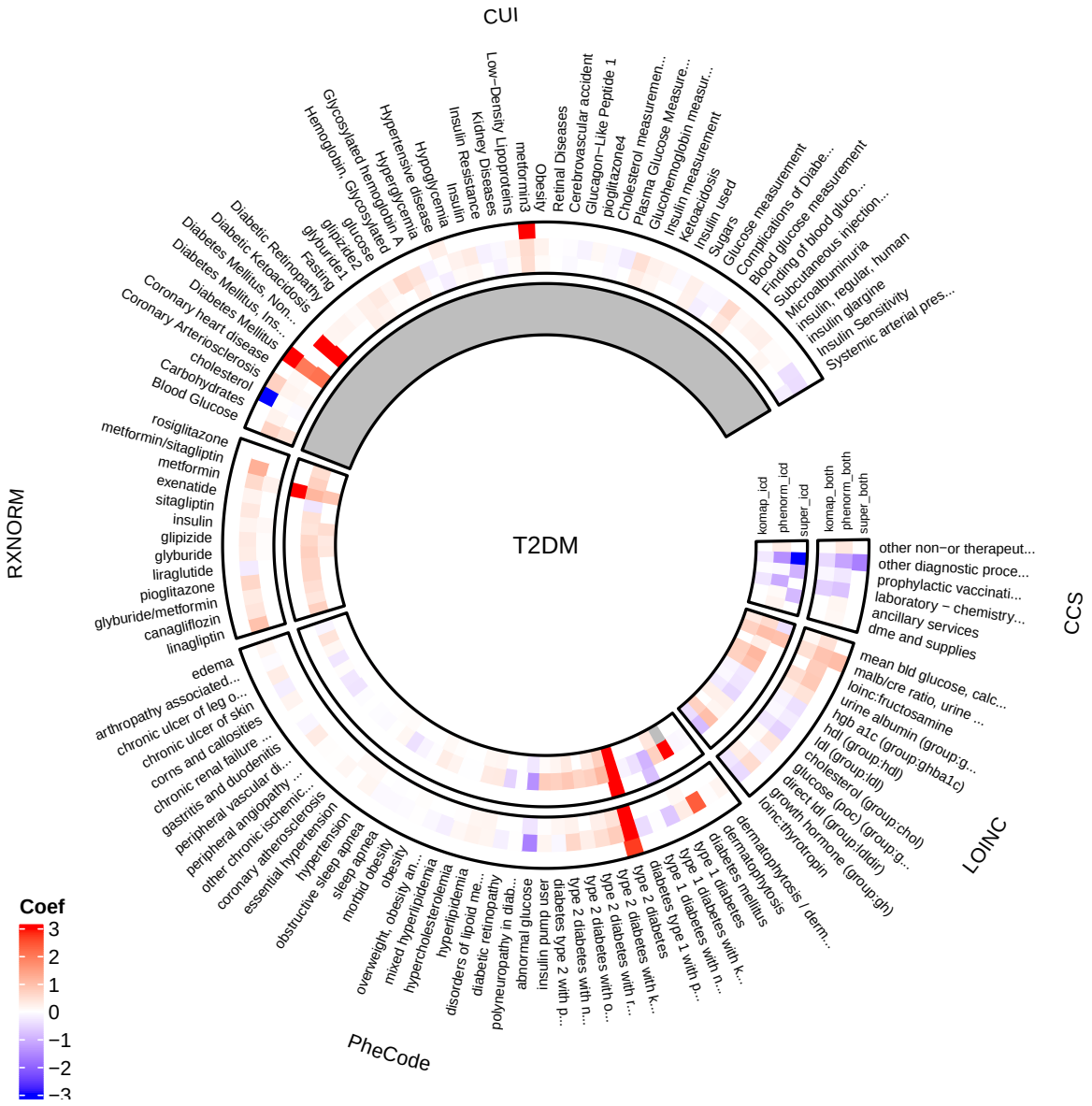

Figure 19: Coefficients of features for T2D estimated by KOMAP, PheNorm and supervised-learning model with codified and NLP data (outer circles) or only with codified data (inner circles).