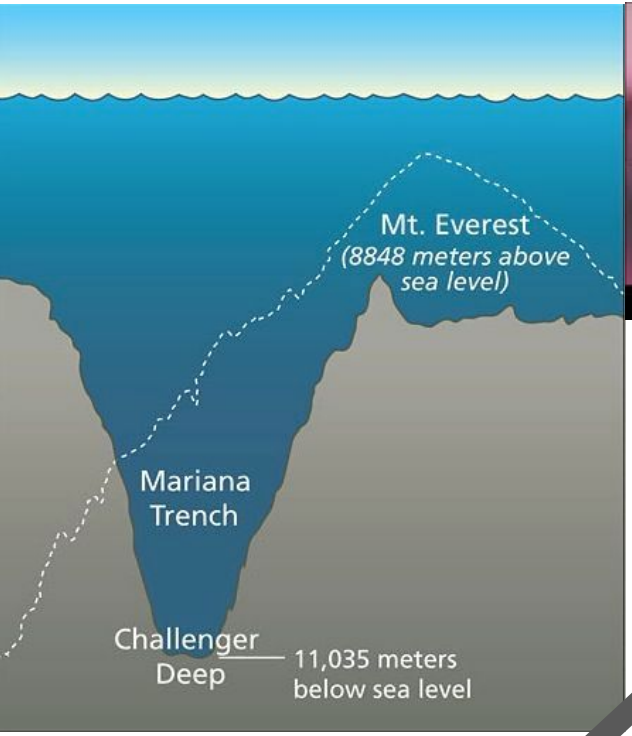
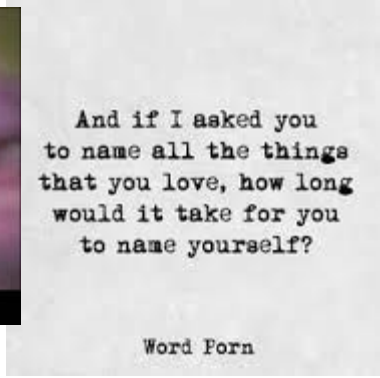
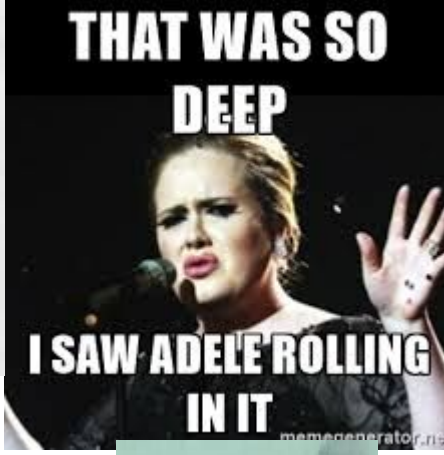




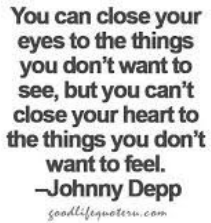


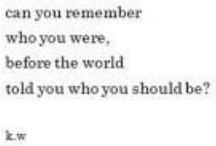
Deep Patient

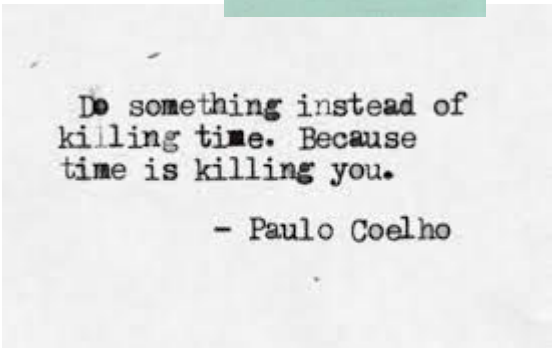
Deeper$wag

13000 meters

A mother's love



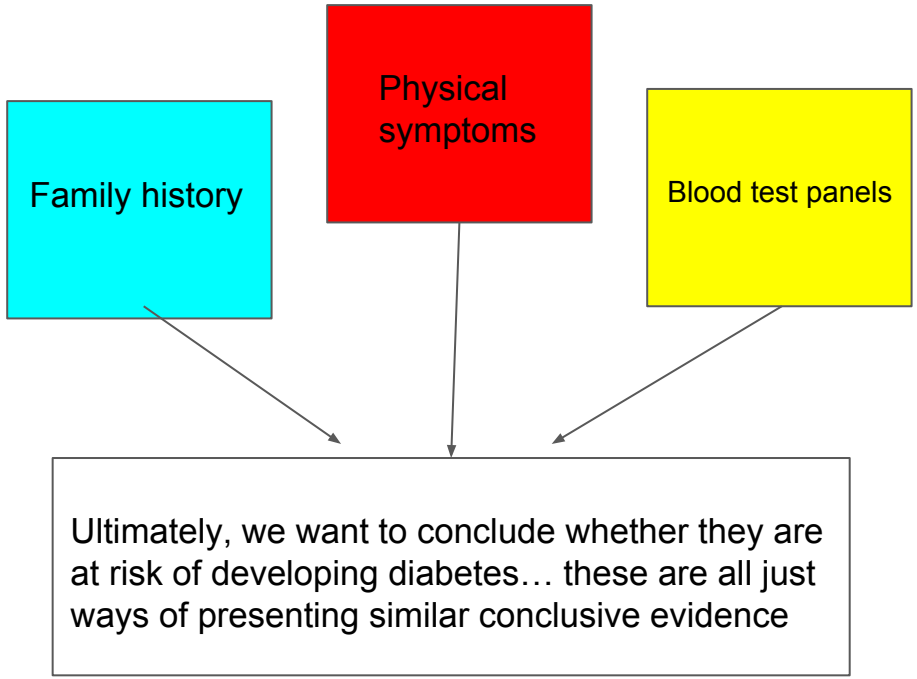




Introduction

DeepPatient aims to learn representations of patients using diverse sources of data for downstream tasks

- Capture “key features”
- Data sources are inconsistent, sparse, noisy … we just want to know the big picture!
- Many different ways to capture information about a disease (i.e. diabetes)



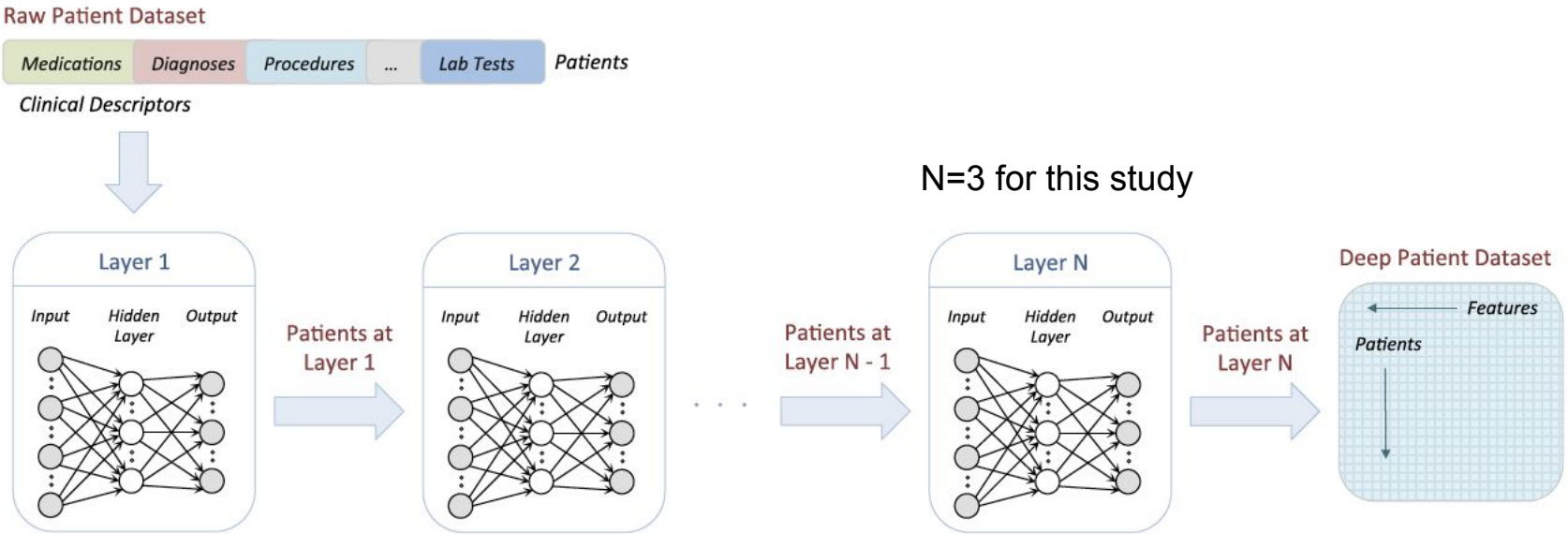
Data featurization

Aggregate sources of data

- Electronic health records (EHR)
  - Demographic details
  - Clinical descriptors (ICD-9 codes)
  - Medications
  - Procedures
  - Lab tests
  - Clinical notes (free-text)

- Deep denoising autoencoder architecture
  - Multiple “autoencoder” layers
  - Trained greedily (i.e. sequentially)

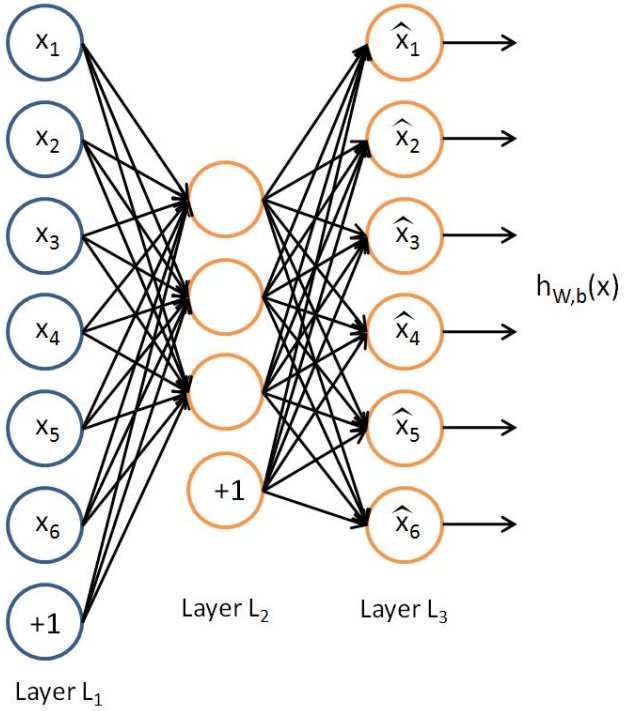
Data featurization



Problem formulation: why autoencoders?

Main idea: feature learning for downstream tasks (i.e. representation learning)

- Primarily a proof of concept
- Evaluated based on performance on future disease prediction vs. learning from raw features
- Compact representation serves as better predictor under their comparison



Concerns regarding model

DeepPatient: first use of deep learning architectures on EHR data

- Wrong baseline? Simple models on raw features not as informative
- Compare against direct deep learning model on tasks?
- Unclear if this representation is actually better for standard tasks
    - Human prediction baseline?

How does the patient data get in a format the algorithm can use?

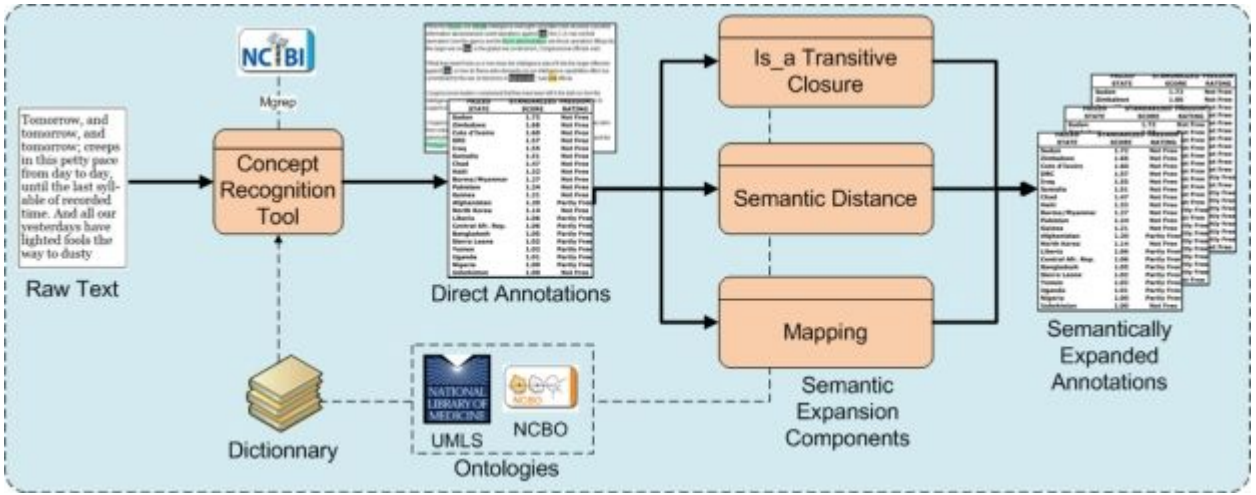
- Start with a data warehouse of structured, semi-structured and unstructured data, turn this into a feature vector.
- The entire experiment is affected by choices made here.

Structured data

- Age, gender, race, etc.

Semi-structured data

- Text, but with some structure.
- Diagnoses, medications, procedures, tests, etc. are normalized using the Open Biomedical Annotator.



Unstructured data

- How can free-text clinical notes be featurized?
- Latent Dirichlet allocation (LDA) is a very popular topic modelling method.
  - The intuition: A generative model where each document is a mixture of hidden (latent) topics.
  - Each note is modelled as a continuous 300-dimensional vector.
  - All the notes for a patient are averaged.

Electronic Health Records

Clinical Notes

Diagnoses

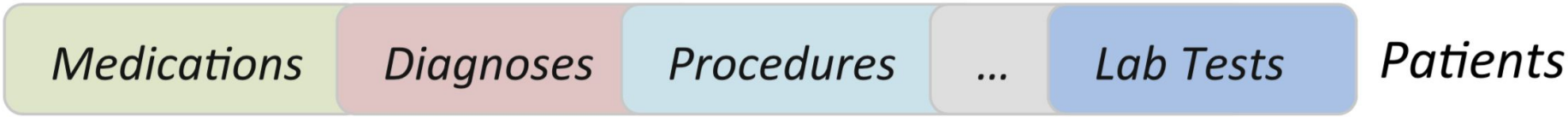
Medications

Laboratory Tests

Demography

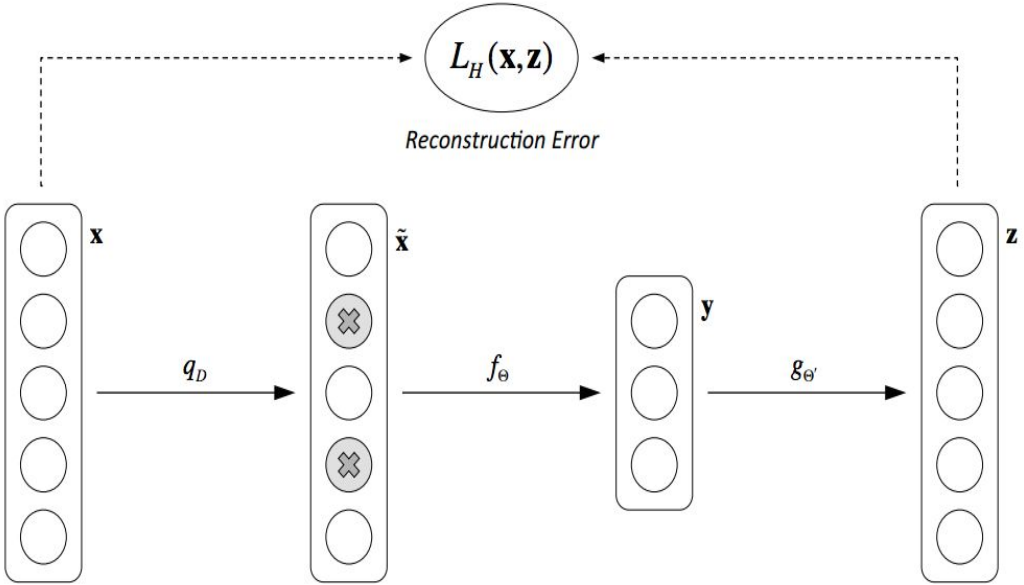
Etc.

Raw Patient Dataset



Clinical Descriptors

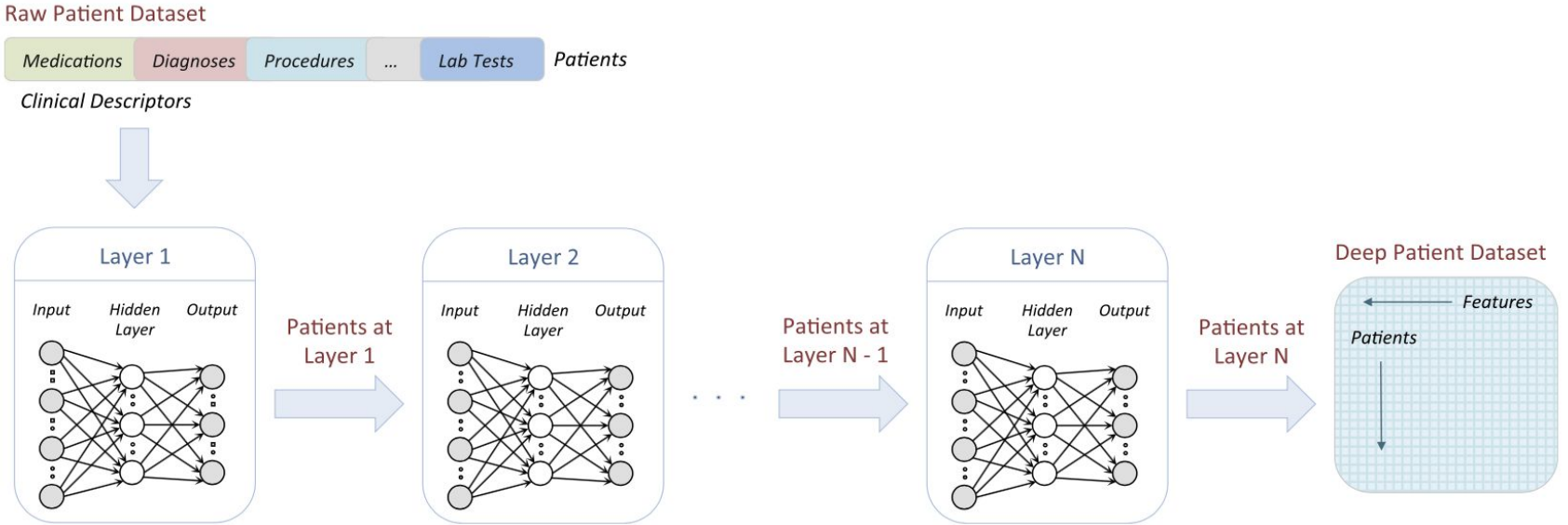
Denoising Autoencoders



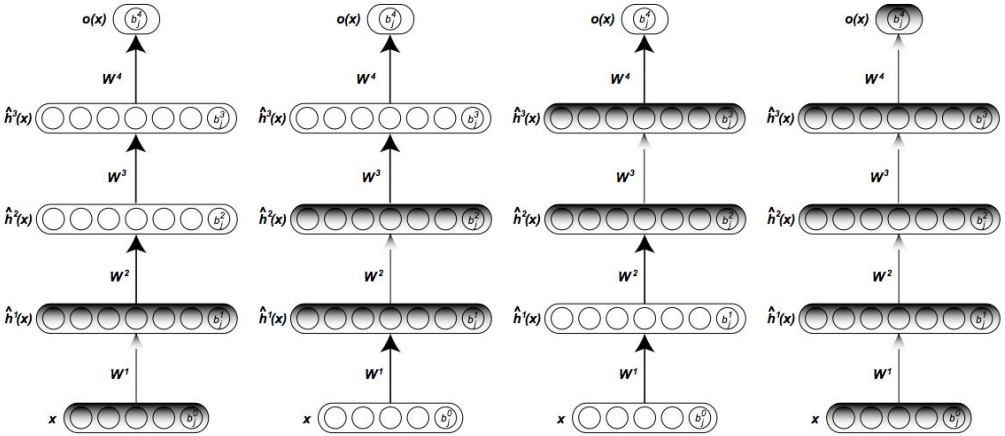
Reconstruct input from corrupted data.

1. Corruption acts as a stochastic regularizer (forces generalization by preventing learning Identity function)
2. Has a natural interpretation for EHR- missing documents

Stacked (Deep) Denoising Autoencoders



Parameter Fitting



(a) First hidden layer pre-training (b) Second hidden layer pre-training (c) Third hidden layer pre-training (d) Fine-tuning of whole network

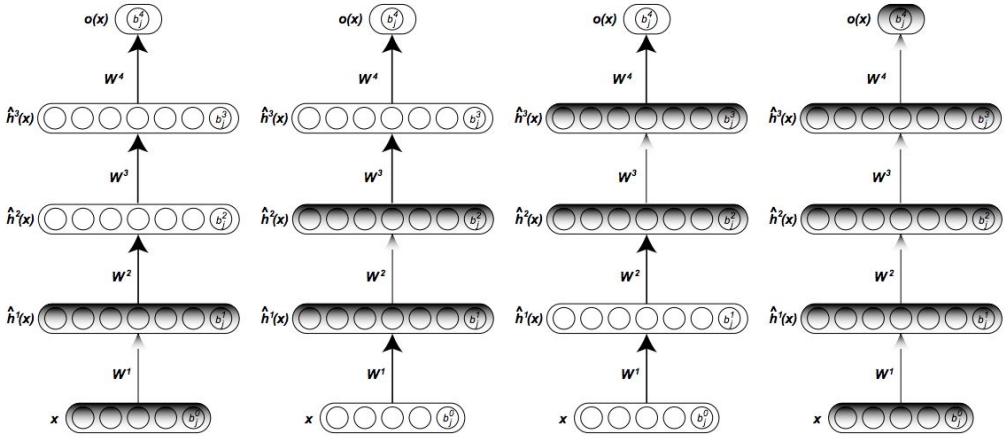
- Weights (Θ)

Optimization criterion:
Binary cross-entropy loss

$$L_H(\boldsymbol{x}, \boldsymbol{z}) = -\sum_{k=1}^{d} [x_k \, logz_k + (1 - x_k)\log(1 - z_k)].$$

Technique: Mini-batch stochastic GD

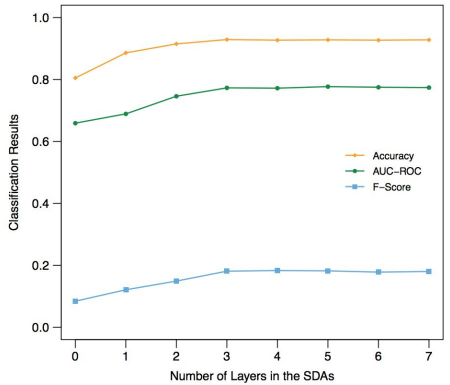
Hyperparameter Tuning



(a) First hidden layer pre-training (b) Second hidden layer pre-training (c) Third hidden layer pre-training (d) Fine-tuning of whole network

- Number of layers: 3
- Hidden activation units: 500
- Corruption frequency: 0.05

Optimization Criterion:
Supervised classification performance



Questions

1. Limited hyperparameter tuning
    a. Did not consider other activation functions (than sigmoid)
    b. Shared hyperparameters for each layer (number of hidden activation units, corruption frequency)
2. Greedy training
   Optimize parameters one layer after another instead of all at once.

Results and Evaluation of DeepPatient

- **2 clinical evaluation metrics:**
  - Disease classification (evaluation by disease)
  - Patient disease tagging (evaluation by patient)
- **Benchmark:**
  - Deep patient representation compared with PCA, k-means clustering, GMM, ICA
  - RawFeat -- patient data represented with original descriptors

Evaluation by Disease

Time Interval = 1 year (76,214 patients)			
		Classification Threshold = 0.6	
Patient Representation	AUC-ROC	Accuracy	F-Score
RawFeat	0.659	0.805	0.084
PCA	0.696	0.879	0.104
GMM	0.632	0.891	0.072
K-Means	0.672	0.887	0.093
ICA	0.695	0.882	0.101
DeepPatient	**0.773***	**0.929***	**0.181***

DeepPatient metrics are superior to all other data representations

Disease-Specific Results

Time Interval = 1 year (76,214 patients)			
	Area under the ROC curve		
Disease	RawFeat	PCA	DeepPatient
Diabetes mellitus with complications	0.794	0.861	**0.907**
Cancer of rectum and anus	0.863	0.821	**0.887**
Cancer of liver and intrahepatic bile duct	0.830	0.867	**0.886**
Regional enteritis and ulcerative colitis	0.814	0.843	**0.870**
Congestive heart failure (non-hypertensive)	0.808	0.808	**0.865**
Attention-deficit and disruptive behavior disorders	0.730	0.797	**0.863**
Cancer of prostate	0.692	0.820	**0.859**
Schizophrenia	0.791	0.788	**0.853**
Multiple myeloma	0.783	0.739	**0.849**
Acute myocardial infarction	0.771	0.775	**0.847**

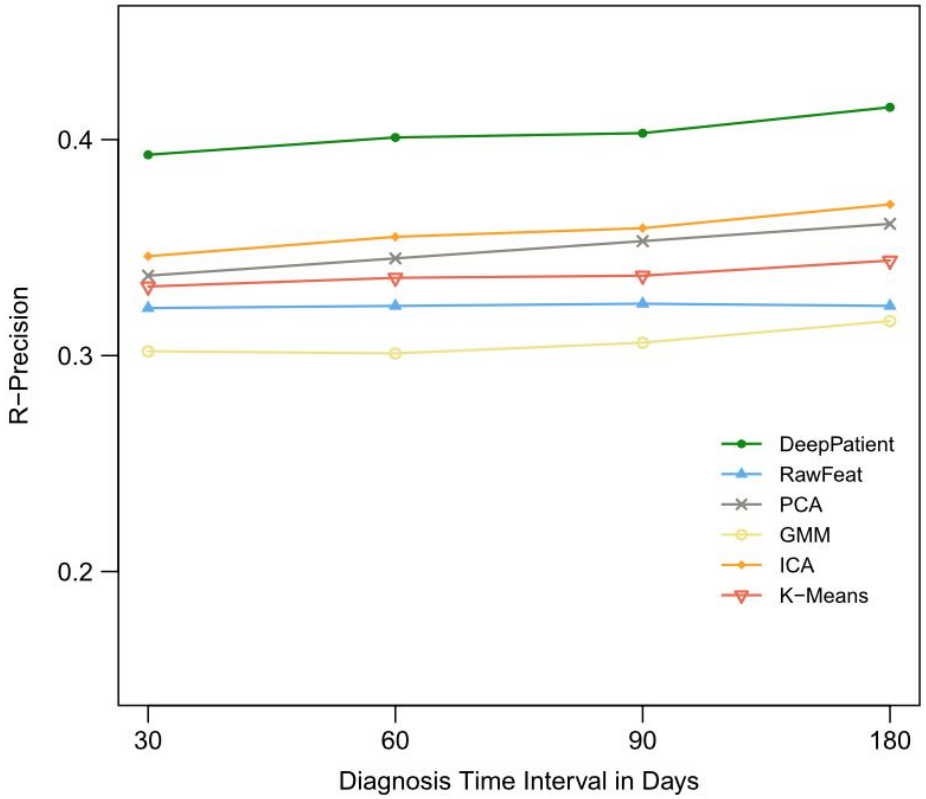
DeepPatient outperforms RawFeat and PCA on 77 diseases out of 78

Patient Disease Tagging

Time Interval	Metrics	UppBnd	Patient Representation			
			RawFeat	PCA	ICA	DeepPatient
30 days *(16,374 patients)*	Prec@1	1.000	0.319	0.343	0.345	**0.392***
	Prec@3	0.492	0.217	0.251	0.255	**0.277***
	Prec@5	0.319	0.191	0.214	0.215	**0.226***
60 days *(21,924 patients)*	Prec@1	1.000	0.329	0.349	0.353	**0.402***
	Prec@3	0.511	0.221	0.254	0.259	**0.282***
	Prec@5	0.335	0.199	0.216	0.219	**0.230***
90 days *(25,220 patients)*	Prec@1	1.000	0.332	0.353	0.360	**0.404***
	Prec@3	0.521	0.243	0.257	0.262	**0.285***
	Prec@5	0.345	0.201	0.219	0.220	**0.232***
180 days *(33,607 patients)*	Prec@1	1.000	0.331	0.361	0.363	**0.418***
	Prec@3	0.549	0.246	0.261	0.265	**0.290***
	Prec@5	0.370	0.207	0.221	0.224	**0.236***

DeepPatient shows a 5-15% improvement over every other method across all times

Patient Disease Tagging (cont'd)



Critiques of the supervised algorithm

Train: 200,000 patients (71%), Val: 5,000 patients (2%), Test: 76,214 patients (27%)

Unbalanced Validation set (2%) used to tune:

- The supervised model hyper-parameters (# trees)
- The feature extraction models hyper-parameters:
  - # neurons/layer for the denoising autoencoders
  - # principal components for the PCA
  - # clusters for K-means

Why Random Forest?

Pros:

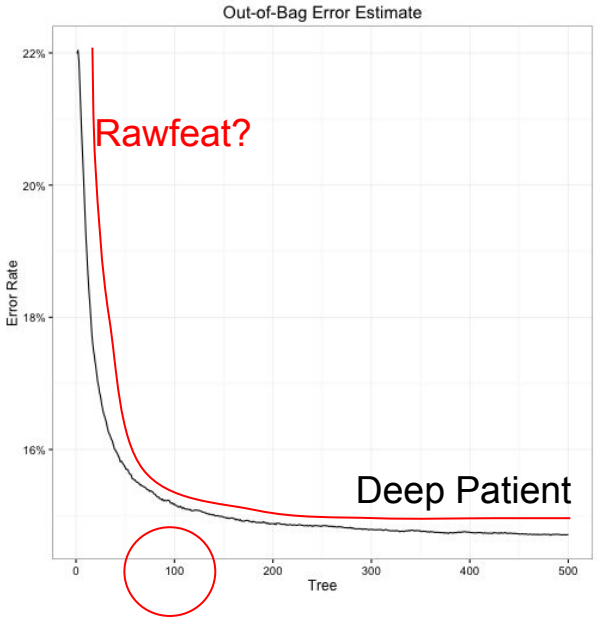
- Random Forest is faster to train and less sensitive to outliers than other methods
- Allows to compare with other benchmarks, interpretability

Cons:

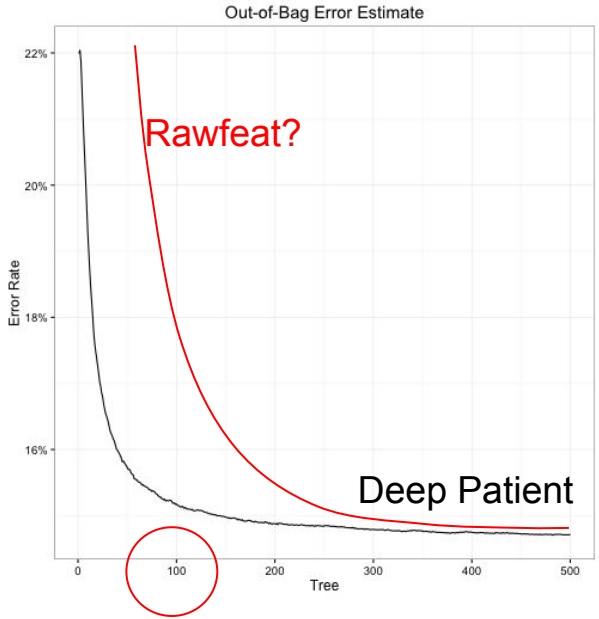
- They could have used other predictive models for the supervised task (SVM, Neural nets)
- Why do we need feature dimension reduction in the first place? What about using deep learning on raw features to make prediction? Deep learning is able to handle 40,000 raw features.

On the number of trees

The number of trees was calibrated for the 500 output features of Deep Patient: 100 trees.
Valid for PCA (100 principal components), Kmeans (500 clusters), GMM (200 mixtures).
Maybe not enough for the raw features (40,000): underfitting?



Not very clear



Clinical applicability of future disease prediction

Clinical applicability has yet to be proven

- They predict ICD9 codes that might be present in previous records
- They excluded rare diagnoses (raw feat might be better at this task)
- They excluded negation tags: absence of information relevant in bioinformatics
- Assume that the physician judgement is the “ground truth”. If the algorithm predicts a diagnoses that the physician didn’t think of: False positive.
- Could be used for automatic surveillance but unable to outperform the physician

Conclusion

1. Curated of data from each patient into a vector
    a. Medications, diagnoses, procedures, lab tests (but not their results!), ….
    b. Used NLP techniques to extract information from text
2. Unsupervised learning of dense representation
    a. Used a stack of denoising autoencoders to learn compact representation
    b. Three layer network, each greedily optimized
    c. Each maps to 500 features
3. Evaluated the utility of their representation by fitting predictive models
    a. Show strong predictive performance on a diverse set of diseases
    b. Outperform other methods of dimensionality reduction

Their Vision

Develop a system that can be used to augment clinician judgements

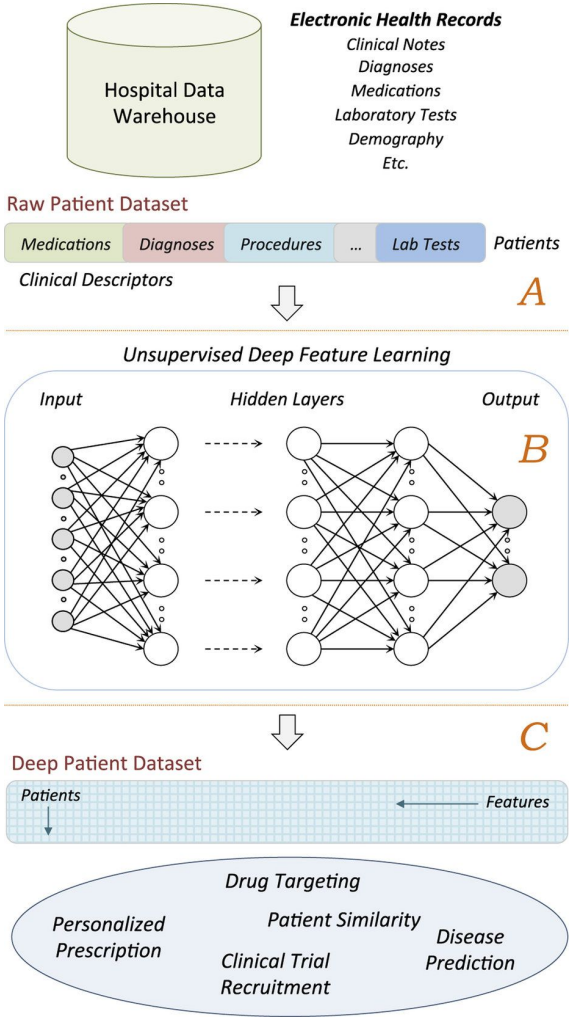
- Predicting patient risk factors
- Personalized prescriptions and treatment recommendations
- Patient clustering and similarity
- Data sharing between hospitals, better models through combining data
- Identification of diseases common in other areas

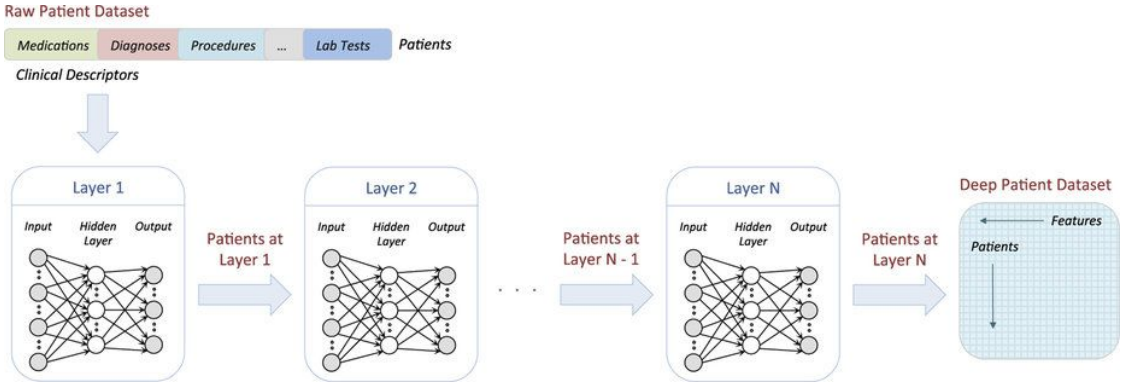
Other Applications

- Clinical trial recruitment
- Setting insurance premiums

Discussion

- First application to a high risk area we've seen so far. Who should be responsible if the AI makes a mistake?
- Is their representation valuable if it is not interpretable?





In this study N=3