PREDICTIVE UNCERTAINTY QUANTIFICATION AND
EXPLAINABLE MACHINE LEARNING IN HEALTHCARE

A Dissertation
Submitted to
the Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree of
DOCTOR OF PHILOSOPHY

by
Djordje Gligorijevic
August 2018

Examining committee members:

Dr Zoran Obradovic, Advisory Chair, Department of Computer and Information Sciences
Dr Slobodan Vucetic, Department of Computer and Information Sciences
Dr Professor Kai Zhang, Department of Computer and Information Sciences
Dr Rob J. Kulathinal, External Member, Department of Biology
ABSTRACT

Predictive modeling is an ever-increasingly important part of decision making. The advances in Machine Learning predictive modeling have spread across many domains bringing significant improvements in performance and providing unique opportunities for novel discoveries. A notably important domains of the human world are medical and healthcare domains, which take care of peoples’ wellbeing. And while being one of the most developed areas of science with active research, there are many ways they can be improved. In particular, novel tools developed based on Machine Learning theory have drawn benefits across many areas of clinical practice, pushing the boundaries of medical science and directly affecting well-being of millions of patients. Additionally, healthcare and medicine domains require predictive modeling to anticipate and overcome many obstacles that future may hold. These kinds of applications employ a precise decision–making processes which requires accurate predictions. However, good prediction by its own is often insufficient. There has been no major focus in developing algorithms with good quality uncertainty estimates. Ergo, this thesis aims at providing a variety of ways to incorporate solutions by learning high quality uncertainty estimates or providing interpretability of the models where needed for purpose of improving existing tools built in practice and allowing many other tools to be used where uncertainty is the key factor for decision making.

The first part of the thesis proposes approaches for learning high quality uncertainty estimates for both short- and long-term predictions in multi-task learning, developed on top for continuous probabilistic graphical models. In many scenarios, especially in long–
term predictions, it may be of great importance for the models to provide a reliability flag in order to be accepted by domain experts. To this end we explored a widely applied structured regression model with a goal of providing meaningful uncertainty estimations on various predictive tasks. Our particular interest is in modeling uncertainty propagation while predicting far in the future. To address this important problem, our approach centers around providing an uncertainty estimate by modeling input features as random variables. This allows modeling uncertainty from noisy inputs. In cases when model iteratively produces errors it should propagate uncertainty over the predictive horizon, which may provide invaluable information for decision making based on predictions.

In the second part of the thesis we propose novel neural embedding models for learning low-dimensional embeddings of medical concepts, such are diseases and genes, and show how they can be interpreted to allow accessing their quality, and show how can they be used to solve many problems in medical and healthcare research. We use EHR data to discover novel relationships between diseases by studying their comorbidities (i.e., co-occurrences in patients). We trained our models on a large-scale EHR database comprising more than 35 million inpatient cases. To confirm value and potential of the proposed approach we evaluate its effectiveness on a held-out set. Furthermore, for select diseases we provide a candidate gene list for which disease-gene associations were not studied previously, allowing biomedical researchers to better focus their often very costly lab studies. We furthermore examine how disease heterogeneity can affect the quality of learned embeddings and propose an approach for learning types of such heterogeneous diseases. Finally, we evaluate the quality of low-dimensional embeddings on tasks of predicting hospital quality indicators such as length of stay, total charges and mortality likelihood, demonstrating their superiority over other approaches.

In the third part of the thesis we focus on decision making in medicine and healthcare domain by developing state-of-the-art deep learning models capable of outperforming human performance while maintaining good interpretability and uncertainty estimates.
To Jelena.

To my parents Slobodan and Dragica.

To my late uncle Dragan.

To Gligorijević, Paunović and Stojanović families.

To all of my friends.

Thank you for your constant support and love throughout the writing of this thesis and for always being there for me.
ACKNOWLEDGEMENTS

I would like to express a sincere gratitude and appreciation to my advisor prof. Zoran Obradovic for his continuous support during my PhD studies and research, for his patience, motivation, enthusiasm, and immense knowledge. During the years of research and writing this thesis his guidance was a invaluable source of ideas and motivation.

Special thanks to Dr. Slobodan Vucetic, Dr. Kai Zhang and Dr. Rob J. Kulathinal, great professors and researchers from whom I learned a lot, for serving on my dissertation committee and for providing me with useful comments and valuable suggestions.

I would especially like to thank Jelena Gligorijevic my favorite teammate and co-author on most of my papers for her enthusiasm, pertinacity, impeccable working habits and discussions that have been a significant factor in our achievements and papers.

I would like to thank a lot to my other co-authors Milos Jovanovic, Vladan Radosavljevic, Nemanja Djuric, Mihajlo Grbovic, Rob J. Kulathinal, Ivan Stojkovic, Wayne Satz, Kraftin Schreyer, Daniel del Portal, Xiao Bai, Aravindan Ranghuveer, Amit Goyal, Martin Pavlovski, Elizabeth Milkovits, Lucas Glass, Kevin Grier, Praveen Vankireddy for their valuable ideas, discussions and contributions to our joint papers.

I thank professors and staff at the Department of Computer & Information Sciences for making such a friendly and pleasant working environment. I would especially like to thank Julie Srocki, Hailey King and professor Justin Y. Shi, for all the help and for always being there for us graduate students.
Special thanks to professors Richard Beigel, Yuhong Guo, Edward Crotty and Andrew Rosen, for whom I was working as a teaching assistant. Their experience and enthusiasm were of great inspiration throughout my years of teaching. I am also grateful to all my students at the Department of Computer and Information Sciences.

I thank my fellow labmates, postdocs and visiting scholars at the Center for Data Analytics and Biomedical Informatics: Alexey, Ana K., Ana S., Branimir, Branko, Chao, Dusan, Fang, Ivan, Jelena G., Jelena S., Jesse, Jumanah, Kosta, Marija, Martin, Milan B., Milan V., Milos J., Milos J., Milos R., Mladen, Mohamed, Nancy, Nima, Noor, Nouf, Shoumik, Tamiris, Thom, Tijana, Vladan, Vladisav, Xi for sharing their knowledge and perspectives during our lab meetings.

I would like to acknowledge the support of IQVIA and the DARPA GRAPHS program during my studies.

I would also like to thank my friends, colleagues and mentors at Yahoo and Oath for the summer internship opportunities in their groups and leading me working on diverse exciting projects. Those were invaluable and unforgettable experiences.

Finally, I would like to thank all my family and friends back home in Serbia, here in Philadelphia and across the United States of America for being with me throughout all these years.
TABLE OF CONTENTS

ABSTRACT ii
ACKNOWLEDGEMENTS v
LIST OF TABLES x
LIST OF FIGURES xii

CHAPTER

1 INTRODUCTION 1

2 MODELING UNCERTAINTY IN CONTINUOUS PROBABILISTIC GRAPHICAL MODELS 7

2.1 Introduction 7

2.2 Data description 8

2.2.1 Electronic Health Records Data 8

2.2.2 Disease graphs 9

2.2.3 Climate precipitation data: 11

2.3 Probabilistic graphical models for regression: Continuous Conditional Random Fields (CCRF) 12

2.4 The Gaussian Conditional Random Fields (GCRF) 13

2.5 The GCRF bias problem and uncertainty quality 16

2.6 A solution to GCRF bias and single step ahead prediction 17

2.6.1 Baseline Models 18
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>RMSE results of admission rate prediction. First column are averaged results over 12 months of prediction period and the following columns are prediction results for each month in the year 2011. The best results are bolded, second best are underlined and third best are in italic.</td>
<td>28</td>
</tr>
<tr>
<td>2.2</td>
<td>NLPD results of admission rate prediction. First column are averaged results over 12 months of prediction period and the following columns are prediction results for each month in the year 2011. The best results are bolded, second best are underlined and third best are in italic.</td>
<td>28</td>
</tr>
<tr>
<td>3.1</td>
<td>Precision and gene overlap for various competing models for the task of phenotype discovery, evaluated using disease-gene associations.</td>
<td>69</td>
</tr>
<tr>
<td>3.2</td>
<td>The four nearest disease neighbors for Chronic Kidney Disease Stage I.</td>
<td>71</td>
</tr>
<tr>
<td>3.3</td>
<td>Ten nearest disease neighbors of the Multiple Sclerosis phenotype retrieved by the disease2vec.</td>
<td>72</td>
</tr>
<tr>
<td>3.4</td>
<td>Ten nearest disease neighbors for the Sepsis and Congestive heart failure phenotypes retrieved by the disease2vec.</td>
<td>72</td>
</tr>
<tr>
<td>3.5</td>
<td>precision@K results for gene discovery task.</td>
<td>74</td>
</tr>
<tr>
<td>3.6</td>
<td>ICD-9 codes related to septic inpatients.</td>
<td>80</td>
</tr>
<tr>
<td>3.7</td>
<td>Accuracy and F-1 measure aggregated over 90 experiments for Logistic Regression model used on features learned by 6 embedding models: 4 type-specific and 2 global, for three values of hyperparameter b. The best results are bolded.</td>
<td>88</td>
</tr>
<tr>
<td>3.8</td>
<td>Sepsis (995.91 code) vector and its 5 nearest neighbors in the embedded global disease space vs 4 type-specific embedded disease space.</td>
<td>91</td>
</tr>
<tr>
<td>3.9</td>
<td>Five segments of Sepsis (995.91 code) and their 5 nearest neighbors in the embedded disease space.</td>
<td>92</td>
</tr>
</tbody>
</table>
3.10 Four segments of **Severe Sepsis** (995.92 code) and their 5 nearest neighbors in the embedded disease space .......................... 94

3.11 Four segments of **Septic Shock** (785.52 code) and their 5 nearest neighbors in the embedded disease space .......................... 95

3.12 Four segments of **Septicemia** (038.9 code) and their 5 nearest neighbors in the embedded disease space .......................... 96

3.13 Number of inpatient stays and number of diagnoses and procedure codes used for different healthcare providers .......................... 103

3.14 Association of procedures to two high-mortality diseases discovered by measuring cosine distance on features obtained using dp2v embedding model105

3.15 Overview of linear models used in this study .......................... 107

3.16 Average total charges, length of stay in days, and survival rate for four datasets from SID California database .......................... 111

3.17 $R^2$ results obtained for predicting total charges by five regression models for four insurance categories .......................... 113

3.18 $R^2$ results obtained for predicting LoS by five regression models for four insurance categories .......................... 114

3.19 Mortality prediction accuracy by five classification models for four insurance categories .......................... 115

4.1 Comparison of models that utilize only structured data, against DAM models trained on only unstructured, and on both types of input data. .......................... 132

4.2 DAM vs baselines performances for the binary classification task (more than 2 resources, or less). .......................... 134

4.3 DAM vs baselines performances for the multi-class classification task (number of resource category). .......................... 135

4.4 Approximating number of resources category by first acuity level prediction 135

4.5 Nurses’ performance approximation for the task of **Number of Resources Category** prediction via their **First Acuity Level** assessment .......................... 136

4.6 DAM performance for the task of **Number of Resources Category** prediction (grouped by the First Acuity Level approximation groups) .......................... 136
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Comorbidity graph variogram. Blue line represents disease comorbidity similarity and red line represents overall variance.</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>JS divergence graph variogram. Blue line represents disease similarity according to JS divergence and red line represents overall variance.</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>Similarity history variogram. Blue line represents disease history similarity and red line is overall variance.</td>
<td>11</td>
</tr>
<tr>
<td>2.4</td>
<td>Confidence interval plots for the three GCRF methods. Blue lines represent true values of admission rate for Hepatitis disease (24 months for years 2010 and 2011), while red lines represent prediction of the respective GCRF.</td>
<td>25</td>
</tr>
<tr>
<td>2.5</td>
<td>Min and obtained NLPD (y-axis; smaller is better) for all 231 diseases (x–axis) for three GCRF models with Linear Regression as unstructured predictor. Red bars represent the average minimal NLPD value for given prediction of the node over 12 months and blue bars are the average obtained NLPD value for that node over 12 months.</td>
<td>27</td>
</tr>
<tr>
<td>2.6</td>
<td>Iterative multiple-steps-ahead prediction in a temporal network represented by a vector of input attributes ($X_t$) and target variables $y_t$ for each time step $t$. Initial $L$ time steps (green boxes) are observed and remaining steps (blue boxes) are predicted iteratively in a one time step ahead process (red box).</td>
<td>30</td>
</tr>
<tr>
<td>2.7</td>
<td>MSE of one (blue) and multiple (red) 48 months ahead predictions of admission rate (a) and mortality (b) for all 260 disease categories obtained by 3 methods.</td>
<td>43</td>
</tr>
<tr>
<td>2.8</td>
<td>Predictions (red lines) and uncertainty estimates (gray area) of GCRF and IGP models for Septicemia disease admission rate (orange line) for up to 48 months (4 years) ahead</td>
<td>44</td>
</tr>
</tbody>
</table>
2.9 [y-axis]: **Mortality rate** predictions (red lines) and uncertainty estimates (gray area) by the extended GCRF for six of the most frequent diseases causing death in California for up to 48 months (4 years) ahead [x-axis]. True mortality is shown as orange lines.

2.10 MSE of one (blue) and multiple (red) 96 months ahead predictions of **precipitation** on all stations by 3 methods.

2.11 [y-axis]: Precipitation predictions (red lines) and uncertainty estimates (gray area) by GCRF and IGP iterative models for up to 96 months (8 years) ahead [x-axis]. True precipitation is shown as orange lines.

2.12 Predictions (red lines) and uncertainty estimates (gray area) of GCRF for **mortality rate** of top twelve killing diseases in California (orange lines) for 48 months (4 years) ahead (x-axis).

2.13 MSE of one (blue) and multiple (red) steps ahead predictions of **precipitation** on all stations using **iterative methods** with included input variables.

2.14 Predictions (red lines) and uncertainty estimates (gray area) of GCRF and IGP **iterative models** with included inputs for **precipitation** (true values–orange line) for 96 months (8 years) ahead.

2.15 MSE of one (blue) and multiple (red) steps ahead predictions of **precipitation** on all stations using **direct methods**.

2.16 Predictions (red lines) and uncertainty estimates (gray area) of DGCDF and DLR **direct models** of **precipitation** (true values–orange line) for 96 months (8 years) ahead.

3.1 Histogram of diagnoses counts for 35 million hospitalizations in California (on average 6.78 diagnoses were given per patient hospitalization).

3.2 Histogram of procedures counts for 35 million hospitalizations in California (on average 1.61 procedures were administered per patient hospitalization).

3.3 Graphical summary of the approach proposed in this study. Heterogeneous data obtained from large scale discharge records and hand curated disease-gene associations is used to jointly learn meaningful vector representations of disease and gene concepts in a latent vector space, where interactions of diseases and genes are retrieved and discovered.

3.4 Graphical representations of the **disease2vec** and **disease&gene2vec** models illustrated on projecting Acute Myocardial Infarction (AMI) diagnoses and AMI-related genes to AMI-associated diagnoses.
3.5 Precision@$K$ for disease2vec model with different dimension $D$ of the embedding space ................................................................. 67

3.6 Admission (blue) and mortality (red) trends of sepsis diagnoses in California for the period 2003-2009 ...................................................... 79

3.7 Prevalence of sepsis related diagnoses in SID California database ........ 81

3.8 Graphical representations of disease2vec models ................................. 82

3.9 Graphical representations of the disease2vec MSSG model .................. 85

3.10 In the embedded space (here displayed 2D reduced space) each of seven sepsis-related ICD-9 diagnoses is partitioned to five types marked in different colors. ................................................................. 89

3.11 Fraction of disease types 1-4 from Figure 3.10 in four sepsis-related ICD-9 diagnoses code groups (995.91, 995.21, 038.9 and 785.52) ............... 90

3.12 Pipeline of the proposed approach: 1) Use the proposed embedding methodology to learn vector representation of diseases and procedure $v_d, v_d \in \mathbb{R}^M$ using raw EHR data $\in \mathbb{R}^{[D]+|P|};$ 2) generate inpatient representation $X$ from the learned embeddings $v_d$ and $v_d;$ 3) train regression and classification models to predict important indicators of healthcare quality $y$ (LoS, TOTCHG and mortality for certain procedures and medical conditions of an inpatient). ................................................................. 101

3.13 Graphical representations of the disease+procedure2vec model. The model uses central disease/procedure $h_i$ to predict $b$ diseases/procedures (colored yellow and blue, respectively) that come before and $b$ that come after it in the discharge record. ................................................................. 102

3.14 $R^2$ results obtained by varying vector dimension ($M$) and window size ($b$) for SkipGram (sg) and CBOW (cbow) models for the task of predicting total charges. ................................................................. 104

3.15 Distribution of California inpatient hospital admissions by the primary payer (for a 2003-2011 period) ................................................................. 105

4.1 Emergency Severity Index (ESI): A Triage Tool for Emergency Department 120

4.2 ESI Level 3 patients prevalence ............................................................. 122

4.3 Deep Attention Model Architecture ........................................................ 127

4.4 Examples of patient complaints during triage process with attentions. .... 138
Obtaining meaningful predictions and insight from data brings a number of challenges and opportunities. Obtaining accurate predictions is very important, however, going beyond the predictive modeling is imperative for a range of reasons. These reasons span from unlocking future research efforts and debugging the performance of existing models, over providing unified approach to many practical problems such as anomaly detection, active and reinforcement learning, to allowing for a proper decision making. In practice, considering such facets of predictive modeling are imperative for the adequate usage of automated intelligent systems, especially where errors can have high costs.

Facets of predictive modeling that go beyond the prediction include model predictive uncertainty and interpretability. Historically, a trade-off was made between these facets and model discriminativeness (Bishop and Qazaz, 1996). More powerful and discriminative models with no notion of uncertainty and interpretability were used in fields like computer vision (CV), natural language processing (NLP) or even advertising where errors are not expected to yield serious consequences. On the other hand, simpler and less discriminative models, that are interpretable and have notion of uncertainty, are used in more sensitive fields such as healthcare informatics. The main reason for such tradeoff in
the modeling exists since developing models with good uncertainty estimates and interpretability can be a tedious task. Development of such models often results in non-elegant solutions that require non-trivial changes to the training algorithm thus hampering their implementation in real world systems, examples of which can be numerous (Bishop and Qazaz, 1996; Blundell et al., 2015). Another important challenge is that quantifying and evaluating these facets has not been well established by the community, thus increasing difficulty of developing models with such desired properties (Quinonero-Candela et al., 2006).

This thesis will address several of these apparent constraints by applying new solutions to a range of models from probabilistic graphical models to very deep neural models. All models will be examined on a variety of problems in the wide field of healthcare informatics where the uncertainty estimates and the interpretability are imperative for successful deployment of the model and its usage in real world systems.

It is important to first provide definitions of the model predictive uncertainty and interpretability:

---

Definition of model predictive uncertainty (Coccia and Todini, 2011): Predictive uncertainty is defined as the probability of occurrence of a predictand value conditional upon prior observations and knowledge, as well as on all the information we can obtain on that specific future value from model forecasts.

Uncertainty in modeling arises due to partial observability, nondeterminism or the combination of the two. Take the following as an example:

Sometimes pneumonia is hard to diagnose because it may cause symptoms commonly seen in simple colds or they can be a part of systemic diseases like flu. Further, symptoms seen in pneumonia are found in mild bronchitis, which is less severe disease and requires different treatment. Patients may not realize it’s more serious disease, until it lasts longer than these less severe conditions. Doctors diagnose pneumonia based on patient’s medical
history, physical exam, imaging study and lab results. There are few different types of pneumonia, based on the way in which a patient gets the infection. These different types can cause difficulties to physicians to diagnose them, since they need to have in mind all these types of pneumonia. To find out whether patients have bacterial, viral, or fungal pneumonia, doctors will ask questions, but sometimes patients give answers that could be misleading. Also, patients can have different level of disease and express symptoms from atypical, mild, all the way to severe pneumonia. Next, findings on imaging studies can be atypical for pneumonia, or lab tests could be missing and incomplete. Finally, medical science does not have a complete theory for the domain, thus even after all necessary observations the physician can still have hard time at making the diagnosis.

Having abovementioned in mind, we can say that it is reasonable to assume that decisions made by physicians in diagnosis of pneumonia come with a certain level of uncertainty (say physician is 90% confident that the patient has pneumonia). Furthermore, different treatments can have variable success on different patients, thus introducing uncertainty further into the treatment. The **rational decision** must then be made for the treatment of a patient, and the only way to achieve that is to weight relative importance of the goal (to cure the patient or not) and likelihood that the goal will be achieved.

The obvious solution for the problem above would be to devise exhaustive logical rules based on perfect observations, however as noted in the literature that is often infeasible due to three main reasons (Russell et al., 1995):

- **Laziness:** An exceptionless set of rules might be too large to make and it can be tedious to use.

- **Theoretical ignorance:** As stated above, medical science does not have a complete theory as is the case with many other fields.

- **Practical ignorance:** Even after all rules are known, uncertainty can arise due to missing information, such as a lab test that takes more time than allowed for the
decision to be made.

Due to these reasons intelligent systems can provide only a **degree of belief** for the relevant cases. The tool designed for working with degrees of belief that alleviates problems of laziness and ignorance is *probability theory*, as it allows for an elegant formalism of the abovementioned concepts.

As for this thesis, we will mainly focus on the predictive modeling (as a subset of intelligent systems), thus it is important to formalize uncertainty in relevant terms. There are three types of uncertainty often encountered (notably in Bayesian modeling): *model uncertainty*, *inherent noise* and *model misspecification*.

**Model uncertainty**, or also often called **epistemic uncertainty**, captures ignorance about the most suitable model for the data or ignorance in model’s parameters. Epistemic uncertainty decreases with increase of the observed data (thus its alternative name “reducible uncertainty”). Capturing this uncertainty requires generating multiple realizations of the model and estimating variance on the fixed sample of data.

**Inherent uncertainty** (often called **aleatoric uncertainty**) captures noise in data generation process and is irreducible. This includes noise from the environment, measurement, missingness or random noise, etc. This noise is often modeled with the likelihood by assuming some noise process on top of functions output. A very popular noise process used in regression is assuming Gaussian noise, however other processes are used as well, such as Laplace noise.

In term of formalizing the previous two types of uncertainty we can think of a learned model $f^W(x)$, with $\hat{W}$ being optimal parameters. Given a new test data point $x^*$ we produce predictions as: $y^* = f^\hat{W}(x^*)$. To cope with aleatoric uncertainty we can assume Gaussian noise on parameters $W$ as $W \sim \mathcal{N}(0, \sigma^2 I)$ obtaining probability predictor:

$$p(y|W) \sim \mathcal{N}(f^W(x), \sigma^2). \quad \text{(1.1)}$$

To obtain the predictive distribution we can factor out the noise from the data (which
is frequently modeled over noisy parameters $W$) by marginalizing it as:

$$p(y^*|x^*) = \int_W p(y^*|f^W(x^*))p(W|X,Y),$$

where $X,Y$ represent training data. Variance of the predictive distribution captures uncertainties, and we can estimate it using the Law of total variance as:

$$\text{Var}(y^*|x^*) = \text{Var}(E(y^*|W,x^*)) + E[\text{Var}(y^*|W,x^*)] = \text{Var}(f^W(x)) + \sigma^2. \tag{1.3}$$

The first part of the Eq. 1.3 represents epistemic uncertainty and second part is aleatoric uncertainty. It should be noted that above formalization is for the general case of $f^W(x)$, however for convex models (whose variance of parameters equals zero) we may alter it as shown in Section 2.7.

Finally, **model misspecification** is seldom declared as a type of uncertainty separately from the epistemic uncertainty, and not as an extreme case of it. In this thesis we recognize it as an individual type of uncertainty. It includes cases where testing samples come from a different distribution than the training samples.

**Definition of interpretability:** To interpret means to explain or to present in understandable terms (Doshi-Velez and Kim, 2017). In the context of the machine learning systems, we will define interpretability as the ability to explain or to present model’s results in terms understandable to a human that can furthermore be quantified.

In this thesis we will, in addition to the predictive uncertainty, focus on models that are interpretable either as having interpretable representations of interpretable predictions.

The organization of the thesis will be outlined as such. In Chapter 2 we focus on a powerful and interpretable probabilistic graphical model where we address issue of resolving model bias and providing uncertainty estimates of highest quality. Chapter 3 focuses on learning interpretable representations of medical concepts, such are diseases, genes and procedures from electronic health records (EHRs) and evaluation their usefulness in
several healthcare related problems. Finally, Chapter 4 addresses the interpretability of powerful deep neural models notorious for the lack of notion of such modeling facet.
CHAPTER 2

MODELING UNCERTAINTY IN CONTINUOUS PROBABILISTIC GRAPHICAL MODELS

2.1 Introduction

Conditional probabilistic graphical models provide a powerful framework for structured regression in spatio-temporal datasets with complex correlation patterns. It has been shown that models utilizing underlying correlation patterns (structured models) can significantly improve predictive accuracy as compared to models not utilizing such information (Radosavljevic et al., 2010, 2014; Ristrovski et al., 2013; Wytock and Kolter, 2013; Stojanovic et al., 2015).

These models have been successfully applied in many real-world problems, providing improvements over given state-of-the-art models while maintaining reasonable scalability. Some of the applications are in the climate domain (Djuric et al., 2015a; Radosavljevic et al., 2010; Stojanovic et al., 2015; Ristrovski et al., 2013; Pavlovski et al., 2018), traffic estimation (Djuric et al., 2011), power systems (Dokic et al., 2016), healthcare (Radosavljevic et al., 2013; Gligorijevic et al., 2015a, 2016c; Polychronopoulou and Obradovic, 2014), biomedicine (Stojkovic et al., 2016a), image de-noising problems (Ristrovski et al.,...
Next, we discuss datasets used in studies covered in this chapter.

2.2 Data description

2.2.1 Electronic Health Records Data

The primary data source used in this thesis is the State Inpatient Databases (SID) which is an archive that stores the universe of inpatient discharge abstracts from data organizations. It is provided by the Agency for Healthcare Research and Quality and is included in the Healthcare Cost and Utilization Project (HCUP) (1, 2009). Particularly, we have access to the SID California database, which contains 35,844,800 inpatient discharge records over 9 years (from January 2003 to December 2011) for 19,319,350 distinct patients in 474 hospitals (436 AHAID identified; about 400 per year). For each patient there are up to 25 diagnosis codes in both CSS and ICD9 coding schemas, together with up to 25 procedures applied during this admission of the patient. There is also some demographic information about each patient, like age, birth year, sex, race, etc., as well as information about hospital stays, length of stay, total charges, type of payment, a payer, discharge month, survival information.

Availability of this information made possible building of several graphs, and exploring usefulness of attributes and links of the graph for a chosen target attribute. In this chapter, we focused on disease based graphs and prediction of number of admissions for each disease. Graphs are constructed for these 9 years in monthly resolution. In our experiments, we used 231 out of 259 diseases that we were able to follow throughout all 9 years. For each node, we include temporal information, meaning that there are one, two or three previous values of the target variable (we refer to these attributes as lag1, lag2, and lag3) as attributes of a node in the current time step (more details about utilization of those attributes are given in Section 2.6.3). When it comes to the structure, we explored several
cases.

2.2.2 Disease graphs

We present several examples of graphs that can be built from the EHR records found in HCUP California database and discuss their properties.

Comorbidity graph

Disease-based graphs can be built as comorbidity graphs (graphs of disease co-occurrences within patient’s visit) based on phenotypic patient data (Davis and Chawla, 2011). However, this type of graph did not provide satisfactory results, since variogram (Uversky et al., 2014) for the selected target attributes was not appropriate (Figure 2.1). Variograms were inspected for two behaviors: displaying a decreasing trend in variance at higher values of similarity, and falling below the line of the overall variance of the data. By (Uversky et al., 2013) this similarity measure would be characterized as a “bad” similarity measure. Therefore, we researched several other disease links, like those based on common patient profile or common history.

![Figure 2.1: Comorbidity graph variogram. Blue line represents disease comorbidity similarity and red line represents overall variance.](image)
**Jenson-Shannon divergence graph**

As a first case, we measured Janson-Shanon divergence between distribution of two diseases based on their attributes.

\[
JSD(P||Q) = \frac{1}{2}(KLD(P||M) + KLD(Q||M)),
\]

(2.1)

where \(KLD\) is the Kullback-Leibler divergence, \(P\) and \(Q\) are the distributions of the selected disease attribute for two observed diseases and \(M = \frac{1}{2}(P + Q)\). The similarity obtained from this divergence is

\[
S(y_p, y_q) = \frac{1}{JSD(x_p||x_q)}.
\]

(2.2)

Utilizing the variogram technique showed that using the distribution of white people admitted in California in period 2003.–2011. for each disease showed the best performance among all other attributes of each disease (Figure 2.2).

**Figure 2.2:** JS divergence graph variogram. Blue line represents disease similarity according to JS divergence and red line represents overall variance.

**Common history graph**

The best performance so far was obtained by the structure built on common history links as can be seen in Figure 2.3, calculated using formula:

\[
S(y_i, y_j) = \exp(-\text{mean}(\text{abs}(x^h_i - x^h_j))).
\]

(2.3)
Here, \( x^h_i \) represents vector of length \( h \) utilized for given attribute of the node \( x \). For example, if we were observing previous 2 time steps the similarity would be

\[
S(y_i, y_j) = \exp\left(-\frac{\text{abs}\left((x_{t1}^i - x_{t1}^j) + (x_{t2}^i - x_{t2}^j)\right)}{2}\right). \tag{2.4}
\]

The variogram was obtained using historical information of admitted whites for each observed disease in previous three timesteps. As this variogram is the best, we have decided to use this measure in our experiments.

![Figure 2.3: Similarity history variogram. Blue line represents disease history similarity and red line is overall variance.](image)

To demonstrate the generalizability of the proposed approaches, we have conducted experiments with different datasets as well.

2.2.3 Climate precipitation data:

A dataset of precipitation records from meteorological stations across the USA has been acquired from NOAA’s National Climate Data Center (NCDC) (Menne et al., 2009). The precipitation data includes 1218 meteorological stations. Most of these stations are U.S. Cooperative Observing Network stations generally located in rural locations, while some are National Weather Service First-Order stations that are often located in more urbanized environments. A temporal graph is constructed such that nodes at each time slice represent 1218 stations. The spatial information is used for calculating similarities (correlations) be-
tween stations, but the graph is constructed such that only stations within certain diameter are connected, thus the graphs structure is sparse.

In addition to precipitation, there are 6 more variables at each node which we use as input attributes for each station. These variables are acquired from the NCEP/NCAR Re-analysis 1 project (Kalnay et al., 1996), which is using a state-of-the-art analysis/forecast system to predict climate parameters using past data from 1948 to the present (data available on NOAA website: http://www.esrl.noaa.gov/psd/). These 6 variables are omega (Lagrangian tendency of air pressure), precipitable water, relative humidity, temperature, u-wind, and w-wind (zonal and meridional components of the wind, respectively). Our goal is to make use of these variables and try to exploit inter-dependencies between stations in order to improve the prediction of precipitation amounts in these stations. Since these attributes are obtained on the lower resolution than individual stations we used the values of attributes from the nearest neighbor. To improve predictions, we perform square root transformation of the target variable and did cross-validation during the training to learn the hyper-parameters of the models.

2.3 Probabilistic graphical models for regression: Continuous Conditional Random Fields (CCRF)

Structured methods have emerged in the previous decade, and there are many evidences of benefits of using structured methods over unstructured (Radosavljevic et al., 2010). As many datasets can be presented as graphs these methods have gained popularity and are considered state-of-the art methods in many domains.

Continuous Conditional Random Fields were proposed by (Qin et al., 2009), and model conditional distribution:

\[
P(y|X) = \frac{1}{Z(X, \alpha, \beta)} \exp(\phi(y, X, \alpha, \beta)).
\]

(2.5)

The term in the exponent \(\phi(y, X, \alpha, \beta)\) and the normalization constant \(Z(X, \alpha, \beta)\) are
defined as follows

\[ \phi(y, X, \alpha, \beta) = \sum_{i=1}^{N} A(\alpha, y_i, X) + \sum_{i \sim j} I(\beta, y_i, y_j, x), \] (2.6)

\[ Z(X, \alpha, \beta) = \int_{y} \exp(\phi(y, X, \alpha, \beta)) dy, \] (2.7)

Function \( A \) is called the association function, and it represents any function that handles mapping from \( X \rightarrow y \) with respect to input variables \( X \). Function \( I \) is called interaction potential and it handles any relation the two data instances \( y_i \) and \( y_j \) have. To efficiently define the CRF form, association and interaction potentials are defined as linear combinations of feature functions \( f \) and \( g \) (Lafferty et al., 2001),

\[ A(\alpha, y_i, X) = \sum_{k=1}^{K} \alpha_k f_k(y_i, X), \] (2.8)

\[ I(\beta, y_i, y_j, x) = \sum_{l=1}^{L} \beta_l g_l(y_i, y_j, X). \] (2.9)

To make an efficient CRF model, the choice of feature functions \( f \) and \( g \) is crucial.

2.4 The Gaussian Conditional Random Fields (GCRF)

Gaussian Conditional Random Fields (GCRF) (Radosavljevic et al., 2010) is a discriminative structured regression model. The model captures both the network structure of variables of interest (\( y \)) and relationship between attribute values of the nodes (\( X \)) and the target variable \( y \). It is a model over a general graph structure (not only chains or trees), and can represent the relationships of the nodes as a function of time, space, or any user-defined structure. It models the structured regression problem by estimating a joint continuous distribution over all nodes. GCRF takes the following log-linear form,
modeling feature functions \( f \) and \( g \) as quadratic functions:

\[
P(y|X) = \frac{1}{Z(x, \alpha, \beta)} \exp(\phi(y, X, \alpha, \beta)) =
\]

\[
\frac{1}{Z(x, \alpha, \beta)} \exp(\sum_{i=1}^{N} A(\alpha, y_i, X) + \sum_{i\sim j} I(\beta, y_i, y_j, X)) =
\]

\[
\frac{1}{Z} \exp(-\sum_{i=1}^{N} \sum_{k=1}^{K} \alpha_k(y_i - R_k(X))^2 - \sum_{i\sim j} \sum_{l=1}^{L} \beta_l S_{ij}^{(l)}(y_i - y_j)^2) \quad (2.10)
\]

The first part of the log-linear form \( A(\alpha, y_i, X) = -\sum_{k=1}^{K} \alpha_k(y_i - R_k(X))^2 \) is called association potential and it aims to model associations \( X \rightarrow y_i \) using \( K \) different functions \( R_k(X) \), which we will call unstructured predictors, as they are modeling these associations independently by learning from data or by using domain knowledge. Parameters of the association potential \( \alpha_k \) are learned as degrees of belief towards each unstructured regressors. Given by the squared error \( \sum_{i=1}^{N} (y_i - R(X))^2 \), larger belief \( \alpha \) is learned to correspond to the more accurate unstructured predictor.

The second part \( I(\beta, y_i, y_j, X) = -\sum_{l=1}^{L} \beta_l S_{ij}^{(l)}(y_i - y_j)^2 \) is called interaction potential and its goal is to utilize a graph structure \( S \), that should be a weighted undirected network whose edges \( S_{ij} \) denote how similar two nodes are, or more precisely, how similar their response values \( y_i \) and \( y_j \) are. Parameters \( \beta \) are learned as degrees of belief towards similarity metrics and their values are governed by the product of similarity metric and squared distance \( \sum_{i\sim j} S_{ij}(y_i - y_j)^2 \). If this distance is small, relative value of \( \beta \) will be larger and the entire model will take the structure as an important source of information.

The normalization term

\[
Z(x, \alpha, \beta) = \int_y \exp(\phi(y, X, \alpha, \beta))dy \quad (2.11)
\]

and in general case, estimating this term is intractable. However, using quadratic feature functions, as demonstrated in Eq. 2.10, enables an elegant representation of the log-linear
form as a multivariate Gaussian distribution (Radosavljevic et al., 2010):

\[
P(y|X) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp \left( -\frac{1}{2} (y - \mu)^T \Sigma^{-1} (y - \mu) \right),
\]

which allows efficient convex optimization. Here, \(\Sigma^{-1}\) represents the diagonally dominant inverse covariance matrix, and for this model takes the form:

\[
\Sigma^{-1} = \begin{cases}
2 \sum_{k=1}^{K} \alpha_k + 2 \sum_{g} \sum_{l=1}^{L} \beta_l S_{ij}^{(l)} (x), & i = j \\
-2 \sum_{l=1}^{L} \beta_l S_{ij}^{(l)} (x), & i \neq j
\end{cases}
\]

(2.13)

The posterior mean is given by \(\mu = \Sigma b\), where \(b\) is defined as

\[
b_i = 2 \left( \sum_{k=1}^{K} \alpha_k R_k(X) \right).
\]

(2.14)

**Learning and inference**

The learning task is to optimize parameters \(\alpha\) and \(\beta\) by maximizing the conditional log-likelihood,

\[
(\hat{\alpha}, \hat{\beta}) = \underset{\alpha, \beta}{\arg \max} \log P(y|X; \alpha, \beta).
\]

(2.15)

Parameters \(\alpha\) and \(\beta\) are learned by a gradient-based optimization. Gradients of the conditional log-likelihood are:

\[
\begin{align*}
\frac{\partial \mathcal{L}}{\partial \alpha_k} &= -\frac{1}{2} (y - \mu)^T \frac{\partial \Sigma^{-1}}{\partial \alpha_k} (y - \mu) + \frac{\partial b^T}{\partial \alpha_k} (y - \mu) + \text{Tr}(\Sigma \frac{\partial \Sigma^{-1}}{\partial \alpha_k}) \\
\frac{\partial \mathcal{L}}{\partial \beta_l} &= -\frac{1}{2} (y + \mu)^T \frac{\partial \Sigma^{-1}}{\partial \beta_l} (y - \mu) + \text{Tr}(\Sigma \frac{\partial \Sigma^{-1}}{\partial \beta_l})
\end{align*}
\]

(2.16)

(2.17)

Maximizing the conditional log-likelihood is a convex objective, and can be optimized using standard Quasi-Newton optimization techniques. Constraint of positive-semi definiteness of matrix \(\Sigma^{-1}\) ensures that the distribution is Gaussian. Therefore, to make the
optimization unconstrained, the exponential transformation of parameters \( \alpha_k = e^{u_k} \) and \( \beta_l = e^{v_l} \) is used in GCRF (Radosavljevic et al., 2010).

Optimization of the GCRF has high asymptotic complexity of \( O(N^3) \) due to inverse of precision matrix (can be \( O(N^2) \) for sparse covariance matrices). While this thesis does not consider scaling the model to very large datasets, several references show that it is quite possible to do so in linear time \( O(N) \): (Ristovski et al., 2013; Stojkovic and Obradovic, 2017; Stojkovic et al., 2017).

In this model, prediction is governed by two parts, the association and interaction potentials. The association potential guides the main prediction power of the GCRF model and clearly, the more accurate the unstructured models are, the more GCRF will assimilate those predictions. On the other hand, as these unstructured predictors usually do not consider the structure information, interaction potential will compensate that by introducing similarity matrix \( S \), and bringing the predictions of the connected nodes closer together. A combination of the two potentials provides better accuracy than the unstructured predictors alone.

2.5 The GCRF bias problem and uncertainty quality

As both unstructured predictor \( R \) and similarity matrix \( S \) are given as learned prior to GCRF model learning, they introduce a bias in the model. There is a clear tradeoff between minimizing epistemic uncertainty via convex objective and obtaining good quality uncertainty estimate. Namely, as model does not have observation on measurements (only on unstructured predictor values and similarity matrix), it yields larger aleatoric uncertainty than needed, resulting in overblown uncertainty of the model when predicting. To alleviate this problem, we proposed several solutions that allow model to adaptively learn parameters tied to unstructured predictors and similarity matrices. This yields in lower model bias and much better overall uncertainty estimate. Details of the mentioned models are given
below in two different studies on single and multi-step-ahead prediction approaches.

2.6 A solution to GCRF bias and single step ahead prediction

Among many difficult tasks, capturing global trends of diseases are the ones that intrigue many healthcare practitioners (Jones et al., 2008; L. et al., 2014). Being able to confidently predict future trends of diseases, may lead to better anticipation of healthcare systems and allow better decision making, which should consequently provide higher quality service to those who need it.

As many diseases are related, graphical modeling of disease data may be beneficial for predicting disease trends. As such, several types of graphs may be built and several prediction tasks may be imposed on these graphs. In the literature, several networks have been constructed to study the connection between human diseases. The nodes of the networks are disease codes while the links are derived based on common genes (K.I. et al., 2007), shared metabolic pathways (D.S. et al., 2008), connection to common miRNA molecule (Lu et al., 2008), observed comorbidity (Hidalgo et al., 2009). These networks have been used for the study of obesity (Barabasi et al., 2011), illness progression (Hidalgo et al., 2009), association of diseases with the cellular interactions (Park et al., 2009), properties of the genetic origin of diseases (K.I. et al., 2007), etc.

Since the HCUP are EHR data (Section 2.2.2), we are more interested in modeling of phenotypic networks. These kinds of networks are introduced in (Davis and Chawla, 2011; Hidalgo et al., 2009). In (Davis and Chawla, 2011) a novel multi-relational link prediction method is proposed and it is shown that disease comorbidity can enhance the current knowledge of genetic association. In our study, we are primarily concentrated on disease-based graphs (we have opted modeling 253 diseases coded with CCS schema rather than modeling diseases with ICD-9 codes, because the CCS is the empirically built schema interpretable by wider audience rather than medical experts). Specifically, disease-
based graphs can be built as comorbidity graphs based on phenotypic patient data (as described in Section 2.2.2), but also other disease links based on common genes (Davis and Chawla, 2011), ontologies, common patient profile, or common history.

As mentioned before, in previous work on diseases data, many graphs were proposed, and we now aim to utilize such constructed graphs to improve predictive power of unstructured models, which to our knowledge was not done before. For this task we are using a Gaussian Conditional Random Fields (GCRF) model. The GCRF model (Radosavljevic et al., 2010) has been successfully applied in many real-world problems.

When models provide prediction for real-world problems, it can be very important to report uncertainty estimation of the prediction. This is especially true for domains where predictions are used for important decision-making, such as health. The objective of this paper is to improve the estimation quality of prediction uncertainty in the GCRF model for evolving graphs in order to more accurately represent the confidence of the prediction for healthcare applications. For example, instead of predicting that the number of admissions for a disease is going to be $15.026 \pm 10.000$, making a different estimation of $15.000 \pm 150$, can be more useful for decision making process. Therefore, we aim to address this important topic in this paper. We experimentally demonstrate improvement of predictive uncertainty by removing bias of the GCRF framework via representing its parameters as functions. We propose two approaches in Section 2.6.2, one that models parameters as a function of an unstructured predictors’ uncertainty estimation and one that models GCRF parameters as neural networks (Radosavljevic et al., 2014).

### 2.6.1 Baseline Models

We aim to model complex trends of diseases in an autoregressive manner utilizing linear and non-linear models. These models will be called unstructured predictors as they have no information on graph structure, that might prove beneficial for predictive performance. Therefore, the structured GCRF framework explained in Section 2.4 will be used
for introducing graph structure to these models.

Unstructured Models

For modeling disease trends, we will be using several unstructured predictors: Linear regression and Gaussian Processes regression models with several lags for learning each. These methods are evaluated and the ones with best performance are chosen as inputs for the GCRF framework, which will then introduce graph structure information to the prediction, providing a certain level of correction. As uncertainty of the unstructured predictors is the additional information we are using in some of our experiments, we will provide formulas for retrieving uncertainty from unstructured predictors.

Linear auto-regressive model

Linear regression form of auto-regressive (AR) representation is:

$$y_i = w^T x_i + \varepsilon, \varepsilon \sim N(0, \sigma_y^2)$$  \hspace{1cm} (2.18)

where $w$ is an unknown set of weights. The weight and noise variance are estimated by

$$\hat{w} = (X^T X)^{-1} X^T y.$$  \hspace{1cm} (2.19)

The variance of the Linear predictor is given by

$$\sigma_y^2 = \frac{(y - \hat{w}^T X)^T(y - \hat{w}^T X)}{N - k - 1},$$  \hspace{1cm} (2.20)

where $X$ is matrix representation of all data available for training, $N$ is the number of training examples and $k$ is the number of attributes. Prediction $y_*$ and uncertainty estimation $\sigma_*^2$ of the test data $x_*$ are found using

$$y_* = \hat{w}^T x_*,$$  \hspace{1cm} (2.21)

$$\sigma_*^2 = \sigma_y^2(1 + x_*{X^T X}^{-1}x_*^T),$$  \hspace{1cm} (2.22)

as described in (Smith, 2013).
**Gaussian Processes Regression Model**

A Gaussian process (GP) is a generalization of a multivariate Gaussian distribution over finite vector space to a function space of infinite dimension (Rasmussen and Williams, 2006). Assumption of a Gaussian prior is present over functions that map $x$ to $y$:

$$y_i = f(x_i) + \varepsilon, \varepsilon \sim \mathcal{N}(0, \sigma^2_y).$$

(2.23)

Gaussian processes are defined with

$$f(x) \sim GP(m(x), k(x, x')),$$

(2.24)

where $m(x)$ is the mean function and $k(x, x')$ is the covariance function in the form of a kernel that is required to be positive definite. In our implementation we are using a Gaussian kernel:

$$k(x_i, x_j) = \sigma^2_y \exp \left( -\frac{1}{2} \sum_{d=1}^{D} \frac{(x_{i,d} - x_{j,d})^2}{w_d^2} \right).$$

(2.25)

Here, $\sigma^2_y$ is the white noise variance parameter of the Gaussian prior.

If we denote covariance of training part as $C = K + \sigma^2_y I_N$, $K_{ij} = k(x_i, x_j)$, the joint density of the observed outputs $y$ and test output $y_*$ is presented as

$$\begin{pmatrix} y \\ y_* \end{pmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} C & k_* \\ k_*^T & c_* \end{bmatrix} \right),$$

(2.26)

where $k_*$ is the covariance vector for the new test point $x_*$. The posterior predictive density is given by

$$p(y_* | x_*, X, y) = \mathcal{N}(y|0, C),$$

(2.27)

$$\mu_* = k_*^T C^{-1} y, \quad \sigma^2_* = c_* - k_*^T C^{-1} k_*.$$

(2.28)

The $\sigma^2_*$ is the predictive variance or uncertainty at test point $x_*$ (Rasmussen and Williams, 2006).
2.6.2 Parameters of GCRF as functions

The GCRF (Radosavljevic et al., 2010) intrinsically possesses uncertainty estimation. This uncertainty estimation is highly biased towards the data the model was trained on as GCRF does not depend on input variables directly (Gligorjevic et al., 2015a). Once parameters of the GCRF model are introduced as functions, these functions impact both parameters values and scale (note that $\sum_k \alpha_k + \sum_l \beta_l = I$, where $I$ is a unit vector). Parameters in the GCRF represent degree of belief toward each unstructured predictor and similarity measure. If they are modeled to be dependent on input variables, the bias problem will be solved and thus uncertainty estimation should be improved. The uncertainty estimation improvement is achieved both by better fitting parameters and by altering the unit scale to better fit the data as well. We experimentally evaluate our assumptions on modeling parameters of the GCRF model as functions in terms of uncertainty estimation and prediction accuracy (Section 2.7.7). We now summarize potential ways of handling the bias in the GCRF models primarily focused on parameters $\alpha_k$.

Parameters $\alpha_k$ as functions of unstructured predictor’s uncertainty

First, we address the problem of model bias by modeling the overall model uncertainty as a function of the uncertainty of each unstructured prediction. The principal assumption is that the chosen unstructured predictors can output uncertainty for their predictions. Initial GCRF uncertainty estimation improvements were done on modeling time series of patients’ response to acute inflammation treatment (Radosavljevic et al., 2013) showing that the uncertainty estimation of GCRF modeled in this way provides a higher quality of uncertainty than the quality of this measure for utilized unstructured predictors. The new parameters of the GCRF model are modeled such that:

$$\alpha_{k,p} = \frac{e^{u_{k,p}}}{\sigma_{k,1}^2}, \beta = e^v$$  \hspace{1cm} (2.29)
where $\sigma_{k,1}^2$ represents the uncertainty estimation of unstructured predictor k for the first time step ($p = 1$), while $\alpha$ and $\beta$ are coefficients for the feature functions.

We have extended this work by relaxing a few assumptions made in (Radosavljevic et al., 2013), as well as by applying uncertainty estimation on evolving graphs data (rather than applying it on linear-chain data). The first assumption of the previous work was that the log likelihood of GCRF would be optimized with respect to the uncertainty of each unstructured predictor, but only considering the first time step of the respective model. This follows the homoscedasticity assumption that the variance of the model will not change significantly through time. This strong assumption of homoscedasticity is dropped in our study and the parameters of the GCRF model have been optimized with respect to the uncertainty estimation of each unstructured predictor at each time step. This allows the model to optimize different parameters for different prediction horizons. Thus, if predictions of the unstructured predictor increase uncertainty further in the future, GCRF will also adjust accordingly.

We have further improved uncertainty estimation by penalizing the predictors whose uncertainty estimation was not good enough during validation. This could be done by any quality measure of uncertainty estimation on the training data. We use the percentage of nodes that fall into the 95% confidence interval ($c_{i_k,p}$) as a quality index to augment our approach:

$$\alpha_{k,p} = \frac{e_{u_{k,p}}}{\sigma_{k,p}^2} c_{i_k,p}. \quad (2.30)$$

**Parameters $\alpha_k$ as functions of input variables**

The previous approach can be further generalized by observing parameters $\alpha$ as functions of input parameters, which was proposed in (Radosavljevic et al., 2014). However, in (Radosavljevic et al., 2014) there were no experimental results on uncertainty improvement and the paper does not mention this aspect of the extension. We can observe the parameter $\alpha$ as a parameterized function of input variables $\alpha(\theta_k, x)$, where $\theta_k$ are the parameters, and
$x$ are input variables of function $\alpha$. Due to the positive semi-definiteness of the precision matrix constraint, function $\alpha(\theta_k, x)$ becomes

$$
\alpha_k = e^{u_k(\theta_k, x)},
$$

(2.31)

where $u_k(\theta_k, x)$ is in (Radosavljevic et al., 2014) a feed-forward neural network. This method is easily optimized using gradient descent methods. These functions are titled as uncertainty functions, since they provide significant improvement to the covariance matrix in terms of bias correction.

2.6.3 Experiments

We have characterized mentioned methodology approaches to the HCUP diseases evolving graph. Our goal is to predict the number of people admitted for a disease as primary one in twelve months of the year 2011 in the state of California, which is the last year in our database, one month at a time. We have normalized the admission count to a 0-1 scale, making admission rate as the target variable. Predicted values can easily be converted back to counts of people admitted for that disease. Between nodes in each timestep we have defined the similarity values based on the exploration in Section 2.2.2.

For the unstructured predictors we have used a linear and non-linear model. Both models were trained in an autoregressive fashion for each disease separately, observing several previous timesteps to infer one timestep ahead. For each of the two models we have used target variables of up to three timesteps in history (lags) as features of the models. Unstructured predictors were optimized with a sliding window of 12 months. Among the unstructured predictors, the best linear and non-linear predictors were chosen as inputs into the GCRF model. Separate GCRF models were optimized with linear and non-linear predictors so the effect of each on the results could be characterized separately.

We are training and comparing three different GCRF models. The first is an original GCRF model (described in Section 2.4), with a slight difference that an $\alpha$ parameter was
optimized for each node in a graph separately for fair comparison with the other GCRF models. We will denote this model simply as GCRF. The second GCRF model is the one described in Section 2.6.2, which includes the uncertainty estimation of the unstructured predictor to rescale parameter $\alpha$ with uncertainty of the unstructured predictor. In our results this method is called uGCRF. Finally, the third GCRF model is the one described in Section 2.6.2, where parameters $\alpha$ were modeled as feed-forward neural networks. The last GCRF model, with uncertainty functions, will be denoted as ufGCRF in our results.

We will characterize all methods for their predictive power using the root mean squared error (RMSE). As we have mentioned before, an important goal of mentioned approaches is the improved uncertainty estimation power, which will be evaluated using negative log predictive density as we describe below.

**Uncertainty estimation quality measure.** The quality of uncertainty estimates is evaluated by measuring the average Negative Log Predictive Density (NLPD), a measure that incorporates both the accuracy and uncertainty estimate of the model. NLPD metric penalizes both over and under confident predictions. This measure is also used in data analysis competitions (Quinonero-Candela et al., 2006) where uncertainty estimation was of major importance. Smaller values of NLPD correspond to better quality of the estimates. For given prediction $y_{i*}$, NLPD reaches a minimum for $\sigma^2_{i*} = (y_i - y_{i*})^2$.

\[
NLPD = \frac{1}{2} \sum_{i=1}^{N} \frac{(y_i - y_{i*})^2}{2\sigma^2_{i*}} + \log\sigma^2_{i*} \tag{2.32}
\]

where $\sigma^2_{i*}$ is predicted uncertainty.

**Experimental results.** In our experiments, unstructured predictors provide higher predictive accuracy when using only the previous timestep as input (lag 1). As such, we have used the Linear regression and Gaussian Processes regression with lag 1 as unstructured models for the structured GCRF model. Results are shown in Table 2.1, the best results are bolded, the runner-up results are underlined, and third best results are in italic. From
the Table 2.1 it can be observed that the GCRF model using GP as unstructured predictor
gives the most accurate result and that two other GCRF models introduce slightly more
error to the original GCRF model. The main objective of uncertainty functions is to scale
belief towards unstructured predictors with respect to change of their respective input vari-
ables. Rescaling variance values (from which uncertainty is expressed) is another, more
robust effect of uncertainty functions that significantly benefits the model’s uncertainty es-
timation, as described in Section 2.6.2. Following the original GCRF method results, the
ufGCRF model introduces less error than the uGCRF method deeming it superior among
the two extensions in this experiment. It should be noted that all GCRF methods displayed
outperform all of the unstructured predictors. This evidently comes from the structure in-
formation included in the prediction. As a reminder, we are using common history graph
described in Section 2.2.2.

The other aspect of characterizing the methods includes evaluation of uncertainty qual-
ity, which is an important concept in this study. Uncertainty estimation is evaluated using

![Graph showing confidence interval plots for the three GCRF methods.]
the NLPD metric and the results are shown in the Table 2.2. The original GCRF model provides lowest quality of the uncertainty estimation among all of the other methods. This can be attributed to the underconfident predictions of the GCRF model, which can be observed also in Figure 2.4. By underconfidence we refer to the estimated confidence interval being too high (large gray area in Figure 2.4 in the case of the original GCRF). For instance, in Figure 2.4 we observe that GCRF is giving estimate for admission of Hepatitis of about $0.61 \pm 0.78$, making predictions of the GCRF model less useful. The two extensions narrow down the unnecessary large confidence interval in the original GCRF case, where their predictions are $0.81 \pm 0.16$ for uGCRF and $0.71 \pm 0.25$ for ufGCRF, which can be more helpful for decision making process. Since NLPD measure considers both accuracy of the model and uncertainty estimation quality, we see that this ability to narrow confidence interval of the two extensions, is also reflected in the results shown in Table 2.2. The fact that ufGCRF has better predictive power gives it better results in NLPD than uGCRF, which shows good uncertainty estimation, but introduce little bit more error than ufGCRF. Actually, the best performance in terms of NLPD measure gives ufGCRF using LR as unstructured predictor, but right after it goes again ufGCRF with GP, so we can conclude that this GCRF extension gives the best uncertainty estimation in our experiments among all models. The comparison of uncertainty estimation quality of the three GCRF methods is also given in Figure 2.5 (this figure shows result using LR as unstructured predictor, however results for using GP are similar). Each bar represents mean NLPD measure for each disease averaged over the 12-months testing period. Red bars represent minimal NLPD value for given predictions (the best possible uncertainty estimation), while blue bars represent actual uncertainty quality obtained for NLPD metric in experiments. Values are sorted in descending order to show rate of the blue and red surface, where the model that has more dominant blue surface is better. We can see that GCRF fails to reach the minimal NLPD for its predictions, while uGCRF does a much better job. However, the best performance is provided by ufGCRF, where we see that obtained NLPD is very close to the
minimal NLPD for given predictions in most cases. Both uGCRF and ufGCRF appear to estimate uncertainty much closer to the true error variance, with ufGCRF having superior performance in both uncertainty estimation and predictive power among two extensions. Also, note that both uGCRF and ufGCRF outperform the GCRF model for each disease in terms of uncertainty estimation.

We have demonstrated the power of the new solution for the GCRF bias on a single step ahead prediction, and we move further to multi-steps-ahead prediction and modeling input noise in the GCRF.
Table 2.1: RMSE results of admission rate prediction. First column are averaged results over 12 months of prediction period and the following columns are prediction results for each month in the year 2011. The best results are bolded, second best are underlined and third best are in italic.

<table>
<thead>
<tr>
<th>RMSE</th>
<th>Average</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
<th>July</th>
<th>August</th>
<th>September</th>
<th>October</th>
<th>November</th>
<th>December</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR lag1</td>
<td>0.3168</td>
<td>0.3027</td>
<td>0.3267</td>
<td>0.3504</td>
<td>0.3083</td>
<td>0.3004</td>
<td>0.3153</td>
<td>0.3119</td>
<td>0.3196</td>
<td>0.3154</td>
<td>0.3112</td>
<td>0.3142</td>
<td>0.3204</td>
</tr>
<tr>
<td>LR lag2</td>
<td>0.3296</td>
<td>0.3208</td>
<td>0.3262</td>
<td>0.3644</td>
<td>0.3191</td>
<td>0.3051</td>
<td>0.3228</td>
<td>0.3230</td>
<td>0.3331</td>
<td>0.3341</td>
<td>0.3320</td>
<td>0.3326</td>
<td>0.3393</td>
</tr>
<tr>
<td>LR lag3</td>
<td>0.3536</td>
<td>0.3483</td>
<td>0.3476</td>
<td>0.3967</td>
<td>0.3509</td>
<td>0.3312</td>
<td>0.3553</td>
<td>0.3465</td>
<td>0.3487</td>
<td>0.3588</td>
<td>0.3490</td>
<td>0.3535</td>
<td>0.3566</td>
</tr>
<tr>
<td>GP lag1</td>
<td>0.3098</td>
<td>0.3013</td>
<td>0.3306</td>
<td>0.3309</td>
<td>0.2933</td>
<td>0.2935</td>
<td>0.3097</td>
<td>0.3063</td>
<td>0.3176</td>
<td>0.3124</td>
<td>0.3035</td>
<td>0.2988</td>
<td>0.3200</td>
</tr>
<tr>
<td>GP lag2</td>
<td>0.3129</td>
<td>0.3073</td>
<td>0.3281</td>
<td>0.3338</td>
<td>0.2940</td>
<td>0.2952</td>
<td>0.3132</td>
<td>0.3127</td>
<td>0.3221</td>
<td>0.3141</td>
<td>0.3071</td>
<td>0.3029</td>
<td>0.3236</td>
</tr>
<tr>
<td>GP lag3</td>
<td>0.3158</td>
<td>0.3092</td>
<td>0.3299</td>
<td>0.3354</td>
<td>0.2961</td>
<td>0.2949</td>
<td>0.3182</td>
<td>0.3172</td>
<td>0.3239</td>
<td>0.3160</td>
<td>0.3092</td>
<td>0.3061</td>
<td>0.3330</td>
</tr>
</tbody>
</table>

Table 2.2: NLPD results of admission rate prediction. First column are averaged results over 12 months of prediction period and the following columns are prediction results for each month in the year 2011. The best results are bolded, second best are underlined and third best are in italic.

<table>
<thead>
<tr>
<th>NLPD</th>
<th>Average</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
<th>July</th>
<th>August</th>
<th>September</th>
<th>October</th>
<th>November</th>
<th>December</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR lag1</td>
<td>-0.0092</td>
<td>-0.0085</td>
<td>-0.0082</td>
<td>-0.0082</td>
<td>-0.0083</td>
<td>-0.0084</td>
<td>-0.0083</td>
<td>-0.0083</td>
<td>-0.0083</td>
<td>-0.0083</td>
<td>-0.0083</td>
<td>-0.0083</td>
<td>-0.0083</td>
</tr>
<tr>
<td>LR lag2</td>
<td>-0.0075</td>
<td>-0.0085</td>
<td>-0.0077</td>
<td>-0.0071</td>
<td>-0.0069</td>
<td>-0.0071</td>
<td>-0.0076</td>
<td>-0.0077</td>
<td>-0.0076</td>
<td>-0.0077</td>
<td>-0.0077</td>
<td>-0.0077</td>
<td>-0.0064</td>
</tr>
<tr>
<td>GP lag1</td>
<td>-0.0079</td>
<td>-0.0081</td>
<td>-0.0079</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0066</td>
</tr>
<tr>
<td>GP lag2</td>
<td>-0.0079</td>
<td>-0.0086</td>
<td>-0.0080</td>
<td>-0.0078</td>
<td>-0.0079</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0080</td>
<td>-0.0079</td>
<td>-0.0079</td>
<td>-0.0079</td>
<td>-0.0079</td>
<td>-0.0067</td>
</tr>
<tr>
<td>GCRF + LR</td>
<td>-0.0044</td>
<td>-0.0047</td>
<td>-0.0044</td>
<td>-0.0046</td>
<td>-0.0046</td>
<td>-0.0046</td>
<td>-0.0046</td>
<td>-0.0044</td>
<td>-0.0043</td>
<td>-0.0043</td>
<td>-0.0043</td>
<td>-0.0043</td>
<td>-0.0038</td>
</tr>
<tr>
<td>uGCRF + LR</td>
<td>-0.0055</td>
<td>-0.0088</td>
<td>-0.0094</td>
<td>-0.0086</td>
<td>-0.0086</td>
<td>-0.0087</td>
<td>-0.0087</td>
<td>-0.0087</td>
<td>-0.0087</td>
<td>-0.0087</td>
<td>-0.0086</td>
<td>-0.0086</td>
<td>-0.0070</td>
</tr>
<tr>
<td>uGCRF + GP</td>
<td>-0.0046</td>
<td>-0.0048</td>
<td>-0.0046</td>
<td>-0.0047</td>
<td>-0.0047</td>
<td>-0.0048</td>
<td>-0.0047</td>
<td>-0.0046</td>
<td>-0.0045</td>
<td>-0.0045</td>
<td>-0.0045</td>
<td>-0.0045</td>
<td>-0.0040</td>
</tr>
</tbody>
</table>

2.7 Multi-steps-ahead prediction with uncertainty propagation

This study is aimed to support structured regression for long-term decision making, which has been of interest in many high impact applications. For example, to prepare beds, personnel and medications, hospitals are interested in estimating future number of patients admitted in different departments. Providing long-term predictions of healthcare trends could immensely increase quality of decisions, which could lead to better hospital care (Dey et al., 2014; Stiglic et al., 2014; Ghalwash et al., 2014; Gligorijevic et al., 2015b), drug coverage and health insurances for clients in need. On the other hand, long-term predictions of geophysical variables, such are predictions of precipitation and lightning strikes, are very important in agriculture, telecommunications, power systems and else-
where (Sivakumar and Berndtsson, 2010; Dokic et al., 2016), thus enabling more efficient fund management and providing more stable services.

However, good predictive accuracy is not always sufficient for the long-term decision making. Uncertainty estimation, a tally of reliability for model predictions, is an important quality indicator used for decision making (Smith, 2013). It is also reasonable to assume that reliability of a model decreases when predicting further in the future. This is due to noisiness in input data, caused by a change in distribution or accumulated error of iterative predictions, and it should be reflected in the increase of estimated uncertainty of the model predictions. This effect is often referred to as uncertainty propagation. Therefore, to decide the time point on the prediction horizon, or level of certainty with which the predictive model could be considered as useful and reliable, it is important to have a proper uncertainty propagation estimate for reasoning under uncertainty.

Thus, a particular interest of this paper is long-term forecasting on non-static networks with continuous target variables (structured regression) and proper uncertainty propagation estimate in such evolving networks. This is motivated by climate modeling of long-term precipitation prediction in spatio-temporal weather station networks, as well as prediction of different disease trends in temporal disease-disease networks.

Methods that address uncertainty propagation in multiple steps-ahead prediction can be viewed as direct and iterative (Smith, 2013). In both types, error estimates made by the models should be taken into account, to ensure uncertainty propagation. Direct methods are the ones where uncertainties can be explicitly computed while predicting in the future. They assume that input variables will be available in the entire prediction horizon. However, this is a strong assumption when predicting far ahead, thus limiting many of the models. Moreover, these methods usually need more training data than iterative methods to produce useful predictions. Therefore, we will not focus on direct methods here, and will only briefly address them towards the end of this chapter. Iterative methods, on the other hand, can provide prediction any number of steps ahead, up to the desired hori-
These methods are iteratively predicting one step ahead and use lagged predictions as model inputs, as shown in Figure 2.6. In this study we assume that input variables in each of the steps are normally distributed as $\mathcal{N}(\mu_{X_{T+k}}, \Sigma_{X_{T+k}})$, and that the new point estimate $\hat{y}_{T+k} = \mu_{T+k}$ is obtained using a predictive model.

Figure 2.6: Iterative multiple-steps-ahead prediction in a temporal network represented by a vector of input attributes ($X_t$) and target variables $y_t$ for each time step $t$. Initial $L$ time steps (green boxes) are observed and remaining steps (blue boxes) are predicted iteratively in a one time step ahead process (red box).

To address uncertainty propagation in multiple steps ahead forecasting on evolving networks we propose a novel iterative uncertainty propagation model for structured regression that extends Continuous (Gaussian) Conditional Random Fields (GCRF) (Radosavljevic et al., 2010, 2014). In the proposed model, uncertainty that naturally comes from the data itself is taken into account when estimating uncertainty of the model predictions. Such setup enables iterative multiple-steps-ahead prediction with the GCRF as an iterative uncertainty propagation method.

In the past, iterative methods were developed for the Gaussian models (Girard et al., 2003; Candela et al., 2003; Kocijan et al., 2004, 2005), however, without strong empirical analysis. Moreover, to the best of our knowledge this is the first study addressing
uncertainty propagation through iterative propagation methods for regression on graphs.

To evaluate the quality of the proposed method, we compare to the several benchmarks from the group of unstructured iterative models: iterative Linear Regression (ILR) (Smith, 2013) and iterative Gaussian Processes (IGP) (Girard et al., 2003; Candela et al., 2003). Results show clear evidence of benefits of using structure information for prediction and uncertainty propagation estimation in multiple-steps-ahead setup.

2.7.1 Uncertainty Propagation Models

We describe two models found in the available literature capable of including noise in multi-step-ahead prediction.

2.7.2 Gaussian Processes as an iterative uncertainty propagation model

We refer to the description of the Gaussian Processes provided in Section 2.6.1.

In order to take account of the uncertainty of future predictions which provide the 'inputs’ for estimating further means and uncertainties, test points are considered as random inputs $x_* \sim \mathcal{N}(\mu_{x_*}, \Sigma_{x_*})$ and a Gaussian approximation shown in (Girard et al., 2003; Candela et al., 2003; Girard and Murray-Smith, 2005). Then, the predictive distribution of this iterative uncertainty propagation method has mean and variance

$$
\mu_* = k(\mu(x_*))^T C^{-1} y,
$$

$$
\sigma_*^2 = \sigma^2(\mu(x_*)) + \frac{1}{2} Tr \left( \Sigma_{x_*} \frac{\partial^2 \sigma^2(x_*)}{\partial x_* \partial x_*} \right) + \frac{\partial \mu(x_*)}{\partial x_*}^T \sum_{x_*} \frac{\partial \mu(x_*)}{\partial x_*} \Sigma_{x_*}
$$

(2.34)

This approach has been successfully applied in the past in a model-based predictive control framework for control of pH process benchmark (Kocijan et al., 2004).

2.7.3 Linear regression as an iterative and direct model

From the family of direct uncertainty propagation models we will use a linear parameterized model (DLR) (Smith, 2013), also described in Section 2.6.1. Linear regression form
representation is:

\[ y = Xw^T + \varepsilon, \varepsilon \sim \mathcal{N}(0, \sigma_y^2) \]  

(2.35)

where \( w \) is an unknown set of weights. The weight and noise variance are estimated by

\[ \hat{w} = (X^TX)^{-1}X^Ty \]

and

\[ \sigma_y^2 = \frac{(y - X\hat{w}^T)(y - X\hat{w}^T)}{N - d - 1} \]  

(2.36)

where \( X \) is matrix representation of all data available for training, \( N \) is the number of training examples and \( d \) is the number of attributes. For the auto-regressive representation of Linear model we have variance estimation, given prediction \( y_{T+k} \) (Smith, 2013):

\[ \sigma_{T+k}^2 = \sigma_y^2 \left( 1 + X_{T+k}(X^TX)^{-1}x_{T+k}^T \right). \]  

(2.37)

Then, the construction of confidence intervals for the new prediction \( y_{T+k} \) are given by T-distribution with \( n - d - 1 \) degrees of freedom for \((1 - \alpha) \times 100\%\) interval estimator

\[ y_{T+k} \pm t_{n-d-1, \alpha/2} \hat{\sigma} \sqrt{1 + x_{T+k}(X^TX)^{-1}x_{T+k}^T}. \]  

(2.38)

In the experimental section, this model will be noted as DLR. We will also apply it in an iterative setup, and call it ILR.

### 2.7.4 GCRF for direct uncertainty propagation (DGCRF)

To enable the GCRF model to propagate directly, we could allow it to be sensitive to the uncertainty of the unstructured predictors, as described in Section 2.6.2.

Results of direct uncertainty propagation with GCRF will be provided in Section 2.7.9.

### 2.7.5 Structured Regression Models

In this section we first describe a graph based structured regression model as extension to the GCRF model, followed by a description of the proposed long-term predictive model as well as mathematical background for forecasting from noisy inputs.

For readability purposes we provide a table of notation symbols:
Symbol | Notation meaning
---|---
$X_t/Y_t$ | Input/Output variables for entire network in time step $t$
$x_i/y_i$ | Input/Output variable for particular node in network
$x_i^{(d)}$ | $d$'th dimension of input variable for node $i$
$\mu_x$ | Means of inputs distributions
$\{\Sigma_{X*}\}$ | Covariances of inputs distributions
$\mu_*$ | Predictive mean
$\Sigma_{**}$ | Predictive variance

**Adaptive Gaussian Conditional Random Fields**

Gaussian Conditional Random Fields (GCRF) (Section 2.4) is a structured regression model that depends on separately learned unstructured predictors $R_k$’s and similarity matrices $S^{(l)}$’s. To enable the model to learn unstructured predictors and similarity matrices jointly with the rest of its parameters the model takes the following log-linear form (Gligorijevic et al., 2016c; Stojanovic et al., 2016a):

$$p(y|x) = \frac{1}{Z} \exp \left( \sum_{k=1}^{K} \sum_{i=1}^{N} \alpha_k (y_i - R_k(X, \theta_k))^2 + \sum_{l=1}^{L} \sum_{i \sim j} \beta_l S^{(l)}(x_i, x_j, \psi_l)(y_i - y_j)^2 \right) \tag{2.39}$$

where $\alpha$ and $\beta$ are parameters of the feature functions, which model the association of each $y_i$ and $X$, and the interaction between different $y_i$ and $y_j$ in the graph, respectively. Here, $R_k(X, \theta_k)$ functions are any specified unstructured predictors that map $X \rightarrow y_i$ independently, and might also be used to incorporate domain specific models. $\theta_k$ are parameters of the $k$-th unstructured model. Similarity metric $S^{(l)}(X, \psi_l)$ is used to define the weighted undirected graph structure between labels, for which parameters $\psi_l$ might be specified.

Learning the structure via a predefined similarity metric, rather than simply using a given graph structure, is a novelty for the GCRF model considered in this study. We have, thus, enabled this model to perform structure learning of node labels correlations. Both the unstructured predictors and similarity metrics should be differentiable such that all...
parameters can be optimized using gradient methods. To ensure positive semi-definiteness we enforce positivity of parameters $\alpha$ and $\beta$.

The final form of this GCRF model is defined by its mean $\mu$ and covariance $\Sigma^{-1}$ matrices which we specify as

$$\Sigma^{-1} = \begin{cases} 2\alpha + 2 \sum_l \sum_q \beta_l S_l^{(l)}(x_i, x_q, \psi_l), i = j \\ -2 \sum_l \beta_i S_l^{(l)}(x_i, x_j, \psi_l), i \neq j \end{cases}$$

(2.40)

and:

$$\mu = 2\Sigma \left( \sum_k \alpha_k R(x_i, \theta_k) \right).$$

(2.41)

Quadratic form in Eq. 2.39 can be represented as a multivariate Gaussian. This specific way of modeling allows efficient inference and learning. Additionally, the GCRF model can, due to its Gaussian form, intrinsically highlight areas of the input space where prediction quality is poor by indicating the higher variance around the predicted mean.

**Learning the GCRF model:** The learning task is to optimize parameters $\hat{\alpha}, \hat{\beta}, \hat{\theta}, \hat{\psi}$ by maximizing the conditional log-likelihood,

$$\hat{\alpha}, \hat{\beta}, \hat{\theta}, \hat{\psi} = \arg\max_{\alpha, \beta, \theta, \psi} \log P(y|X; \alpha, \beta, \theta, \psi).$$

(2.42)

All the parameters are learned by a gradient-based optimization. Note that both the unstructured predictors and similarity metrics should be differentiable such that all parameters can be optimized using gradient methods.

Gradients of the conditional log-likelihood are
Maximizing the conditional log-likelihood is a non-convex, however smooth objective, and can be optimized using standard Quasi-Newton optimization techniques. Partial derivatives in Eq. 2.43 are specific for the choice of unstructured predictor and similarity metric. Note that $R_k(x, \theta_k)$ and $S(p, l)$ functions can be any differentiable predictor and similarity function. The GCRF model is Gaussian and, therefore, the maximum a posteriori estimate of $y$ is obtained at the expected value $\mu$ of the GCRF distribution.

In this paper, for the simplicity, the choice of the unstructured predictor is a linear function $R_k(x, \theta_k) = x^T \theta_k$, and choice of parametrized similarity metric is the Gaussian kernel similarity:

$$S(x_i, x_j, \psi) = \psi_0 \exp \left( -\frac{1}{2} \sum_{d=1}^{D} \frac{(x_{i,d} - x_{j,d})^2}{\psi_d^2} \right).$$  

(2.44)

The optimization problem of this model is non-convex. Therefore, there are no guarantees that the solution will be optimal, as the model can be potentially optimized to a local minimum (Radosavljevic et al., 2014). However, a good initialization of parameters based approach should lead to a close to optimal solution for such deep architectures as the one proposed in this paper (Bengio, 2012).

Presented here is a more complex, non-convex generalization of the GCRF model from Section 2.4, where $R$ and $S$ are learned within the GCRF framework. This extension
will remove the bias of using pre-trained inputs. However, the bias will still be present in the form of chosen $R$ and $S$ functions. The tradeoff between model convexity and performance is a well-studied topic and a number of studies have pointed out that convexity does not necessarily lead to the more powerful models (Bengio et al., 2007; LeCun et al., 1998). The new model will optimize the $R$ and $S$ for the overall regression goal and as such will improve its representational power.

It should be noted that model now possesses larger epistemic uncertainty, however due to lower bias, aleatoric uncertainty will be lowered, overall resulting in uncertainty estimated of higher quality. To completely capture the noise, we must model the noise coming from the input data properly. Traditionally, this has been performed by using various sampling techniques, while we propose a novel sampling-free approach described in detail in further text.

### 2.7.6 Uncertainty propagation by modeling noisy inputs

Uncertainty estimation should always take into account uncertainty that naturally comes from the data itself. Our approach gravitates around inclusion of uncertainty coming from input variables, when previously obtained predictions are used as inputs. Such setup enables multiple-steps-ahead prediction with the GCRF as an iterative uncertainty propagation method, which could be applied in practice for the problems addressed in this study.

In order to model the distribution of input variables, a reasonable assumption is that input variables $x$ are generated by some process $u$, and that process has a Gaussian error. Thus, the distribution of input variables can be presented as $p(x) = \mathcal{N}(u, \Sigma_x)$. The new data point for prediction will be annotated as $x_*$. In the general case, we predict on the entire set of points representing a single snapshot of a network, so we annotate these testing points with $X_*$. The distribution of the target variable can then be expressed by the marginalization of
input variables distribution:

\[ p(y_*|D) = \int p(y_*|X_*, D)p(X_*)dX_*. \]  \hspace{1cm} (2.45)

As the distribution of \( p(y_*|X_*, D) \) is Gaussian in the GCRF model, and the distribution of \( X_* \) is conjugate to the target variable distribution, marginal distribution \( p(y_*|D) \) is a Gaussian as well. Since this integral is intractable for estimation in most of the cases, potential ways of solving it include sampling methods, variational Bayes or direct approximation of the moments of distribution as shown in (Girard, 2004). For large or complex non-linear parametrized models, sampling-based uncertainty propagation is often computationally infeasible, thus this work approximates the moments of the resulting distribution in Eq. 2.45, similarly to (Girard et al., 2003), however applied to evolving networks.

It is useful to first formalize conditional Gaussian prediction form of the GCRF at point \( X_* \). The Gaussian of the GCRF has the form:

\[ P(y_*|X_*) = \mathcal{N}\left( \begin{bmatrix} \mu_* \\ \Sigma \end{bmatrix}, \begin{bmatrix} \Sigma & \Sigma_* \\ \Sigma_*^T & \Sigma_{**} \end{bmatrix} \right). \]  \hspace{1cm} (2.46)

with predictive mean \( \mu_* \) and variance \( \Sigma_{**} \).

In order to approximate the resulting distribution in Eq. 2.45, we approximate its first two moments. They can be expressed using the law of iterated expectation and conditional variance and solved using Laplace’s method. Such methods of uncertainty propagation that are done by truncating multi-dimensional Taylor expansions of quantities of interest to approximate uncertainty criteria are called perturbation methods in the literature. Accuracy of such an approach is governed by the order of Taylor expansion (Smith, 2013).

**Approximating first moment - the mean**

The first moment of the distribution specified in Eq. 2.45 can be estimated by the Law of iterated expectations:

\[ m(X_*) = E_{X_*}[E[y_*|X_*]] = E_{X_*}[^{\mu_*}]. \]  \hspace{1cm} (2.47)
The predictive mean $m(X_*)$ can be estimated by approximating $\mu_*$ by its first order Taylor expansion around $\mu_{X_*}$ (by $\mu_{X_*}$, we annotate the mode of the distribution of input variables of all nodes in the graph $X_*$)

$$m(X_*) = \mu_*\big|_{X=\mu_{X_*}} + J^T_{\mu_*}(X_* - \mu_{X_*}) + O(\|X_* - \mu_{X_*}\|^2), \quad (2.48)$$

where Jacobian

$$J_{\mu_*} = \nabla d \frac{\partial \mu_*}{\partial X_*} \big|_{X=\mu_{X_*}}. \quad (2.49)$$

The expected value of this Taylor expansion yields

$$E_{X_*}[\mu_*] = E_{X_*}\left[\mu_*\big|_{X=\mu_{X_*}} + J^T_{\mu_*}(X_* - \mu_{X_*})\right] = \mu_* \quad (2.50)$$

We can see that, within the first order Taylor expansion, the prediction mean at any $y_*$ does not provide any correction over the zeroth order.

**Approximating second moment - the variance**

The second moment is estimated by the Law of conditional variance:

$$v(X_*) = E_{X_*}[\text{var}_{y_*}(y_*|X_*)] + \text{var}_{X_*}(E_{y_*}[y_*|X_*]) = E_{X_*}[\Sigma_{**}] + \text{var}_{X_*}(\mu_*) \quad (2.51)$$

In order to obtain predictive variance, $v(X_*)$, on the other hand, we need to approximate $E_{X_*}[\Sigma_{**}]$ and $\text{var}_{X_*}(\mu_*)$. The natural choice for approximating $E_{X_*}[\Sigma_{**}]$ is second order Taylor expansion:

$$\Sigma_{**} = 
\Sigma_{**}\big|_{X=\mu_{X_*}} + J^T_{\Sigma_{**}}(X_* - \mu_{X_*}) + \frac{1}{2} (X_* - \mu_{X_*})^T H_{\Sigma_{**}}(X_* - \mu_{X_*}) + O(\|X_* - \mu_{X_*}\|^3), \quad (2.52)$$

where Jacobian and Hessian are:

$$J_{\Sigma_{**}} = \nabla d \frac{\partial \Sigma_{**}}{\partial X_*} \big|_{X=\mu_{X_*}}, \quad (2.53)$$
The part of the Eq. 2.52, \( \frac{1}{2} (X_{*} - \mu_{X_{*}})^T H_{\Sigma_{**}} (X_{*} - \mu_{X_{*}}) \) is solved using the expression of quadratic form under Gaussian:

\[
\int (y - \mu)^T \Sigma^{-1} (y - \mu) \mathcal{N}(\mu, \Sigma) dx = (y - \mu)^T \Sigma^{-1} (y - \mu) + \text{Tr}[\Sigma^{-1} \Sigma]. \tag{2.55}
\]

The expected value of then becomes:

\[
E_{X_{*}}[\Sigma_{**}] = \Sigma_{**} \bigg|_{X=\mu_{X_{*}}} + \frac{1}{2} \text{Tr} \left[ H_{\Sigma_{**}} \{ \Sigma_{X_{*}} \} \right], \tag{2.56}
\]

where we find a new term \( \Sigma_{X_{*}} \) introduced as variance from distribution of \( X_{*} \). The notation \( \{ \Sigma_{X_{*}} \} \) serves to signify that rather than maintaining a single covariance matrix for all nodes in the graph, we can opt for maintaining a covariance matrix for each node in the graph. This is a point where information from distribution of input variables \( X \) provides a correction over predictive uncertainty of the GCRF.

Now, we can calculate \( \text{var}_{X_{*}}(\mu_{*}) \) using previously obtained \( m(X_{*}) \)

\[
\text{var}_{X_{*}}(\mu_{*}) \approx \text{var}_{X_{*}} \left( \mu_{*} \bigg|_{X=\mu_{X_{*}}} + J^T_{\mu_{*}} (X_{*} - \mu_{X_{*}}) \right) = J^T_{\mu_{*}} \{ \Sigma_{X_{*}} \} J_{\mu_{*}}. \tag{2.57}
\]

After combining the previous two results for \( E_{X_{*}}[\Sigma_{**}] \) and \( \text{var}_{X_{*}}(\mu_{*}) \), we obtain the expression for predictive variance:

\[
v(X_{*}) = \Sigma_{**} \bigg|_{X=\mu_{X_{*}}} + \frac{1}{2} \text{Tr} \left[ H_{\Sigma_{**}} \{ \Sigma_{X_{*}} \} \right] + J^T_{\mu_{*}} \{ \Sigma_{X_{*}} \} J_{\mu_{*}}. \tag{2.58}
\]

We see from Eq. 2.58 that there is a correction of the predictive variance influenced by the distribution of input variables via \( \{ \Sigma_{X_{*}} \} \). By solving partial derivatives, we can obtain corrected predictive variance that includes uncertainty coming from input variables. Finally solutions of the three partial derivatives are needed to complete the correction term.
expression: $J_{\mu**}$, $J_{\Sigma**}$ and $H_{\Sigma**}$. As we cannot analytically determine $\Sigma_{**}$ we use the derivative of an inverse rule to solve $J_{\Sigma**}$:

$$J_{\Sigma**} = -\nabla_{d}\Sigma_{**} \frac{\partial \Sigma_{**}^{-1}}{\partial x_*} \Sigma_{**}, \quad (2.59)$$

and for the Hessian $H_{\Sigma**}$:

$$H_{\Sigma**} = \nabla_{d,e} \Sigma_{**} \left( 2 \frac{\partial \Sigma_{**}^{-1}}{\partial X_*^{(d)}} \Sigma_{**} \frac{\partial \Sigma_{**}^{-1}}{\partial X_*^{(e)}} - \frac{\partial^2 \Sigma_{**}^{-1}}{\partial X_*^{(d)} \partial X_*^{(e)}^T} \right) \Sigma_{**}. \quad (2.60)$$

$$J_{\mu*} = \nabla_{d} - \Sigma_{**} \frac{\partial \Sigma_{**}^{-1}}{\partial x_*} \frac{2}{\partial \theta^T X_*} \Sigma_{**} = \Sigma_{**} \frac{2}{\partial \theta^T X_*} \Sigma_{**}, \quad (2.61)$$

where Jacobian in Eq. 2.61 is solved for case when only one linear predictor is used. First and second derivatives of $\Sigma_{**}$ can be calculated from the Precision matrix of the GCRF model.

$$\frac{\partial \Sigma_{**}}{\partial X_*^j} = \left\{ \begin{array}{ll} 2 \sum_\beta g \beta_l S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(d)}} & i = j \\ -2 \sum_\beta g \beta_l S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(e)}} & i \neq j \end{array} \right. \quad (2.62)$$

$$\frac{\partial \Sigma_{**}^{-1}}{\partial x_*^{(d)} x_*^{(e)}} = \left\{ \begin{array}{ll} 2 \sum_\beta g \beta_l (S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(d)}} \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(e)}}) + S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(d)}} \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(e)}} & i = j \\ -2 \sum_\beta g \beta_l (S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(d)}} \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(e)}}) + S(x_i, x_g, \psi_l) \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(d)}} \frac{\partial S(x_i, x_g, \psi_l)}{\partial x_*^{(e)}} & i \neq j \end{array} \right. \quad (2.63)$$

Using derivations obtained in the Eq. 2.59, 2.60, 2.61, which are specific to the GCRF model, in the equation of approximated variance 2.58, we obtain corrected variance for the GCRF model. Now the model’s predictive variance is dependent on variance of input data, assuming input data has a Gaussian error. This allows the GCRF model to be sensitive to significant changes on input data distribution, which results in higher predictive variance when predicting in the unknown.

To ensure propagation of uncertainty we should then apply the iterative approach to multiple-steps-ahead prediction, since we now include uncertainty that is accumulating from the input variables (Girard et al., 2003; Candela et al., 2003).
Uncertainty propagation

In order to properly model previous outputs as inputs as we predict ahead in time, lagged outputs are observed as random variables. The input vectors, will also be random variables, as they incorporate predictions recursively, \( X_T \sim \mathcal{N}(\mu_{X_{T+k}}, \Sigma_{X_{T+k}}) \). Note that for each node in a network we will maintain a \( \mathcal{N}(\mu_{X_{T+k}}, \Sigma_{X_{T+k}}) \) distribution. After each successive prediction, as new predicted values become inputs for the next prediction, \( \Sigma_{X_{T+k}} \) needs to be updated accordingly. In order to update \( \Sigma_{X_{T+k}} \) for the new input \( \hat{y}_{T+k} \), all we need to do is to calculate

\[
\text{cov}(\hat{y}_{T+k}, X_{T+k}) = E_x[E_y[\hat{y}_{T+k} \times X_{T+k}]] - E[\hat{y}_{T+k}]E[X_{T+k}], \tag{2.64}
\]

with \( E[\hat{y}_{T+k}] \) given as the prediction of the model and \( E[X_{T+k}] = \mu_{X_{T+k}} \). We only have to estimate expected value of product of the two random variables which can be expressed as:

\[
E_x[E_y[y_{T+k} \times X_{T+k}]] =
\int X_{T+k} \left[ \mu_{T+k} \bigg|_{X=\mu_{X_{T+k}}} + J_{\mu_{T+k}}^T (X_{T+k} - \mu_{X_{T+k}}) \right] p(X_{T+k}) dx_{T+k}. \tag{2.65}
\]

This gives,

\[
E_x[E_y[y_{T+k} \times X_{T+k}]] = \mu_{T+k} \bigg|_{X=\mu_{X_{T+k}}} \mu_{X_{T+k}} + J_{\mu_{T+k}}^T \{ \Sigma_{X_{T+k}} \}. \tag{2.66}
\]

So that the cross-covariance terms of the \( \Sigma_{X_{T+k}} \) are given by

\[
\text{cov}(y_{T+k}, X_{T+k}) = J_{\mu_{T+k}}^T \{ \Sigma_{X_{T+k}} \}. \tag{2.67}
\]

Now, that we have all components needed inference procedure that handles noisy inputs defined as lagged predictions is described as Algorithm 1.

Algorithm 1. Multiple-steps-ahead GCRF regression
Input: Test data $X$, model $(\alpha_k, \beta_l, \theta_k, \psi_l)$

1. Initialize $\Sigma_{X*}$ for each node in a graph with all zeroes

2. Make a one-step-ahead prediction of $\hat{y}_{T+1}$

   for $k = 2...K$
   
   3. Update inputs according to the previous predictions $\hat{y}_{T+k-1}$
   
   4. Update $\{\Sigma_{X*}\}$ for the previously introduced noisy input using Eq. (2.67)
   
   5. Predict following time step $\hat{y}_{T+k}$ using non-corrected models predictions and
      Eq. 2.58

   end for

end

2.7.7 Experimental results

Set-up of the experiments conducted on two real-world datasets from the medical and climate domains, and results in terms of predictive error (Mean Squared Error - MSE) and plots of predictions with propagating uncertainty are reported in this section. The results of the three iterative models clearly demonstrate benefits of the structured GCRF model, which, in addition to learning a linear mapping $x \rightarrow y$, improved accuracy by including information from the graph structure.

The obtained propagation of uncertainty as the model incrementally predicts further in the future was significantly better than alternatives and it follows the change of data distribution for the GCRF model, while previously developed non structured model IGP often fails to do so. Specific findings on predicting admission and mortality rate based on inpatient discharge data, described in Section 2.2.1, and on predicting precipitation over the North West region of the US, described in Section 2.2.3, are reported in the following subsections.
Experiments on disease networks

For disease admission and for mortality rate prediction we have trained our models on 36 monthly snapshots and iteratively predicted for the following 48 months. For each disease we have used 18 months of previous target variable values as inputs for training and prediction. Admission count for each disease has been normalized. Mortality rate is defined as the number of patients that have died with a disease as the primary cause divided by the number of patients who were admitted to hospitals with the same disease as the primary one.

Experimental results on admission rate prediction

Mean Squared Error for three algorithms for one step and multiple steps ahead prediction are shown at Figure 2.7a.

![Figure 2.7: MSE of one (blue) and multiple (red) 48 months ahead predictions of admission rate (a) and mortality (b) for all 260 disease categories obtained by 3 methods.](image)

As expected, multiple-steps-ahead prediction has larger MSE when compared to the prediction of the first time step only. While accuracy of the proposed GCRF model with linear unstructured predictor is comparable to nonlinear IGP for a single step prediction of admissions and mortality rate, in both applications the extended GCRF was more accurate for the long-term predictions, which clearly demonstrates benefits of using the information from the structure.
In Figure 2.8, we show prediction and uncertainty propagation of GCRF and IGP for Septicemia disease (we do not show ILR since the accuracy was bad for multiple steps ahead prediction and the model does not provide satisfactory uncertainty propagation). We observe that there was a huge change in the test data distribution of admission rate of Septicemia vs. training distribution and so models failed to predict a huge increase in the number of sepsis related admissions that occurred after some point in future. As prediction error was accumulating, the uncertainty propagation for the extended GCRF model properly followed the errors the model was making, which is a desirable feature for a predictive model. This is due to the change of distribution of input variables, which are moving away from the distribution of the input features on the training data, causing the correction term from Eq. 2.58 to become larger and larger. However, such a behavior is not observed when using the IGP model, which is due to the stale predictions where the model’s inputs do not change as IGP’s predictions vary only slightly.

Depending on the purpose and the precision needed for decision making, we may propose that when predicting the number of admissions up to 24 months ahead, results obtained by the extended GCRF model were acceptably reliable, however after that, we should consider waiting for more input values as indicated by increased uncertainty. On the other hand, if we were to trust the IGP based self-estimate of confidence, we would make a huge error in prediction estimate, as early as after 7-th month of prediction and
uncertainly bounds would not provide any evidence that these predictions are poor.

*Experimental results on mortality rate prediction*

Results for the mortality rate prediction are shown in the Figure 2.7b, and follow similar pattern as results for disease admission rate prediction. In the following several plots presented at Figure 2.9, we show predictions using the extended GCRF with their confidence intervals for the top six killing diseases in California for the period of 2003.–2011. In all experiments the uncertainty of the model rightly reflected the noisiness of the real-time signal, which is illustrating the power of the extended GCRF model to properly model uncertainty. Additionally, in all experiments we found that there was no uncertainty propagation unless the model starts making errors in the prediction and the distribution of the inputs changes, in which case uncertainty of the model increases properly.

![Figure 2.9](image)

**Figure 2.9**: [y–axis]: Mortality rate predictions (red lines) and uncertainty estimates (gray area) by the extended GCRF for six of the most frequent diseases causing death in California for up to 48 months (4 years) ahead [x–axis]. True mortality is shown as orange lines.

*Experiments on precipitation network*

Precipitation models were trained on 48 months of data and tested on the following 96 months. As inputs, models were learned using 12 previous values of precipitation variables for each station in the region. Prediction for 96 months ahead is done such that
lagged predictions of the previous 12 months are used as input attributes of the models and prediction is done iteratively for those 96 snapshots of the network. Accuracy results obtained by three models for immediate and for long-term (96 steps ahead) predictions are shown in Figure 2.10. Again, the extended GCRF model provided the best accuracy of the used uncertainty propagation models in both one-step-ahead and multiple-steps-ahead prediction. The predictive accuracy results clearly demonstrate that the structured GCRF model with linear predictor outperforms both linear ILR and nonlinear IGP models, which were previously successfully applied on the climate domain (Drignei, 2009; Sivakumar and Berndtsson, 2010), on both single and multiple-steps-ahead prediction.

The uncertainty propagation when predicting multiple steps ahead is demonstrated in Figure 2.11, on the previously described setup. Plots show 96 months of prediction for the GCRF model in Figure 2.11a and results are compared to those of the Gaussian Processes regression in Figure 2.11b. The plot reflects that the GCRF can offer certain predictions up to about 30 months ahead, after which uncertainty is accumulated on the displayed station of the network. GCRF also accurately highlights areas with larger error across the entire predictive horizon. On the other hand, IGP in Figure 2.11b shows virtually no uncertainty propagation due to the smoothness of the prediction, even though the model is making errors. After a relatively small number of steps, predictions by the IGP model have converged and as lagged inputs did not change, there was no change in input distribution.
Thus, there was no uncertainty propagation as the IGP model has fully propagated after only a few steps.

Additional experimental results on mortality rate prediction

In addition to the top 6 killing diseases in California, shown in Figure 4 in the main paper, here we present prediction of mortality rate for top twelve killing diseases in the state of California for the extended GCRF model in Figure 2.12.

2.7.8 Experiments with adding explanatory features in iterative predictions on precipitation data

In some applications, the input variables might be available in the future. In our climate application, in addition to precipitation, there are 6 more variables at each node which we use as input attributes for each station. These variables are acquired from the NCEP/NCAR Reanalysis 1 project (Kalnay et al., 1996), which is using a state-of-the-art analysis/forecast system to predict climate parameters using past data from 1948 to the present (data available on NOAA website: http://www.esrl.noaa.gov/psd/). These 6 variables are omega (Lagrangian tendency of air pressure), precipitable water, relative humidity, temperature, u-wind, and w-wind (zonal and meridional components of the wind, respectively).

Thus, iterative models can potentially include these input variables in their predictions.
improving predictive accuracy. However, this way of modeling leads to larger input dimensionality of the models and results in more progressive uncertainty propagation, as shown in Figure 2.14. The results of this, iterative multiple steps ahead prediction with input variables in terms of predictive accuracy are shown in Figure 2.13.

**Figure 2.12**: Predictions (red lines) and uncertainty estimates (gray area) of GCRF for mortality rate of top twelve killing diseases in California (orange lines) for 48 months (4 years) ahead (x-axis).

**Figure 2.13**: MSE of one (blue) and multiple (red) steps ahead predictions of precipitation on all stations using iterative methods with included input variables.
2.7.9  Experimental result for direct models of uncertainty propagation on precipitation data

In our climate application, input variables are available over entire predictive horizon as outputs from climate models, as mentioned in Section 2.2.3. Thus, we could apply direct methods of uncertainty propagation (DLR and DGCRF described in Section 2.7.3 and 2.7.4). Note that this way of modeling does not consider lagged predictions as inputs: only features of the nodes are used as inputs. The accuracy results are shown in Figure 2.15.

In Figure 2.16 we show training and testing time steps to demonstrate the level of uncertainty propagation for direct uncertainty propagation models. The DGCRF’s uncertainty propagation is completely dependent on uncertainty propagation of the DLR model, as it is used as an unstructured predictor to the DGCRF model. We can see that uncertainty
Figure 2.15: MSE of one (blue) and multiple (red) steps ahead predictions of precipitation on all stations using direct methods increases rapidly and then stabilizes after just a few time steps and remains relatively large over the entire prediction period.

Figure 2.16: Predictions (red lines) and uncertainty estimates (gray area) of DGCRF and DLR direct models of precipitation (true values—orange line) for 96 months (8 years) ahead.
CHAPTER 3

MODELING INTERPRETABLE AND SEMANTIC-RICH DISTRIBUTED REPRESENTATIONS OF MEDICAL CONCEPTS

3.1 Motivation and problem set-up

The increased penetration of information technologies in hospital systems in recent years has enabled collections of vast amounts of medical data in the form of electronic health records (EHRs). EHRs contain detailed patient–related data collected over time including past medical history, medications, procedures, immunizations, and diagnostic findings. In addition, EHRs store information concerning all stages of inpatient care, including a patient discharge summary, a detailed report prepared by a clinician at the end of each hospital stay. This document also contains a comprehensive list of patient’s diagnostic findings, as well as the administered procedures. Clearly, such a rich source of patient–specific data presents an unprecedented opportunity to apply data-driven approaches for knowledge discovery in clinical research (Kohane, 2011).

Data mining researchers have recognized the value and potential of inpatient medical data, and have recently proposed effective mining approaches to help obtain actionable
insights for improving healthcare (Madsen, 2014). However, the modeling process is burdened by many challenges, as the data often contains sparse, heterogeneous, and incomplete information due to different hospital and insurance policies, further aggravated by non-standardized physician practices (Hripcsak and Albers, 2013). The existing tools are not fully capable of addressing such a challenging task (Chowriappa et al., 2014), and in order to make use of these multifaceted noisy data, development of novel machine learning approaches is required to allow for efficient and effective analysis. Additionally, a vast amount of medical knowledge is available, even though often incomplete (Menche et al., 2015; Sun et al., 2014b), that could be used to improve the power of these models (Gligorić and Pržulj, 2015; Sun et al., 2014a). Examples of such sources are disease and gene ontologies, protein-protein interactions, and discovered disease-gene associations from previous medical studies. Building models capable of including such available domain knowledge could dually improve over original approaches: first, domain knowledge can increase performance of the original models, and second it can allow for novel applications and discoveries not possible before.

Another important aspect of building products using medical data is that their deployment is dependent on the interpretability of obtained results. Namely, medicine is highly sensitive field where a lot of thought goes into any decision process, and decisions need to be transparent and interpretable to ensure that the risk of any potential errors is minimized and that such errors are resolved quickly. While most of good performing algorithms in machine learning behave like a black box due to their nonlinearity, their limitations in terms of interpretability are preventing them from making a large impact on medical science. Thus, in this thesis we aim to learn high quality representations of medical concepts (diagnoses, procedures, genes, ...) while being able to interpret what they mean and analyze them in depth.

In order to achieve this, we build on the neural distributed embedding approaches originally developed for Natural Language Processing (NLP).
3.2 Learning neural distributed embeddings background

Distributed neural embedding models for language (neural language models) take advantage of word order, and state the same assumption as \( n \)-gram language models that words closer in the sequence are statistically more dependent. Typically, a neural language model learns the probability distribution of the next word given a fixed number of preceding words that act as the context. More formally, given a word sequence \((w_1, w_2, \ldots, w_T)\) from the training data, the objective of the model is to maximize the average log-likelihood function,

\[
\mathcal{L} = \frac{1}{T} \sum_{t=1}^{T} \log \mathbb{P}(w_t | w_{t-b+1} : w_{t-1}), 
\]

where \( w_t \) is the \( t \)th word, and \( w_{t-b+1} : w_{t-1} \) is a sequence of \( b \) successive preceding words that act as the context to the word \( w_t \). A typical approach to approximate probability distribution \( \mathbb{P}(w_t | w_{t-b+1} : w_{t-1}) \) is to use a neural network model architecture (Bengio et al., 2003). The neural network is trained by projecting the vectors for context words \((w_{t-b+1}, \ldots, w_{t-1})\) into a latent representation with multiple non-linear hidden layers and the output softmax layer comprising \( W \) nodes, where \( W \) is a size of the vocabulary, while attempting to predict word \( w_t \) with high probability.

When working with large-scale data, the vocabulary size \( W \) can easily reach the millions. In those cases, training of the neural network becomes a challenging task, as updates of word vectors become computationally expensive. For that reason, recent approaches (Mikolov et al., 2013a) propose log-linear models which aim to reduce the computational complexity. The use of hierarchical softmax (Morin and Bengio, 2005) or negative sampling (Mikolov et al., 2013a) is shown to be effective in substantially speeding up the training.
3.3 Data description

The rapid growth in the development of healthcare information systems has led to an increased interest in utilizing the patient Electronic Health Records (EHR) in attempts to better understand human diseasome.

3.3.1 HCUP electronic health records data

For the purpose of this study we explored the State Inpatient Database (1, 2009) (SID), a set of longitudinal state-specific hospital inpatient databases. This rich dataset is provided by the Agency for Healthcare Research and Quality, and is a part of the Healthcare Cost and Utilization Project (HCUP). Specifically, we collected EHR data from SID California, containing 35,844,800 discharge records from 474 hospitals over a period of 9 years (from January 2003 to December 2011). SID data provides discharge records for each inpatient that may contain up to 25 diagnosis codes and up to 15 procedure codes in ICD9 coding schema that were applied during this particular admission of the patient. This coding schema originates from the 9th revision of the International Classification of Diseases (ICD9)\(^1\), a hierarchical coding scheme which is a part of standard diagnostic tools for epidemiology, health management, and clinical purposes. The disease coding process of

\(^1\) [http://www.who.int/classifications/icd/en/](http://www.who.int/classifications/icd/en/), accessed June 2018
EHR databases is tedious work, even under the most obvious circumstances. It requires proper application of the AHA Coding Clinic guidelines (Association et al., 2008) and the Official Guidelines for Coding and Reporting for inpatient care (for Medicare et al., 2011), and documented physician notes are mandatory for precise coding (Wiedemann, 2007). Thus, diagnoses found in the EHR records are ordered by their importance to the patient’s reason of admission and hospital stay while respecting given guidelines of diagnoses coding. As such, EHR data possess a ‘grammar’ of diagnoses and procedures codes, where contexts of different diseases and procedures in discharge records may provide significant additional information for the prediction of hospital quality indicators.

Additionally, the SID database contains information about a hospital stay, including length of stay, total charges, type of payment, insurance type, discharge month, and survival information. In total, the SID California database covers 13,004 unique disease codes (out of around 14,000 present in ICD9 schema), and 3,830 procedure codes (out of around 4,000 present in ICD9 schema). In our experiments, we limit the population to inpatients who are more than 1 year old.
3.3.2 GWAS gene catalog data

In addition to observational EHR data obtained from hospitals, we used domain knowledge data that contains genetic variations associated with a particular disease, collected from published results of various medical studies. In particular, we used the EBI-NHGRI public GWAS catalog data\(^2\), which contains disease-gene associations for more than 11,000 genes and 71 disease groups (out of 260 disease groups defined in the ICD9 Clinical Classifications Software schema). In order to create a unified mapping between genes and diseases, we map a gene to a disease group if a p-value less than \(10^{-5}\) for that gene’s single nucleotide polymorphisms (SNPs) was found. In addition to hand curating the GWAS database, we have also manually introduced gene-disease mappings from PubMed publications. The ordering of genes for each disease is possible using p-values, however, as studies are conducted on different human populations, such ordering could be potentially biased. Alternatively, we shuffle genes for each disease at each different discharge record in our experiments to ensure the removal of this bias provided by the studies.

Note that there are around 190 disease groups for which no gene associations are currently known (e.g., thyroid disorders). In order to improve the understanding of these understudied diseases, medical researchers can greatly benefit from our methods that suggest potential gene associations. Concentrating on a subset of suggested genes would significantly reduce time and monetary costs needed for research studies.

3.4 Large-Scale Discovery of Disease-Disease and Disease-Gene Associations

First, a novel route is proposed for disease phenotyping and gene discovery (Gligorijevic et al., 2016b), a critical step in the deeper understanding of medical conditions and drug discovery (Emilien et al., 2000). This work is motivated by advances in the field of natural

\(^2\) http://www.gwascentral.org, acc. June 2018
Figure 3.3: Graphical summary of the approach proposed in this study. Heterogeneous data obtained from large scale discharge records and hand curated disease-gene associations is used to jointly learn meaningful vector representations of disease and gene concepts in a latent vector space, where interactions of diseases and genes are retrieved and discovered.

Language processing (NLP) (Mikolov et al., 2013a; Djuric et al., 2015b), and is capable of seamlessly addressing the inherent issues of sparsity and heterogeneity present in medical data records. In particular, a distributed, neural embedding model is proposed for the phenotypic discovery of diseases that often co-occur in patients (referred to as disease comorbidity), and are expected to be governed by the same genetic mutations (Chen et al., 2007). Our proposed approach is further developed to allow inclusion of domain knowledge in terms of previously discovered disease-gene associations, improving over original approach on the disease phenotyping task and allowing for discoveries of previously unknown disease-gene associations. Our proposed approach is shown to be more accurate than other state-of-the-art approaches, with respect to a number of rigorous evaluation tasks. A summary of the proposed approach is illustrated in Figure 3.3.
3.4.1 Relevant background and related work in phenotyping and genotyping

When it comes to the treatment of ailments, the focus of medical practitioners can be roughly divided into two complementary directions: 1) treating the symptoms of already sick patients (reactive medicine); and 2) attempting to understand their causes in order to prevent manifestation and further spread of the disease (preventative medicine). In the first case, the disease symptoms are a part of a wider phenotype profile of an individual, with *phenotype* being defined as the presence of a specific observable characteristic in an organism, such as blood type, response to administered medication, or the presence of a disease (Newton et al., 2013). The identification process of useful, meaningful medical concepts and insights for the purposes of medical treatment is referred to as *phenotyping* (Ho et al., 2014b). Identifying the genetic basis of disease relates to *genotyping*, where researchers aim to discover the relations between exhibited phenotypes and the patient’s genetic constitution (Vissers and Veltman, 2015). Establishing a relationship between a phenotype and associated genes is a major component of *gene discovery* and allows biomedical scientists to gain a deeper understanding of the condition and a potential cure at its very origin (Horvath et al., 2001). Clearly, both phenotyping and gene discovery are important steps in the fight for global health, and advancing tools for these tasks is a critical part of this battle.

EHR records, containing abundant information relating to patients’ phenotypes that have been generated from actual clinical observations and physician-patient interactions, present an unprecedented resource and testbed to apply novel phenotyping approaches. Moreover, the data is complemented by large amounts of gene-disease associations derived from genetic experiments conducted in laboratories under controlled settings. However, current approaches for phenotyping and gene discovery using EHR data rely on highly supervised rule-based or heuristic-based methods, which require manual labor and often a consensus of medical experts (McCarty et al., 2011). This severely limits the scalability
and effectiveness of the process (Hripcsak and Albers, 2013).

Initial work on understanding human disease processes by exploiting EHR databases focused on graphical representations of diseases, genes, and proteins. Disease networks were proposed in Goh et al. (K.I. et al., 2007) where certain genes play a central role in the human disease interactome, which is defined as all interactions (connections) of diseases, genes, and proteins discovered on humans. Follow up studies by Hidalgo et al. (Hidalgo et al., 2009) proposed human phenotypic networks (commonly referred to as comorbidity networks) to map with disease networks derived from EHR datasets, which were shown to successfully associate higher connectivity of diseases with higher mortality. Based on these advances, a body of work focused to link predictions of disease-disease and disease-gene networks (Davis and Chawla, 2011; Sun et al., 2014b) where a mediocre degree of correlation (~40%, also confirmed on data used in this study) was detected between disease and gene networks, indicating potential causality between them. Such studies provided important evidence of modeling disease and human interactome networks for discovery of different phenotypes. Recently, network studies of the human interactome have focused on uncovering patterns (Ghiassian et al., 2015) and, as human interactome is incomplete, discovering novel relationships (Menche et al., 2015). However, it has been suggested that network-based approaches to phenotyping and discoveries of meaningful concepts in medicine, have yet to be fully exploited and tested (Emmert-Streib et al., 2013). This study offers a novel approach to represent diseases and genes by utilizing the same sources of data as network approaches, however in a different manner, as discussed in greater detail in the section, below.

Additionally, to create more scalable, effective tools, recent approaches distinct from networks have focused on the development of data-driven phenotyping with minimal manual input and rigorous evaluation procedures (Hripcsak and Albers, 2013; Che et al., 2015; Liu et al., 2015). Part of the emerging field of computational phenotyping includes the proposed methods of (Zhou et al., 2014) which formulates EHRs as temporal matrices of
medical events for each patient, and proposes an optimization-based technology for discovering temporal patterns of medical events as phenotypes. Further, (Ho et al., 2014a) formulated patient EHRs as tensors, where each dimension is represented by a different medical event, and the use of non-negative tensor factorization in the identification of phenotypes. Deep learning has also been applied to the task of phenotyping (Che et al., 2015), as well as graph mining (Liu et al., 2015) and clustering (Schulam et al., 2015), used to identify patient subgroups based on individual clinical markers. Finally, (Žitnik et al., 2013), conducted a study on non-negative matrix factorization techniques for fusing various molecular data to uncover disease-disease associations and show that available domain knowledge can help reconstruct known and obtain novel associations. Nonetheless, the need for a comprehensive procedure to obtain manually labeled samples remains one of the main limitations of modern phenotyping tools (Ho et al., 2014b). Although state-of-the-art machine learning methods have been utilized to automate the process, current approaches still observe degraded performance in the face of limited availability of labeled samples that are manually annotated by medical experts (Chen et al., 2012).

Experiments compare representatives of the above approaches against our proposed approach in a fair setup and, overall, demonstrate the benefits of the neural embedding approach (described below) on several tasks in a quantifiable manner.

### 3.4.2 The proposed approach

To address the shortcomings of the existing state-of-the-art methods for disease phenotyping, we propose a radically new approach, motivated by the recent success of distributed language models in Natural Language Processing (short NLP) applications (Turian et al., 2010; Mikolov et al., 2013a). In the context of NLP, distributed models can learn useful word representations in low-dimensional continuous vector spaces in an unsupervised manner, without the need for expensive labeling/annotation efforts. The methods use the surrounding context of a word in a sentence, and learn word representations such that in
the resulting embedding space semantically similar words are close to each other (Mikolov et al., 2013a). Our objective is to take advantage of this property for the task of disease phenotyping, and learn disease representations in a low-dimensional space where diseases that occur in the same contexts are nearby. As a result, and in contrast to comorbidity methods commonly used in practice, related diseases could have a high similarity score even if they do not co-occur in the same patients. This would allow identification of similar diseases through straightforward $K$-nearest-neighbor search in the disease embedding space, without using supervised signals during the learning process. A similar approach has been successfully applied to extracting features from medical texts (Wang et al., 2015). However, adopting such an approach to extract meaningful concepts from EHR databases coupled with other heterogeneous sources, as proposed in our study, is the first work of its kind.

Adapting distributed language models to the task of disease phenotyping is not an easy endeavor. Finding distributed disease representation, as opposed to finding word representations, brings very unique challenges quite different from those found in everyday NLP problems. Contrary to everyday language where linguistic rules and notions of words and sentences are clearly defined, there are no existing notions of "sentence of diseases" or surrounding contexts that are equivalent to the NLP domain, as though there are rules of the disease coding provided, it heavily depends on physicians and hospitals to follow those rules, which is not always true (thus the evident noise in the disease codings among different hospitals/physicians).

In this paper, we address these issues, and propose two methods that bring state-of-the-art distributed language models to the setting of disease phenotyping: 1) disease2vec, where we exploit inpatient discharge summaries from EHR records, from which we create "disease sentences" and apply recently proposed language model (Mikolov et al., 2013a); and 2) disease&gene2vec, where we propose a novel method to learn disease and gene vector representations simultaneously by incorporating domain knowledge regarding
known disease-gene associations into the inpatient discharge observational data. The disease&gene2vec method learns low-dimensional representations of diseases and genes in the same embedding space (Djuric et al., 2015b), which opens doors for application of the proposed method to many important tasks, such as the discovery of new disease–gene associations.

**Low-dimensional embedding models**

Let us assume that we are given a set $\mathcal{P}$ of patient discharge records and a set $\mathcal{D}$ of possible diseases. Then, a discharge record $p_i = (d_{i1}, \ldots, d_{iM_i}) \in \mathcal{P}$ of the $i^{th}$ patient is defined as a sequence of diseases $d_i \in \mathcal{D}$ at the end of a hospital stay, where $M_i$ is the number of diagnosed diseases in the sequence. Moreover, each disease $d_m \in \mathcal{D}$ is associated with $N_m$ genes, called a genotype of the disease, represented as a sequence of genes $d_m = (g_{m1}, g_{m2}, \ldots, g_{mN_m})$, $g \in \mathcal{G}$, where $\mathcal{G}$ is the set of all possible genes. Then, using the set $\mathcal{P}$, the objective is to find $D$-dimensional real-valued representations $\mathbf{v}_d \in \mathbb{R}^D$ for every disease $d$ and $\mathbf{v}_g \in \mathbb{R}^D$ for every gene $g$, such that diseases with similar phenotypes and common gene origins lie nearby in the vector space.

The main idea of neural language models has been discussed in detail in Section 3.2

**disease2vec method**

Building on the ideas of neural embedding approaches we propose the disease2vec approach for learning disease representations, more particularly we build upon ideas introduced by the word2vec algorithm (Mikolov et al., 2013a). The key insight is that we can represent the patients’ lists of diseases and medical conditions from EHRs as sequences of tokens, and view each sequence as a sample from some unknown language. Following this reasoning, the language model learns representations of diseases in a low-dimensional space using each patient discharge record as a ”sentence” and the diseases within the record as ”words”, to borrow the terminology from the NLP domain. The diseases in each
Low-dimensional disease representations are learned by maximizing the objective function $\mathcal{L}$ over the entire set $\mathcal{P}$ of records as follows,

$$
\mathcal{L} = \sum_{p \in \mathcal{P}} \sum_{d_m \in p} \sum_{-b \leq i \leq b, i \neq 0} \log \mathbb{P}(d_{m+i}|d_m).
$$

(3.2)

The probability $\mathbb{P}(d_{m+i}|d_m)$ of observing some ”neighboring” disease $d_{m+i}$ given the current disease $d_m$ is defined using the soft-max function as

$$
\mathbb{P}(d_{m+i}|d_m) = \frac{\exp(\mathbf{v}_d^T \mathbf{v}_{d_{m+i}}')}{\sum_{d=1}^{[|\mathcal{P}|]} \exp(\mathbf{v}_d^T \mathbf{v}_d')},
$$

(3.3)

where $\mathbf{v}_d$ and $\mathbf{v}_d'$ are the input and output $D$-dimensional vector representations of disease $d$, and hyper-parameter $b$ represents the length of the context for disease records.

As illustrated in Figure 3.4a and equation 3.3, disease2vec uses the central disease $d_m$ to predict $b$ diseases that come before and $b$ diseases that come after it in the discharge record. As a result, diseases that often co-occur and diseases with similar contexts
(i.e., with similar neighboring diseases) will have similar representations as learned by our model.

*disease&gene2vec method*

In the previous section we described how we can learn disease representations directly from EHR records in an unsupervised manner. However, there exists a large amount of domain knowledge related to the observed diseases, and omitting this valuable information during modeling and training stages would clearly lead to suboptimal performance of any approach (Kannry and Williams, 2013). In this section we describe *disease&gene2vec*, a method that allows straightforward incorporation of external information into the training procedure, resulting in improved vector embeddings.

The *disease&gene2vec* model assumes that a subset of diseases from the training data \( \mathcal{D} \) are associated with genes, where the associations are provided by domain experts and considered as domain knowledge. We leverage this information by assigning a vector representation to each gene, and make use of disease contexts in the training data to jointly learn both disease vectors and gene vectors in the same low-dimensional space. To this end, given the diseases associated with genes, we extend the set of patient discharge records \( \mathcal{P} \) to obtain data set \( \mathcal{P}_g \), where associated genes were added to the discharge records. In particular, assuming that a disease in the EHR record is accompanied by a non-empty set of associated genes, whenever a vector of central disease \( d_m \) is updated to predict the surrounding diseases, the vectors of genes assigned to \( d_m \) are updated as well.

More formally, assuming central disease \( d_m \) is associated with \( N_m \) of \( |\mathcal{G}| \) genes in total, \( d_m = \{g_{m1}, \ldots, g_{mN_m}\} \), the *disease&gene2vec* learns disease and gene representations by maximizing the following objective function \( \mathcal{L} \),

\[
\mathcal{L} = \sum_{p \in \mathcal{P}} \sum_{d_m \in p} \sum_{b \leq i \leq b, i \neq 0} \left( \log \mathbb{P}(d_{m+i}|d_m) + \sum_{g \in d_m} \log \mathbb{P}(d_{m+i}|g) \right). \tag{3.4}
\]
Probability \( P(d_{m+i}|g) \) of observing neighboring disease \( d_{m+i} \), given gene \( g \) associated with the central disease \( d_m \), is defined using the soft-max,

\[
P(d_{m+i}|g) = \frac{\exp(v_g^T v'_{d_{m+i}})}{\sum_{d=1}^{D} \exp(v_g^T v'_d)}.
\] (3.5)

The disease\&gene2vec model is depicted in Figure 3.4b, where we illustrate how the context disease vectors are influenced both by the central disease and by its associated genes.

We solve both 3.2 and 3.4 using stochastic gradient descent, suitable for large-scale problems. However, computation of gradients is proportional to the number of unique diseases and genes in the data, which may be computationally expensive in practical tasks. As an alternative, we use negative sampling (Mikolov et al., 2013a), which significantly reduces the computational complexity and allows fast training of the embeddings on data with millions of patient records. Lastly, once the disease and gene vectors are trained, we measure similarity between them using the cosine distance.

Both disease2vec and disease\&gene2vec models can be seen as weighted matrix factorization models of underlying disease context structure (Levy and Goldberg, 2014). This neural embedding approach can be compared to other matrix factorization models on different disease network and covariance matrices, with the advantage of being better able to explore disease co-occurrence (Levy and Goldberg, 2014).

The proposed approach has certain drawbacks in terms of modeling. For instance, parameters \( D \) and \( b \) are not automatically selected. Additionally, each disease in this study receives a single vector representation, whereas, in reality, the same disease can have several modules: for example, sepsis caused by the pneumonia and sepsis caused by external injury. Also, the current model does not consider disease hierarchical structure which can carry significant information. These issues will be addressed as a follow up: the main goal of this study is to characterize the power of disease representations of the proposed neural embedding models.
3.4.3 Experiments

Data sets used to evaluate the proposed embedding methods were described in Section 3.3, introduce baseline methods, and discuss the experimental setup and evaluation results.

Experimental setup

To demonstrate the power of the models, we evaluate them on two tasks:

- **Disease Phenotyping**: Identifying diseases with similar contexts as the query disease.

- **Disease-Genes Association Discovery**: Identifying (novel) disease-genes relations of the query disease.

For the first task, the models are trained in two set-ups. To train the disease2vec model, we used only EHR data (\(P\) dataset). To train the disease\&gene2vec model (for both tasks), we extended patient discharge records by associating diseases to corresponding genes according to GWAS data (we found 2,739 diseases that have gene associations, or 23.5% of the entire disease set) in \(P_g\) dataset. In order to remove any bias across studies, gene lists are shuffled while assigning the list to the disease.

Dimensionality of the embedding space \(D\) was explored for 11 choices in the 50 to 1,000 range. Context neighborhood size was set to \(b = 8\) chosen to be close to average number of diagnosis of all inpatient records (7.58 in our dataset). Finally, we used 25 negative samples in each vector update for negative sampling as suggested in the literature (Mikolov et al., 2013a). Following reported distributed language models (Mikolov et al., 2013a), the most frequent diseases and genes were subsampled during training.

Embedded dimensionality \(D\) was chosen to be smallest \(D\) where Precision@\(K\) starts saturating as dimensionality grows. In our experiments, this point was observed at dimensionality \(D = 200\), which provided an acceptable tradeoff between good accuracy
of the model and its training speed which scales linearly with the dimensionality. While increasing $D$, we have observed drop in accuracy and halted further examinations of dimensionality in this study.

**Figure 3.5:** Precision@$K$ for disease2vec model with different dimension $D$ of the embedding space

**Baseline models set-up** We evaluated the proposed methods against state-of-the-art approaches, such as 1) Latent Dirichlet Allocation (LDA) (Blei et al., 2003), 2) spectral clustering (Tang and Liu, 2011), and 3) modularity (Newman, 2006), which have been successfully applied to EHR analysis (Tamang and Parsons, 2011; Chen et al., 2015). The LDA model was trained using the same data as disease2vec and disease&gene2vec. The spectral and modularity models representation in $R^d$ from the first $d$ eigenvectors were trained by decomposing the Laplacian of the graph $G$ and modularity matrix of graph $G$, respectively. We define two types of graph $G$ in which:

- Nodes represent diseases and genes, and links are determined by gene co-occurrence in GWAS and disease co-occurrence in EHR data.

- Nodes represent diseases, and links are determined by the comorbidities in the EHR data as proposed in (Hidalgo et al., 2009). For each link, a Pearson correlation is defined, and link rejection decided using a t-statistic (Sun et al., 2014b) (disease-
gene network was not built using comorbidities statistics, as such an approach is not used in the literature).

It should be noted that there are other ways to generate interactome networks of human diseases (D.S. et al., 2008; Barabasi et al., 2011; Ghiassian et al., 2015; Menche et al., 2015; Davis and Chawla, 2011), however, these are not easily applicable for a general disease phenotyping task this study addresses, and as such are not included.

The diseases and genes are then mapped into a $\mathbb{R}^D$ space by projecting onto the subspace spanned by the largest eigenvectors. In order to compare to the largest body of work on disease representation, we have drawn disease phenotypes by choosing the nearest neighbors (the largest link weight) of the query disease in the 4) disease comorbidity network, as well as in the 5) diseases and diseases-genes co-occurrence network. The 5) can be seen as a baseline that for a particular disease returns neighbors that were most frequently commonly observed in the EHR data. In addition, disease comorbidity representation was calculated by applying random walks on the comorbidity network (Perozzi et al., 2014), however this approach failed to provide satisfactory results due to graph sparsity, as such, those results are omitted from the Results section.

disease2vec based disease-disease associations

In each approach we map diseases to $D = 200$ dimensional space. Then, disease-disease closeness values are measured in the embedded space using the cosine distance metric.

In the first set of experiments we evaluated the quality of disease representations obtained using the two proposed methods. Specifically, we selected 2,739 diseases found in the GWAS data and for each retrieved $K$ nearest diseases, with $K \in \{1, 2, 5, 10\}$. Each of the retrieved diseases was labeled as positive if it shares a gene with the query disease, and labeled negative otherwise, which is used as a proxy for having the same phenotype (Weatherall, 2001). Then, we computed precision@$K$ for each disease as a fraction of positive neighbors within the $K$ retrieved ones, and report the average precision over all
Table 3.1: Precision and gene overlap for various competing models for the task of phenotype discovery, evaluated using disease-gene associations

<table>
<thead>
<tr>
<th>Data</th>
<th>Average precision @K</th>
<th>Average perc. of overlapping genes @K</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>disease2vec</td>
<td>0.9449</td>
<td>0.9367</td>
</tr>
<tr>
<td>Modularity (Adjacency)</td>
<td>0.8575</td>
<td>0.8457</td>
</tr>
<tr>
<td>Spectral (Adjacency)</td>
<td>0.7181</td>
<td>0.7007</td>
</tr>
<tr>
<td>Modularity (Comorbidity)</td>
<td>0.8493</td>
<td>0.8412</td>
</tr>
<tr>
<td>Spectral (Comorbidity)</td>
<td>0.8375</td>
<td>0.8316</td>
</tr>
<tr>
<td>Comorbidity graph</td>
<td>0.7268</td>
<td>0.7134</td>
</tr>
<tr>
<td>Disease co-occurrence</td>
<td>0.5616</td>
<td>0.5516</td>
</tr>
<tr>
<td>LDA</td>
<td>0.5260</td>
<td>0.5094</td>
</tr>
<tr>
<td>disease&amp;gene2vec</td>
<td>0.9598</td>
<td>0.9444</td>
</tr>
<tr>
<td>Modularity (Adjacency)</td>
<td>0.8711</td>
<td>0.8604</td>
</tr>
<tr>
<td>Spectral (Adjacency)</td>
<td>0.9165</td>
<td>0.9102</td>
</tr>
<tr>
<td>Disease and genes co-occurrence</td>
<td>0.6978</td>
<td>0.6985</td>
</tr>
<tr>
<td>LDA</td>
<td>0.5795</td>
<td>0.3874</td>
</tr>
</tbody>
</table>

2,739 diseases in Table 3.1.

\[
\text{precision@K} = \frac{1}{|D|} \sum_{\text{disease}_i} \frac{\# \text{positive neighbors } \text{disease}_i}{K}
\]  

(3.6)

Our proposed methods outperform other approaches by a significant margin, for all values of \(K\) and for both training data sets, suggesting the suitability of the approaches for the task of phenotyping.

Moreover, as each disease has more than one gene associated with it in the GWAS data, we computed an overlap of the genes between the query disease and its neighbor. Then, for each query disease we computed the percentage of overlapping genes (Sun et al., 2014b) as

\[
\text{percentage of overlapping genes@K} = \frac{\# \text{overlapped genes}}{\text{total } \# \text{genes}},
\]  

(3.7)

giving a stronger measure of genetic similarity between the neighboring diseases. We report the average overlap over all diseases in the right side of Table 3.1. Again, based on the reported results we find that our proposed approaches obtained the best results, providing much better disease representations than any of the state-of-the-art methods.
Case studies of disease2vec-based retrieval of disease-disease associations

To illustrate the usefulness of the disease2vec model we discuss disease-disease associations discovered by this approach in the context of four specific high-impact diseases. Case studies demonstrate the power of improved disease phenotyping in increasing clinical knowledge by both generating novel association discoveries and decreasing uncertainty by validating assumptions physicians may hold. We demonstrate the potential impact of using very large patient databases to answer a variety of questions clinicians may ask, as well as providing potential evaluation directions. Our provided case studies are meant to deepen the readers’ understanding of embedding model behavior. In each case study, the top ten most related disease conditions in the embedded space are retrieved, and their associations are discussed.

As a reminder, disease2vec is using only EHR records (list of diseases a patient was diagnosed) and no domain knowledge information. The model is then learning vector representation for each of the diseases such that contextually similar diseases are closer in the embedded space. Displayed use cases show that embedded space can be characterized as discovering conditions in phenotypes that are i) a similar condition (including same disease present on different organ), ii) different stages of the same condition, and iii) causative and/or effective conditions to central disease.

Case study 1: Chronic kidney disease Stage I (ICD-9 code: 585.1)  As an example, we show the nearest neighbors of Chronic kidney disease Stage I (CKD) in Table 3.2. The model was able to learn to accurately map within its closest proximity (given are values of Cosine similarity) successive stages of this chronic disease without including any domain knowledge. The recovery of disease stages was observed in other case studies, including high fatality diseases of acute myocardial infraction (ICD-9 code: 410.00) and lung cancer (ICD-9 code: 162.9).
Table 3.2: The four nearest disease neighbors for Chronic Kidney Disease Stage I

<table>
<thead>
<tr>
<th>Phenotype disease</th>
<th>Cosine Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chronic kidney disease Stage II (mild)</td>
<td>0.9361</td>
</tr>
<tr>
<td>Chronic kidney disease Stage III (moderate)</td>
<td>0.8652</td>
</tr>
<tr>
<td>Chronic kidney disease Stage IV (severe)</td>
<td>0.7647</td>
</tr>
<tr>
<td>Chronic kidney disease unspecified</td>
<td>0.6923</td>
</tr>
</tbody>
</table>

Case study 2: Multiple Sclerosis (ICD-9 code: 340)  
Multiple sclerosis, a chronic disease involving damage to the sheaths of nerve cells in the brain and spinal cord, is discussed next (Table 3.3). The reasons for this disease are not yet well understood, but the autoimmune process appears to be caused both by genetic and environmental factors - e.g., viral infections in early life (National, 2014). Discovered associations in this case study support these statements. From the top 10 retrieved phenotypes, we observe that different inflammations of neural tissue (e.g., spinal cord, optical nerves, brain), late effects of neural tissue bacterial infections as well as late effects of nervous system injuries are highly ranked. A better understanding of these inflammations, bacterial infections, and physical injuries and their relation to multiple sclerosis may help address the heterogeneity found in patients, and also improve the treatment of the disease, including prevention in some cases. Moreover, high ranks of different scleroses (including notorious ALS disease) and spina bifida (a birth defect in which a baby’s spinal cord fails to develop properly) may strongly indicate that diseases in this phenotype are determined by the genes their carriers possess. The ranked list of genes identified by this study can be found in the Supplement.

Case Study 3: Septicemia (ICD-9 code: 995.91)  
Sepsis (blood infection) is a condition caused by an overwhelming immune response to infection. From the left side of Table 3.4 we observe that "Severe sepsis" and "Septic shock" are discovered as the most related to the disease code, "Sepsis", validating previously known relationships (given that these could be considered as stages of sepsis in general). More surprising neighboring disease codes include infections (both bacillus and non-bacillus, including fungi that easily pene-
Table 3.3: Ten nearest disease neighbors of the Multiple Sclerosis phenotype retrieved by the disease2vec

<table>
<thead>
<tr>
<th>Late effect of spinal cord injury</th>
<th>Other causes of myelitis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neuromyelitis optica</td>
<td>Acute infective polyneuritis</td>
</tr>
<tr>
<td>Late effects of intracranial abscess or pyogenic infection</td>
<td>Late effects of viral encephalitis</td>
</tr>
<tr>
<td>Acute (transverse) myelitis NOS</td>
<td>Amyotrophic lateral sclerosis</td>
</tr>
<tr>
<td>Spina bifida without mention of hydrocephalus unspec. region</td>
<td>Primary lateral sclerosis</td>
</tr>
</tbody>
</table>

Late effect of spinal cord injury and other causes of myelitis, neuromyelitis optica, acute infective polyneuritis, late effects of intracranial abscess or pyogenic infection, late effects of viral encephalitis, acute (transverse) myelitis NOS, amyotrophic lateral sclerosis, spina bifida without mention of hydrocephalus unspecified region, and primary lateral sclerosis are some of the nearest disease neighbors retrieved by the disease2vec model for the Multiple Sclerosis phenotype. These diseases are related to the Multiple Sclerosis phenotype in various ways, such as being caused by similar mechanisms or having similar symptoms. Using this knowledge about related phenotypes may help physicians react earlier to potential septic cases and reduce mortality of the biggest killer disease in California (e.g., by reacting earlier to an infection that has not turned septic yet).

Table 3.4: Ten nearest disease neighbors for the Sepsis and Congestive heart failure phenotypes retrieved by the disease2vec

<table>
<thead>
<tr>
<th>Sepsis</th>
<th>Congestive heart failure unspecified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severe sepsis</td>
<td>Other primary cardiomyopathies</td>
</tr>
<tr>
<td>Septic shock</td>
<td>Atrial fibrillation</td>
</tr>
<tr>
<td>Intestinal infection due to Clostridium difficile</td>
<td>Other specified forms of chronic ischemic heart disease</td>
</tr>
<tr>
<td>Candidiasis of other urogenital sites</td>
<td>Atrial flutter</td>
</tr>
<tr>
<td>Other and unspecified mycoses</td>
<td>Other chronic pulmonary heart diseases</td>
</tr>
<tr>
<td>Systemic inflammatory response syndrome</td>
<td>Paroxysmal ventricular tachycardia</td>
</tr>
<tr>
<td>Hyperosmolality and-or hypernatremia</td>
<td>Cardiac pacemaker</td>
</tr>
<tr>
<td>Pressure ulcer stage III</td>
<td>Aortic valve disorders</td>
</tr>
<tr>
<td>Proteus infection</td>
<td>Other left bundle branch block</td>
</tr>
<tr>
<td>Other shock without mention of trauma</td>
<td>Old myocardial infarction</td>
</tr>
</tbody>
</table>

Case Study 4: Congestive Heart Failure (ICD-9 code: 428.0)  From the family of heart diseases we show disease-disease associations for one of the most deadly diagnoses. Congestive Heart Failure (CHF), a disease in which the heart becomes weaker over time (i.e.,
heart’s pumping power is weaker than normal) while usually expanding its volume. Discovered disease-disease associations for CHF (Figure 3.4 right side) are dominated by conditions involving asynchronous heartbeat due to fibrillation, flutter, tachycardia and blockades of cardiac nerve that often cause these asynchronous heartbeat conditions. Longer periods of asynchronous heartbeat cause weakening of heart muscle due to irregular blood flow. Pulmonary disorders can lead to pulmonary hypertension which can result in heart failure. A similar mechanism can be caused by chronic kidney failure condition; however, it is not present in the phenotype. The reason for such an oversight can be lack of chronic kidney disease instances in HCUP database, due to the fact that patients suffering from chronic kidney diseases are regular visitors to the hospitals (regular dialysis treatment) and are not considered inpatients. Thus, proposed models would not be able to learn proper vectors for such a condition, indicating a limitation of the study, but not a weakness of the proposed approach. Heart disorders of aortic valve and hearth cells are also present in the CHF phenotype, and represent indicators of scars on the heart, that are well correlated to the CHF. Similar traits, as in picking disease causes in the phenotype have been observed in three more high fatality diseases: pneumonia (ICD-9 code: 486), acute respiratory failure (ICD-9 code: 518.81) and renal failure (ICD-9 code: 586).

**disease&gene2vec based disease-genes associations**

In order to evaluate the potential power of the *disease&gene2vec* model to identify gene-disease associations, we conducted the following experiment. First, we randomly selected 20% of the diseases with associated genetic data as a test set (i.e., the diseases found in the GWAS dataset). Although we removed genetic information, these diseases still remain in the original data, so that we are able to learn their vector representations. Second, we learn *disease&gene2vec* on the data set where the remaining 80% of diseases were associated with gene information data. Finally, the *disease&gene2vec* model will contain low-dimensional representations of diseases and genes in the same embedding space. We
compared $disease&gene2vec$ to modularity, spectral, and LDA methods, trained on $P_g$ data. Graphs for modularity and spectral were constructed such that diseases and genes represent nodes where links between diseases were based on co-occurrence information, while links between genes and diseases were created based on disease-gene associations. Having learned disease and gene representations for each of the diseases from the test set we found the top $K$ genes based on similarity in a low-dimensional space. In addition, two trivial predictors are included: disease-gene co-occurrence, predicting genes that most commonly appeared in records and most frequent gene, always predicting the most commonly occurring genes.

Like the previous section, in Table 3.5 we report averaged precision@$K$, which is defined as a percentage of genes that are correctly identified out of top $K$ retrieved. We find that the proposed $disease&gene2vec$ method outperforms the baselines for almost all values of $K$, except on the very challenging prediction of $K = 1$, where $disease&gene2vec$ was second best. Interestingly, LDA is the least accurate one in both $disease&gene2vec$ and $disease2vec$ experiments, which can be explained by the fact that this method performs poorly on short ”documents” (the average patient record in our data has only 7.62 diagnoses).

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>5</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$disease2vec$</td>
<td>0.6978</td>
<td><strong>0.8056</strong></td>
<td>0.7293</td>
<td>0.6711</td>
</tr>
<tr>
<td>Modularity</td>
<td>0.4760</td>
<td>0.4874</td>
<td>0.4902</td>
<td>0.4689</td>
</tr>
<tr>
<td>Spectral</td>
<td><strong>0.7803</strong></td>
<td>0.7551</td>
<td>0.6705</td>
<td>0.6387</td>
</tr>
<tr>
<td>LDA</td>
<td>0.2300</td>
<td>0.2570</td>
<td>0.3560</td>
<td>0.4180</td>
</tr>
<tr>
<td>Co-occurrence</td>
<td>0.4691</td>
<td>0.3867</td>
<td>0.2998</td>
<td>0.2416</td>
</tr>
<tr>
<td>Most Frequent</td>
<td>0.2000</td>
<td>0.3467</td>
<td>0.4324</td>
<td>0.3887</td>
</tr>
</tbody>
</table>

To delve deeper into the obtained results, the top results obtained in disease-gene association discovery is provided: we identified all 185 of 185 genes known to be associated via the GWAS to congestive heart failure (ICD-9 code: 428.0), 90/90 genes associated
to hypothyroidism (ICD-9 code: 244.9), 108/111 for chronic airway obstruction (ICD-9 code: 496), and 100/111 for osteoarthrosis (ICD-9 code: 715.90).

The three genes found for chronic airway obstruction, not present in the 111 genes identified from GWAS studies, are SH3RF1, LOC645177 and SPAG16. While examining the literature, we found that all three genes have similar levels of gene expression in a number of tissues including the lungs. Additionally, proteomic assays reveal high expression in platelet blood cells for SH3RF1\(^3\), which are shown to influence chronic airway obstruction in the past (Cordova et al., 1985) and in bone marrow stem cells for SPAG16\(^4\).

As most diseases have no available genetic associations (i.e., gene-disease associations were not available from the EBI-NHGRI GWAS catalog), we find that discovered associated genes often have protein and/or microarray expression in an associated tissue or that there is a mechanism that can potentially explain certain non-obvious associations, which will hopefully be unraveled in the near future by genetic research. The full list of genes ranked according to the disease&gene2vec model is provided in the Supplement for further examination and as a resource for future genetic research.

3.4.4 Conclusions

We proposed a novel model for phenotyping and gene discovery, building upon the latest advances in neural language models. The described approaches allow for unsupervised learning from patient records, as well as seamless incorporation of expert, domain knowledge into the learning process. The methods learn low-dimensional representations of diseases and genes in a common embedding space, setting the foundation for disease-disease and disease-gene relationship discovery through trivial K-nearest neighbor searches in the new vector space. The experiments on large-scale EHR data demonstrate that the proposed approaches significantly outperform the existing state-of-the-art methods on im-

\(^3\)https://www.genecards.org/Search/Keyword?queryString=SH3RF1, acc. June 2018
\(^4\)https://www.genecards.org/Search/Keyword?queryString=SPAG16, acc. June 2018
portant tasks of phenotyping and gene discovery in the emerging area of computational phenotyping. Benefits of the approaches will advance clinical research and practice by accelerating our understanding of disease and gene associations.

3.5 Disease Types Discovery from a Large Database of Inpatient Records: A Sepsis Study

Many diseases have heterogeneous nature, for instance sepsis, a systemic and progressive inflammation that can be caused by many factors, and can have multiple manifestations including on different human organs. Understanding such heterogeneity of the disease can help in addressing many important issues regarding sepsis, including early diagnosis and treatment, which is of huge importance as sepsis is one of the main causes of in-hospital deaths in the United States. This study analyzes state of the art embedding models that have had huge success in many fields including disease embeddings from EHR databases as shown in Section 3.4. Thus, we have a particular interest to learning multi-type representation of heterogeneous diseases, that leads to more homogeneous groups of phenotypes (Gligorijevic et al., 2016a). Our results show evidence that such representations have phenotypes of higher quality and also provide benefit when predicting mortality of inpatient visit.

3.5.1 Introduction

An emphasis has recently been given to the effective mining of big EHR databases in order to obtain actionable insights for improving healthcare, a concept often termed “data driven healthcare” (Dat, 2014; Madsen, 2014). However, mining such data comes with challenges as it is often sparse, heterogeneous, noisy and biased due to different hospital and insurance company polices and non-standardized physician practices (Hripcsak and Albers, 2013).

Computational phenotyping, described in Section 3.4.1, aims at utilizing available
EHR data to learn high quality representation that can potentially be used for more effective computational and statistical modeling aimed towards improved disease characterization and intervention. In clinical practice, these studies can allow medical practitioners to obtain novel insights in the patients’ conditions and therapeutic processes thus improving treatment. Such discoveries are especially important for complex diseases such as sepsis (Stojkovic et al., 2016b), a potentially life-threatening complication of pathogen infection where mechanisms are not yet fully understood and where contemporary treatments are far from optimal (Dellinger et al., 2013).

Causes of health and wellness span multiple body systems and physiologic processes, thus the complexity of the phenotyping process is increased. This creates a nonlinear relationship among observed measurements, making the process of learning robust representations of human physiology challenging (Che et al., 2015). In this study we aim to automatically detect such segments of diseases from large EHR databases by exploiting disease comorbidity information contained in patient discharge records. In the past, many researchers aimed to discover segments of diseases in order to better understand more homogeneous subsets (Bauer-Mehren et al., 2011; He et al., 2011; Kikuchi et al., 2013). Previously studied disease segmentation approaches, often consisted of observing metabolic, genetic or proteogenomic interactions, thus differing from the purely EHR based approach proposed in this study.

To provide evidence of benefits from using the proposed disease multi modal embedding approach, we conducted a case study discovering segments for all sepsis related diagnoses. Sepsis is a potentially life-threatening complication of pathogen infection that triggers the systemic inflammatory response (Dellinger et al., 2013; Stojkovic et al., 2016b). Such systemic and progressive inflammation can lead to multiple organ dysfunction syndrome and even death (Russel, 2008). It can occur due to many reasons (i.e. infection from bacteria, fungi, viruses, or other organisms on different organs, etc.) and it has a wide range of symptoms. Hence, sepsis is not a yet fully understood condition while
treatments are still far from optimal; it is often diagnosed too late, which can result in a mortality rate as high as 30–50% in the case of septic shock (Thiel et al., 2010; Dellinger et al., 2013). Therefore, complicated coding techniques are applied by the physicians to discriminate between different sepsis cases while documenting patients’ discharge records (Wiedemann, 2007). It is a disease that afflicts a large population (Anonymous, 2012) and was the largest cause of death in the state of California from 2003 to 2011 (Figure 3.6). Furthermore, sepsis is recognized as one of the main causes of in-hospital deaths in the United States (Liu et al., 2014), with more than 750,000 cases annually (Zuev et al., 2006), and it contributes to 1 in every 2 to 3 deaths (Levy et al., 2003). In addition to overwhelming presence of the sepsis, hospital costs of over $20 billion in 2011 in the United States (Torio and Andrews, 2013) provide a huge motivation for research in fields of understanding, diagnosing and treating such condition, as the incidence of sepsis is rising (Martin, 2012). In this study, we aim to exploit such information recorded in a large EHR database in order to automatically build multi modal representations of sepsis diagnoses with a purpose to propose a system for improving sepsis diagnosis and potentially aid in early predicting outcomes.

The proposed novel, multi modal neural embedding model (Huang et al., 2012; Nee-lakantan et al., 2015) is adapted for use in medical records for disease embeddings, following from their major success in the field of Natural Language Processing (NLP) and other fields (Mikolov et al., 2013b,a; Djuric et al., 2015b). The goal of this study is to learn low dimensional distributed representations of diseases by utilizing context from inpatient diagnoses and learn multiple embeddings for diseases of interest that would differ in the embedded space according to differences in contexts.

3.5.2 Sepsis inpatient discharge records dataset

We sample only discharge records containing one of the sepsis related codes from the SID CA database. Among the conditions considered we have included Systemic inflam-
Figure 3.6: Admission (blue) and mortality (red) trends of sepsis diagnoses in California for the period 2003-2009

Inflammatory response syndrome (SIRS), sepsis and septicemia (names and ICD-9 codes given in Table 3.6). SIRS is defined as a clinical response to an insult, infection, or trauma that includes a systemic inflammation as well as elevated or reduced temperature, rapid hearth rate, rapid respiration, and elevated blood white blood count. Sepsis is additionally defined as SIRS due to infection without organ dysfunction, while severe sepsis is defined as SIRS due to infection with organ dysfunction. Please note that terms septicemia and sepsis are often used interchangeably, but are not considered synonyms in the ICD-9 coding. Septic shock is defined as a systematic disease associated with the presence of a pathogenic microorganisms within the blood stream only. The selected sepsis targeted subset of the entire SID CA database, described in Section 3.3, constitutes 1,127,114 discharge records, comprising 3.14% of total discharge records over the state of California from 2003 to 2011.

The process of coding sepsis in the EHR databases is a tedious work, even under the most obvious circumstances, and requires proper application of the AHA Coding Clinic guidelines (Association et al., 2008) and the Official Guidelines for Coding and Reporting

79
Table 3.6: ICD-9 codes related to septic inpatients

<table>
<thead>
<tr>
<th>Diagnosis code</th>
<th>Diagnosis name</th>
</tr>
</thead>
<tbody>
<tr>
<td>995.90</td>
<td>Systemic inflammatory response syndrome, unspecified</td>
</tr>
<tr>
<td>995.91</td>
<td>Sepsis</td>
</tr>
<tr>
<td>995.92</td>
<td>Severe sepsis</td>
</tr>
<tr>
<td>995.93</td>
<td>Systemic inflammatory response syndrome due to noninfectious process without acute organ dysfunction</td>
</tr>
<tr>
<td>995.94</td>
<td>Systemic inflammatory response syndrome due to noninfectious process with acute organ dysfunction</td>
</tr>
<tr>
<td>785.52</td>
<td>Septic shock</td>
</tr>
<tr>
<td>038.9</td>
<td>Unspecified septicemia</td>
</tr>
</tbody>
</table>

For inpatient care (for Medicare et al., 2011), as well as well documented physician notes (Wiedemann, 2007). SIRS is relatively easy to diagnose as there are strict physiological parameters that need to be satisfied. The EHR data records are represented by at least two codes, one for the underlying cause of infection (i.e., 038.xx, . . .) and another for the sepsis subcategory (995.9x). Severe sepsis requires a minimum of three codes: a code for systemic infection (i.e., 038.xx, . . .), the code 995.92 and the code for the associated organ failure. Septic shock is defined as severe sepsis with circulatory system failure, and in coding it only differs from severe sepsis in the second code where 995.92 is changed to 785.52. Finally, unspecified septicemia, code 038.9, is used when there is not enough information in the doctors’ notes and other diagnoses do not show a clear sign of what the state of the patient inflammation was (Wiedemann, 2007). As can be seen from Figure 3.7, the SIRS conditions are the least prevalent, including virtually no cases where codes 995.90, 995.93 and 995.94 were used. On the other hand, the difficulty in properly diagnosing septic patients as described above is manifested with the most dominant diagnosis being Unspecified septicemia which was registered in around 36% of patient that were septic.

The discharge record containing a sepsis-related diagnosis is expected to have more than 2 diagnoses related to sepsis. Moreover, in the selected subset of the SID CA
database, 16 diagnoses are observed on average. Thus, the context of ones’ inpatient stay includes other conditions observed in the record, which may provide additional insight in analyzing septic patient cases.

3.5.3 Methodology

Motivated by the results obtained in Section 3.4, we build on the disease2vec approach and extend it to learn type specific embeddings of heterogeneous diseases.

Problem definition

We are given a set \( \mathcal{P} \) of patient discharge records, where patient’s discharge record \( p_i = (d_{i1}, \ldots, d_{iM_i}) \in \mathcal{P} \) is defined as a sequence of \( M_i \) diagnosed diseases \( d_i \in \mathcal{D} \) at the end of the hospital stay. The objective is to find the \( D \)-dimensional real-valued representation \( v_d \in \mathcal{R}^D \) of each disease \( d \) such that diseases with similar phenotypes have similar representation.
Low-dimensional disease embeddings

In Section 3.4 we only defined SkipGram architecture of neural embedding for disease2vec model, while here we also explore continuous bag of words CBOW architecture described below. Graphical representations of both architectures for disease2vec model, for an example of sepsis inpatient diagnosis is given in Figure 3.8.

**CBOW disease2vec representation** In a continuous bag of words (CBOW) approach disease representations are learned by maximizing the objective function $\mathcal{L}$ over the entire set $\mathcal{P}$ of records, as

$$\mathcal{L} = \sum_{p \in \mathcal{P}} \sum_{d_m \in p} \log \Pr(d_m|d_{m-b}, d_{m-1}, \ldots, d_{m+1}, d_{m+b}).$$

(3.8)

Probability $\Pr(d_m|d_{m-b} : d_{m+b})$ of observing a center disease $d_m$ given its disease context $d_{m-b} : d_{m+b}$ is defined using the soft-max function,

$$\Pr(d_m|d_{m-b} : d_{m+b}) = \frac{\exp(\mathbf{v}_d^T \mathbf{v}_d')}{\sum_{d=1}^{D} \exp(\mathbf{v}_d^T \mathbf{v}_d')},$$

(3.9)
where $v_d$ and $v'_d$ are the input and output vector representations of $D$-dimensional disease $d$, and $2b$ is the length of the context for disease records. $\bar{v}$ is obtained by averaging input vector representation of all diseases in observed context,

$$\bar{v} = \frac{1}{T_c} \sum_{c=1}^{T_c} v_{dc}$$  \hspace{1cm} (3.10)

As illustrated in Figure 3.8a and equation 3.9, CBOW disease2vec representation uses surrounding $T_c = 2b$ diseases $d_{m-b} : d_{m+b}$ to predict central disease $d_m$ for each disease $d_m$ in the discharge record. Thus, diseases that often co-occur and diseases with similar contexts (i.e., with similar neighboring diseases) will have similar representations.

**SkipGram disease2vec representation**  In SkipGram-based representation, central disease $d_m$ is used to predict $b$ diseases that occur before and $b$ diseases that occur after it in the discharge record, as illustrated in Figure 3.8b and equation 3.3. The SkipGram model introduces an additional assumption that neighboring diseases are independent of each other. Disease representations are learned by maximizing the objective function $\mathcal{L}$ over the entire set $\mathcal{P}$ of records, as

$$\mathcal{L} = \sum_{p \in \mathcal{P}} \sum_{d_m \in p} \sum_{-b \leq i \leq b, i \neq 0} \log \Pr(d_{m+i}|d_m).$$ \hspace{1cm} (3.11)

The probability $\Pr(d_{m+i}|d_m)$ of observing a “neighboring” disease $d_{m+i}$ given disease $d_m$ is defined using the soft-max function,

$$\Pr(d_{m+i}|d_m) = \frac{\exp(v_{d_m}^T v'_{d_{m+i}})}{\sum_{d=1}^{D} \exp(v_{d_m}^T v'_d)},$$ \hspace{1cm} (3.12)

where $v_d$ and $v'_d$ are the input and output vector representations of disease $d$ with dimensionality $D$, and $2b$ defines the length of the context for disease records.
Multi-type disease2vec disease representation

A major limitation of previously described models is that they assume a single vector representation for each disease. Such a disease representation is aimed to capture global trends in the discharge records, but it will not be able to represent the heterogeneity of each disease appropriately. For example, sepsis is a heterogeneous disease triggered by pneumonia, abdominal infection, kidney infection, bloodstream infection or other causes, and manifested on multiple organs, with different severity. Multi-type representations for such a complex disease can result in a more appropriate low-dimensional representation.

The multi-prototype approach for vector space models, which uses multiple representations to capture different senses and usages of a word is successfully used in the field of NLP (Reisinger and Mooney, 2010) and a related approach is also applied to neural language models (Huang et al., 2012). Here we also extend disease2vec models to a model using multiple types which we call t-CBOW and t-SkipGram. In particular, we represent each discharge record by a sum of vectors of diagnoses found in that record. This global context representation dataset of inpatient visits is then clustered using K-means algorithm (Reisinger and Mooney, 2010) to obtain types of patient records (Dhillon and Modha, 2001) that contained sepsis as a diagnosis. Finally, each sepsis occurrence in the discharge data is re-labeled to its associated cluster. Due to known heterogeneity of the discharge records data sepsis types are obtained by clustering inpatient visit representation rather than observed disease contexts as in (Huang et al., 2012). New vectors of sepsis types are initialized as its global vector, and updated on the dataset such that the original sepsis disease spans a larger portion of the embedded space (via its types) thus capturing novel, previously undiscovered relationships.

This approach works globally for the entire dataset, in the form of a pipeline. However, it is possible to make disease2vec automatically model multiple types for each disease, specifically SkipGram, by locally discriminating contexts of each disease by either
the MaxOut method or the K-means model and then deciding on the type vector update (Neelakantan et al., 2015). Such an approach is described in the following section.

**Multi-sense SkipGram disease representation**

This model, based on *multi-sense SkipGram* (MSSG) (Neelakantan et al., 2015) (Figure 3.9), is capable of learning multiple types for each disease by locally discriminating contexts of each disease by either the MaxOut method or the K-means model. It performs multi-modal learning by clustering the embeddings of context around each disease. For each disease, clusters are maintained, and once the cluster is predicted the disease context representation for a disease type is updated. The difference between this and a multi-type disease2vec approach is that local contexts are clustered to decide the type of the disease and that the entire process is performed jointly by predicting the sense of the disease using the current parameter estimates. In the MSSG model, a global vector $v_g(d)$ is assigned to each disease $d \in D$ and each type of the disease has a separate embedding $v_s(d, k)$ ($k = 1, 2, \ldots, K$), as well as a context cluster with center $\mu(d, k)$ ($k = 1, 2, \ldots, K$).

Clustering is performed in the following manner. First, for each disease $d$, a context vector is obtained by $v_{\text{context}}(c_d) = \frac{1}{T_c} \sum_{c=1}^{T_c} v_g(d_c)$, where $c_d$ is context of disease $d$, and $T_c$ is the size of the context. For context representation global vectors $v_g$ are used rather
than type-specific vectors to avoid additional computational complexity. Context representation $v_{\text{context}}(c_d)$ is then used to predict the type of the disease $d$. In previous work (Neelakantan et al., 2015), two approaches are discussed. Type of the disease $s_k$ can be determined either by the MaxOut method:

$$s_k = \arg\max_{k=1,2,...,K} (v_{s}(d,k)^\top v_{\text{context}}(c_d)), \quad (3.13)$$

or by K-mean clustering:

$$s_k = \arg\max_{k=1,2,...,K} \text{sim}(\mu(d,k), v_{\text{context}}(c_d)). \quad (3.14)$$

Here the cluster center $\mu(d,k)$ is the average of all the context representation observed that belong to that cluster. For $\text{sim}$, cosine similarity is used in our experiments.

Finally, the objective function is obtained as in the SkipGram model (Eq. 3.11), with addition that the softmax function (Eq. 3.12) is conditioned on the cluster in which disease $d$ belongs.

### 3.5.4 Experimental evaluations

In this section we describe experimental setups and the results obtained from such experiments. Mortality prediction results on sepsis-diagnosed patients using both type-specific and global embedding models are shown and an analysis of discovered types of sepsis related diagnoses is conducted.

All models were trained on 1, 127, 114 sepsis diagnosed discharge records using a machine with 32GB of RAM memory and 4 cores. Diseases were mapped into $D = 200$ dimensional space (which was decided based on model complexity, and resulting model performance). The context parameter $b$ was varied in a set {2,4,16}, where 2 and 4 are determined with respect to coding patterns described in Section 3.5.2, and 16 was chosen.
to observe larger heterogeneous context as 16 was the average number of diagnoses in the dataset. We used 25 negative samples in each vector update for negative sampling, following a previously proposed approach for efficient learning (Mikolov et al., 2013a). The number of types $K$ is considered in the range 1 to 15, where 15 is the number of reported different underlying infections causing sepsis according to ICD-9 coding for 038.xx diagnoses\(^5\). The results reported in this section are obtained for $K = 5$ types.

3.5.5 Mortality prediction

In this section we evaluate the representational power of the discovered disease types. Feature vectors are learned in the embedded space for each disease and can be used for predictive tasks as such. Specifically, we used discovered sepsis types to predict patient mortality and compared benefits of such a type-specific approach versus predicting mortality based on global features of sepsis in the embedded representation. The hypothesis evaluated in this experiment was that the multi-type sepsis vectors carry more information about mortality (some causes/effects can be more fatal than others) than the ones learned via global embedding models. We compared embeddings learned by four models from the family of type-specific embeddings ($t$-CBOW, $t$-SkipGram, MSSG MaxOut and MSSG K-means) to two global embedding models (learned by CBOW and SkipGram).

Features learned by those 6 embedding models were the input to Logistic Regression algorithm used for mortality prediction (similar results were obtained by running SVM and neural network based classifiers). The model is trained on different subsets using 10\% to 90\% of data obtained as a balanced random sample and 10-fold validation. Learned models were evaluated on the remaining EHR data. The results show stable performance of both accuracy and F1 measure, with respect to the entire range of training data sizes. Therefore, Table 3.7 aggregates the evaluation results (accuracy and F-1 measure) of 6 models on 90 experiments from 10 validations on 9 different training-test sizes. Addition-

\(^5\) http://www.icd9data.com/2013/Volume1/001-139/030-041/038/, acc. June 2018
Table 3.7: Accuracy and F-1 measure aggregated over 90 experiments for Logistic Regression model used on features learned by 6 embedding models: 4 type-specific and 2 global, for three values of hyperparameter $b$. The best results are bolded.

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th></th>
<th>F1-measure</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$b=2$</td>
<td>$b=4$</td>
<td>$b=16$</td>
<td>$b=2$</td>
</tr>
<tr>
<td>t-CBOW</td>
<td>77.2%</td>
<td>77.2%</td>
<td>76.1%</td>
<td>77.7%</td>
</tr>
<tr>
<td>t-SkipGram</td>
<td>76.6%</td>
<td>76.9%</td>
<td>74.7%</td>
<td>77.1%</td>
</tr>
<tr>
<td>MSSG K-kmeans</td>
<td>67.9%</td>
<td>68.0%</td>
<td>69.3%</td>
<td>69.0%</td>
</tr>
<tr>
<td>MSSG MaxOut</td>
<td>67.9%</td>
<td>68.0%</td>
<td>69.1%</td>
<td>69.0%</td>
</tr>
<tr>
<td>CBOW</td>
<td>56.0%</td>
<td>56.1%</td>
<td>67.1%</td>
<td>58.3%</td>
</tr>
<tr>
<td>SkipGram</td>
<td>55.0%</td>
<td>55.4%</td>
<td>67.1%</td>
<td>57.0%</td>
</tr>
</tbody>
</table>

ally, we have examined the influence of the context window size defined by parameter $b$ chosen from a set \{2,4,16\} to overall predictive accuracy.

All type-specific embedding based sepsis mortality models were more accurate than the global models, where the best performing algorithm was the proposed $t$-CBOW model described in Section 3.5.3. The results were stable with respect to the parameter $b$ when shorter context is used ($b=2$ or 4), while larger context ($b=15$) resulted in slightly decreased accuracy for $t$-CBOW and $t$-SkipGram models. The proposed multi-type approaches were more robust to the context window size. Larger context allows partial learning of broader concepts in a single type, which is why CBOW and SkipGram were more accurate with larger window size $b$. As the accuracy was fairly stable for multi-type models, but highly increased for global models for $b=16$ (when compared to lower values of this parameter), phenotyping results are shown for the embedding with this parameter in Table 3.8 – 3.12. Finally, this experiment provides evidence that discovered sepsis segmentation is clinically relevant.

Disease types and nearest neighbors

In this section we discuss sepsis disease phenotypes discovered by both global and type-specific embeddings. Here, phenotypes are defined as query disease’s nearest neighbors
in the embedded space. For the type-specific models, we discuss phenotypes found by the \textit{t-CBOW} model, as it was the best competing model for mortality prediction, and the \textit{CBOW} model for the global embeddings, as there was no significant difference from the \textit{SkipGram} model on the same task. Parameter $b$ is fixed to be 16 in this section for both models, as results on mortality prediction gave the most balanced accuracy performance over all models examined.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.10.png}
\caption{In the embedded space (here displayed 2D reduced space) each of seven sepsis-related ICD-9 diagnoses is partitioned to five types marked in different colors.}
\end{figure}

We show the 5 embedded disease-types for each of the 7 sepsis diagnoses in Figure 3.10. Discovered five disease types emancipated cluster-like groupings in the new embedded space. Furthermore, we observe that all diagnoses in a same type share similar phenotype properties. Concrete findings will be discussed in more details below. Another interesting finding is the \textit{outlier type} (upper right corner of Figure 3.10). The observed \textit{outlier type} has low prevalence, with less than a thousand cases in our dataset (or less than 1\% of the discharge records). As such, it will be removed from further discussion, even though it forms the purest cluster, given that the main focus of this study are prevalent phenotypes, while analysis of outliers will be left for future work.
Additionally, we have observed that SIRS conditions also have much lower prevalence than sepsis (less than 1% as shown in Figure 3.7). Thus, the analysis reported in this paper is focused to segmentation to four types for each of four sepsis diagnoses as shown in Figure 3.11. Analysis of disease type records shows that each of the remaining four discovered types of diseases occur in at least 10% of discharge records (Figure 3.11), and therefore, are well represented in the dataset.

In Table 3.9, we list five nearest non-sepsis diseases to the sepsis diagnosis 995.91 in the embedded space representation learned by the global embedding model. *Sepsis global* phenotype shows heterogeneous properties where most similar diagnoses are infections.
Table 3.8: **Sepsis** (995.91 code) vector and its 5 nearest neighbors in the embedded global disease space vs 4 type-specific embedded disease space.

<table>
<thead>
<tr>
<th>Rank for (995.91)</th>
<th>global</th>
<th>type 1</th>
<th>type 2</th>
<th>type 3</th>
<th>type 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed fracture of lower and of forearm unspec.</td>
<td>1</td>
<td>5990</td>
<td>9118</td>
<td>604</td>
<td>4965</td>
</tr>
<tr>
<td>Acute upper respiratory infections of unspec. site</td>
<td>2</td>
<td>637</td>
<td>6746</td>
<td>1993</td>
<td>881</td>
</tr>
<tr>
<td>Urinary tract infection site not specified</td>
<td>3</td>
<td>5845</td>
<td>8185</td>
<td>91</td>
<td>274</td>
</tr>
<tr>
<td>Leukocytosis unspec.</td>
<td>4</td>
<td>4643</td>
<td>4649</td>
<td>1408</td>
<td>761</td>
</tr>
<tr>
<td>Legaly induced abortion with other spec. complications</td>
<td>5</td>
<td>9230</td>
<td>350</td>
<td>8770</td>
<td>2797</td>
</tr>
</tbody>
</table>

on different parts of organs, but also abortion or fracture related diagnosis, which are known as possible sepsis causes or effects (Kylänpää-Bäck et al., 1992; Nguyen et al., 2006). For each of the five most similar diseases in the **Sepsis global** phenotype, their rankings by the type-specific models are provided in columns **type 1**- **type 4** (for each of the types). Globally relevant diseases are not particularly close in the embedded space for most homogeneous types of sepsis, which can also be concluded from Figure 3.10. Note that ICD-9 codes provide disease coding on a very fine scale. For instance, the same condition can be present in multiple locations of an organ, and there are multiple codes for such a disease. Fine scale disease coding is the cause of low ranks of globally relevant diseases in type-specific phenotypes as other similar but type-specific conditions are ranked higher. Additionally, for each of the types, there is one globally relevant disease that is higher ranked in that type than other diseases in the same type. For example, in sepsis disease **type 4**, which represents urinary related phenotype, the closest condition is the **urinary tract infection**, while the other conditions are at least three times lower ranked.

Four discovered types of Sepsis (diagnosis 995.91) will be referred to as **Sepsis type 1** to **Sepsis type 4** and will be labeled as 995.91₁ to 995.91₄. For each of the types, five most similar diseases in the embedded space representation were listed in Table 3.9 as obtained
Table 3.9: Five segments of **Sepsis** (995.91 code) and their 5 nearest neighbors in the embedded disease space

<table>
<thead>
<tr>
<th>5 most related diagnoses in the embedded space to Sepsis (995.91)</th>
<th>Rank by types</th>
<th>Global rank</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sepsis type 1 (995.91₁) [36.36%]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transient arthropathy shoulder region</td>
<td>1</td>
<td>1104</td>
</tr>
<tr>
<td>Tension headache</td>
<td>2</td>
<td>525</td>
</tr>
<tr>
<td>Unspec. abortion</td>
<td>3</td>
<td>786</td>
</tr>
<tr>
<td>Unspec. abortion complicated by damage to pelvic organs</td>
<td>4</td>
<td>2919</td>
</tr>
<tr>
<td>Paratyphoid fever A</td>
<td>5</td>
<td>566</td>
</tr>
<tr>
<td><strong>Sepsis type 2 (995.91₂) [33.41%]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variations in hair color</td>
<td>1</td>
<td>1410</td>
</tr>
<tr>
<td>Other persistent mental disorders</td>
<td>2</td>
<td>933</td>
</tr>
<tr>
<td>Paralysis agitans</td>
<td>3</td>
<td>913</td>
</tr>
<tr>
<td>Senile dementia uncomplicated</td>
<td>4</td>
<td>1525</td>
</tr>
<tr>
<td>Unspec. senile psychotic condition</td>
<td>5</td>
<td>4091</td>
</tr>
<tr>
<td><strong>Sepsis type 3 (995.91₃) [16.11%]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Open skull fracture with cerebral laceration and contusion</td>
<td>1</td>
<td>3684</td>
</tr>
<tr>
<td>Nervous system complications from surg. implanted device</td>
<td>2</td>
<td>7486</td>
</tr>
<tr>
<td>Inclusion conjunctivitis</td>
<td>3</td>
<td>5704</td>
</tr>
<tr>
<td>Malignant neoplasm of other and unspec. testis</td>
<td>4</td>
<td>8215</td>
</tr>
<tr>
<td>Anemia of mother unspecified</td>
<td>5</td>
<td>8164</td>
</tr>
<tr>
<td><strong>Sepsis type 4 (995.91₄) [14.09%]</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hypertensive chronic kidney disease (stage V)</td>
<td>1</td>
<td>7013</td>
</tr>
<tr>
<td>End stage renal disease</td>
<td>2</td>
<td>4992</td>
</tr>
<tr>
<td>Infection and inflammatory reaction due to oth. vascular device</td>
<td>3</td>
<td>1418</td>
</tr>
<tr>
<td>Complic. due to renal dialysis device implant and graft</td>
<td>4</td>
<td>7142</td>
</tr>
<tr>
<td>Hypertensive heart and chronic kidney disease</td>
<td>5</td>
<td>5254</td>
</tr>
</tbody>
</table>

Based on the $t$-CBOW model for 995.91. The global rank for each of the listed diseases is also shown as obtained by the global CBOW embedding model.

As compared to the global phenotypes, sepsis type-specific phenotypes are more homogeneous. For example, sepsis in pregnant and postpartum women can develop as the result of many complications, such as miscarriages (spontaneous abortions) or induced abortions, prolonged or obstructed labor, ruptured membranes, cesarean sections, infec-
tion following a vaginal delivery, etc. (Fernandez-Perez et al., 2005; Bauer et al., 2013). Some of these causes related to delivery (i.e., prolonged labor or ruptured membranes) are found in Sepsis outlier type, while causes related to abortions are found in Sepsis type 1. Both types have these causes ranked highly (they are close to sepsis vector in the embedded space) by the t-CBOW model, whereas the global ranking model assigns low ranks (e.g. 1963, 2919, 8583) which shows the over-representational ability of the proposed model over the global embedding.

Sepsis can cause a lot of damage in a person that is affected by this disease and its treatment can also leave different consequences. The kidneys are often among the first organs to be affected by sepsis and published studies report that between 32% and 48% of acute kidney injury cases were caused by sepsis (Waikar et al., 2008). In the case of patients with sepsis and a urinary tract infection, physicians often use the term urosepsis (Wiedemann, 2007). Therefore, it is not surprising that Sepsis type 4 is very related to kidney diseases (not just for sepsis, but also for the other sepsis disease shown in Tables 3.10 - 3.12).

Another type of sepsis consequences are mental disorders, that are covered in Sepsis type 2. It is reported that 17% of sepsis elderly survivors developed dementia and around 40% experienced nervous system damage and could not walk without assistance in the years after (Iwashyna et al., 2010). These diseases are highly ranked by t-CBOW in Sepsis type 2.

Sepsis type 3 covers diseases related to serious brain tissue injuries and nervous system complications from surgically implanted devices, which both can lead to septic inflammation (Okapa et al., 2010) and reproductive system related causations of sepsis (Sinha and Otify, 2012). Since, Sepsis type 1 covers a large fraction of impatient record cases (36.6%), it is expected that this phenotype is the most heterogeneous among all. Therefore, in addition to abortion cases, we observe other possible causes and effects of this disease.

Discovered phenotypes of global and type-specific embeddings of Severe sepsis, Septic
Table 3.10: Four segments of Severe Sepsis (995.92 code) and their 5 nearest neighbors in the embedded disease space

<table>
<thead>
<tr>
<th>5 most related diagnoses in the embedded space to Severe Sepsis (995.92)</th>
<th>Rank in type</th>
<th>Global rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severe sepsis type 1 (995.92₁) [17.49%]</td>
<td>1</td>
<td>1650</td>
</tr>
<tr>
<td>Hemicrania continua</td>
<td>2</td>
<td>1604</td>
</tr>
<tr>
<td>Chronic Eustachian salpingitis</td>
<td>3</td>
<td>3562</td>
</tr>
<tr>
<td>Other nongonococcal urethritis unspecified</td>
<td>4</td>
<td>9435</td>
</tr>
<tr>
<td>Other manifestations of yaws</td>
<td>5</td>
<td>466</td>
</tr>
<tr>
<td>Acute pyelonephritis without lesion of renal medullary necrosis</td>
<td>6</td>
<td>9546</td>
</tr>
<tr>
<td>Severe sepsis type 2 (995.92₂) [41.64%]</td>
<td>1</td>
<td>9054</td>
</tr>
<tr>
<td>Chondrocalcinosis due to pyrophosphate crystals upper arm</td>
<td>2</td>
<td>7550</td>
</tr>
<tr>
<td>Meningitis in sarcoidosis</td>
<td>3</td>
<td>933</td>
</tr>
<tr>
<td>Other persistent mental disorders due to conditions classified</td>
<td>4</td>
<td>6337</td>
</tr>
<tr>
<td>Hyperosmolality and-or hypernatremia</td>
<td>5</td>
<td>913</td>
</tr>
<tr>
<td>Paralysis agitans</td>
<td>6</td>
<td>913</td>
</tr>
<tr>
<td>Severe sepsis type 3 (995.92₃) [29.54%]</td>
<td>1</td>
<td>8730</td>
</tr>
<tr>
<td>Burn involving 50-59 % of body surface w 3. degree burn 40-49%</td>
<td>2</td>
<td>8584</td>
</tr>
<tr>
<td>Letterer-siwe di. unspec. site extranodal and solid organ sites</td>
<td>3</td>
<td>9546</td>
</tr>
<tr>
<td>Pneumococcal peritonitis</td>
<td>4</td>
<td>9352</td>
</tr>
<tr>
<td>Defibrination syndrome</td>
<td>5</td>
<td>6584</td>
</tr>
<tr>
<td>Tuberculosis of intestines peritoneum and mes. glands tubercle bacilli</td>
<td>6</td>
<td>913</td>
</tr>
<tr>
<td>Severe sepsis type 4 (995.92₄) [11.32%]</td>
<td>1</td>
<td>7013</td>
</tr>
<tr>
<td>Hypertensive chronic kidney disease ( V or end stage renal dis.)</td>
<td>2</td>
<td>7728</td>
</tr>
<tr>
<td>Nephrotic syndrome in diseases classified elsewhere</td>
<td>3</td>
<td>4992</td>
</tr>
<tr>
<td>End stage renal disease</td>
<td>4</td>
<td>5307</td>
</tr>
<tr>
<td>Diabetes with renal manifestations type II ...</td>
<td>5</td>
<td>7142</td>
</tr>
</tbody>
</table>

shock, and Septicemia diagnoses are presented in Tables 3.10, 3.11, 3.12, respectively. We observe that disease types show similar traits, as anticipated from Figure 3.10. The phenotypes discovered for the three diseases are consistent with the sepsis types: type 4 sepsis diseases are related to kidney and urinal tract problems, type 2 sepsis diseases are related to nervous system inflammations, while type 1 and type 3 sepsis diseases are related to external irritations such as burns, fractures and different inflammations. As expected,
Table 3.11: Four segments of **Septic Shock** (785.52 code) and their 5 nearest neighbors in the embedded disease space

<table>
<thead>
<tr>
<th>5 most related diagnoses in the embedded space to Septic Shock (785.52₁)</th>
<th>Rank in type</th>
<th>Global rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Septic shock type 1 (785.52₁) [15.65%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chronic Eustachian salpingitis</td>
<td>1</td>
<td>1604</td>
</tr>
<tr>
<td>Other nongonococcal urethritis unspecified</td>
<td>2</td>
<td>3562</td>
</tr>
<tr>
<td>Cocaine dependence episodic</td>
<td>3</td>
<td>2459</td>
</tr>
<tr>
<td>Encounter for removal of intrauterine contraceptive device</td>
<td>4</td>
<td>7827</td>
</tr>
<tr>
<td>Inconclusive mammogram</td>
<td>5</td>
<td>3110</td>
</tr>
<tr>
<td>Septic shock type 2 (785.52₂) [39.76%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chondrocalcinosis due to pyrophosphate crystals upper arm</td>
<td>1</td>
<td>9054</td>
</tr>
<tr>
<td>Meningitis in sarcoidosis</td>
<td>2</td>
<td>7550</td>
</tr>
<tr>
<td>Hyperosmolality and-or hypernatremia</td>
<td>3</td>
<td>6337</td>
</tr>
<tr>
<td>Closed lateral dislocation of elbow</td>
<td>4</td>
<td>5474</td>
</tr>
<tr>
<td>Paralysis agitans</td>
<td>5</td>
<td>913</td>
</tr>
<tr>
<td>Septic shock type 3 (785.52₃) [33.98%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Defibrination syndrome</td>
<td>1</td>
<td>9352</td>
</tr>
<tr>
<td>Pneumococcal peritonitis</td>
<td>2</td>
<td>9546</td>
</tr>
<tr>
<td>Letterer-siwe disease unspec. site extranodal and solid organ sites</td>
<td>3</td>
<td>8584</td>
</tr>
<tr>
<td>Burn involving 50-59 % of body surface w 3. degree burn 40-49%</td>
<td>4</td>
<td>8730</td>
</tr>
<tr>
<td>Acute and subacute necrosis of liver</td>
<td>5</td>
<td>9741</td>
</tr>
<tr>
<td>Septic shock type 4 (785.52₄) [10.61%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hypertensive chronic kidney disease (V or end stage renal dis.)</td>
<td>1</td>
<td>7013</td>
</tr>
<tr>
<td>End stage renal disease</td>
<td>2</td>
<td>4992</td>
</tr>
<tr>
<td>Other complications due to renal dialysis device implant and graft</td>
<td>3</td>
<td>7142</td>
</tr>
<tr>
<td>Nephrotic syndrome in diseases classified elsewhere</td>
<td>4</td>
<td>7728</td>
</tr>
<tr>
<td>Hypertensive heart and chronic kidney disease w. heart failure and chronic kidney disease stage V or end stage</td>
<td>5</td>
<td>8387</td>
</tr>
</tbody>
</table>

Severe sepsis and septic shock phenotypes share 65% of the closest diseases, as they are considered as the same condition with septic shock being a severe sepsis with circulatory system failure.
Table 3.12: Four segments of Septicemia (038.9 code) and their 5 nearest neighbors in the embedded disease space

<table>
<thead>
<tr>
<th>5 most related diagnoses in the embedded space to Septicemia (038.9)</th>
<th>Rank in type</th>
<th>Global rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Septicemia type 1 (038.9) [25.53%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Basal cell carcinoma of scalp and skin of neck</td>
<td>1</td>
<td>2172</td>
</tr>
<tr>
<td>Inappropriate diet and eating habits</td>
<td>2</td>
<td>1703</td>
</tr>
<tr>
<td>Screening for other disorders of blood and blood-forming organs</td>
<td>3</td>
<td>9037</td>
</tr>
<tr>
<td>Impairment of auditory discrimination</td>
<td>4</td>
<td>778</td>
</tr>
<tr>
<td>Other arthropod infestation</td>
<td>5</td>
<td>451</td>
</tr>
<tr>
<td>Septicemia type 1 (038.9) [41.79%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Loose body in joint other specified sites</td>
<td>1</td>
<td>1337</td>
</tr>
<tr>
<td>Other circadian rhythm sleep disorder</td>
<td>2</td>
<td>1901</td>
</tr>
<tr>
<td>Meningitis in sarcoidosis</td>
<td>3</td>
<td>7550</td>
</tr>
<tr>
<td>Other persistent mental disorders due to conditions classified elsewhere</td>
<td>4</td>
<td>933</td>
</tr>
<tr>
<td>Variations in hair color</td>
<td>5</td>
<td>1410</td>
</tr>
<tr>
<td>Septicemia type 1 (038.9) [21.92%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Burn involving 50-59% of body surface w 3. degree burn 40-49%</td>
<td>1</td>
<td>8730</td>
</tr>
<tr>
<td>Congenital anomalies of corneal size and shape</td>
<td>2</td>
<td>6577</td>
</tr>
<tr>
<td>Open fracture of mandible alveolar border of body</td>
<td>3</td>
<td>9357</td>
</tr>
<tr>
<td>Open skull fracture, cerebral laceration, contusion, loss of consciousness</td>
<td>4</td>
<td>3684</td>
</tr>
<tr>
<td>Subarachnoid hemorrhage, open intracranial wound, loss of consciousness</td>
<td>5</td>
<td>5619</td>
</tr>
<tr>
<td>Septicemia type 1 (038.9) [10.75%]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hypertensive chronic kidney disease (V or end stage renal dis.)</td>
<td>1</td>
<td>7013</td>
</tr>
<tr>
<td>End stage renal disease</td>
<td>2</td>
<td>4992</td>
</tr>
<tr>
<td>Nephrotic syndrome in diseases classified elsewhere</td>
<td>3</td>
<td>7728</td>
</tr>
<tr>
<td>Hypertensive heart and chronic kidney disease w. heart failure and chronic kidney disease stage V or end stage</td>
<td>4</td>
<td>5254</td>
</tr>
<tr>
<td>Other ectopic pregnancy without intrauterine pregnancy</td>
<td>5</td>
<td>1951</td>
</tr>
</tbody>
</table>

3.5.6 Conclusions

In this study, several approaches were proposed for addressing disease phenotyping challenges related to disease heterogeneity. As a case study, the proposed methodology is applied to phenotype characterization of sepsis which is a highly heterogeneous disease and one of the main causes of death in the US hospitals. Conducted experiments provide evidence that the proposed approach can effectively discover informative phenotypes for
sepsis. The discovered phenotypes for identified homogeneous groups were more relevant as compared to global vectors for the same diseases. Benefits were also evident for a mortality prediction task where an increase in accuracy and prediction quality was observed when using multi-type disease embedding rather than single global embedding. In our experiments, we have compared two approaches for disease type discovery, a global clustering approach and an automatic approach, where disease types are learned within the model itself. Although easier to use, an automatic approach failed to outperform global clustering (t-models). However, it was better than the original single vector approach. Discovering disease types has shown great promise as a future research direction in electronic phenotyping, and further efforts will be taken to further the understanding of the discovered disease types as well as to build effective models capable of jointly using existing medical knowledge and large data to discover disease embeddings of higher quality.

3.6 Modeling Healthcare Quality via Compact Representations of Electronic Health Records

As aforementioned, obtaining distributed representations can be very useful for numerous tasks in aiding medical science, moreover these representations can be used to model many tasks in healthcare as well. To provide empirical evidence, we conducted a detailed study where patient’s visit representations is learned and examined how can it be used for predicting Inpatient Quality indicators (IQIs) (Stojanovic et al., 2016b).

Inpatient quality indicators were developed as a set of measures that provide a perspective on quality of patient care in hospitals\(^6\). These indicators include inpatient mortality for certain procedures and medical conditions (Dimick et al., 2004), length of stay (Goodney et al., 2003), and total charges of an inpatient stay, and can be considered as important metrics for evaluating quality of care (Marciniak et al., 1998). These measures can be used to help hospitals identify potential problem areas that might need further studies and provide

the opportunity to assess quality of care inside hospitals using administrative data found in typical discharge records. On the other hand, transparency of these indicators may help potential users of hospital care choose a hospital that will fit their needs and their financial constraints. This aspect is becoming an increasingly important issue as healthcare users are reportedly declaring personal bankruptcies during hospitalizations either due to high hospital care prices, or due to inpatient staying too long in a hospital when this might not be necessary (Barlett and Steele, 2006; Gross and Notowidigdo, 2011; Himmelstein et al., 2005; Zhan and Miller, 2003).

Unsurprisingly, one of the important metrics that the patients are worried about is how high their final hospital bill will be. However, computing this value upfront is not a trivial task, as pricing of health care services vary significantly among different providers even for the most common procedures. Each provider considers many parameters before charging a patient, and the process is different for different players in the industry. For example, Medicare takes more than one hundred parameters to determine a hospitalization reimbursement\(^7\). For these reasons, many economists, employers and health plans are advocating for providing the price quote of health care services as a way to encourage consumers to choose low-cost, high-quality providers and to promote competition based on the value of care\(^8\).

Length of stay (LoS) is another important metric for assessing the quality of health care, also useful for planning scheduling capacity within a hospital. For instance, the United Kingdom’s Department of Health treats LoS as a key performance indicator and uses it both to monitor hospital quality and to manage patients’ expectations (Carter and Potts, 2014). The length of time patients spend in hospital beds is known to be a good measure of utilization for a number of hospital resources, including staffing and equipment.


As a result, the department publishes average LoS on the National Health Service (NHS) website\(^9\) as a hospital operations parameter to help patients make more informed choices on which hospital to visit. Through such increased transparency pressure is put on hospitals to improve patient care, which involves providing more cost efficient and standardized services often reflected in duration of the service (Goodney et al., 2003). Thus, gaining a better understanding of LoS provides an opportunity to reduce the time patients stay in hospitals without affecting the quality of service, which is in the financial and personal interests of hospitals and patients. Additionally, early and accurate knowledge of LoS can aid hospital administrators in management of bed occupancy. This is a crucial problem faced by hospitals, which are pressured to shorten the LoS, potentially increasing risk of patient complications after discharge. Medicare was among the first insurance companies to consider predicting length of hospital stay for each inpatient and using it for diagnosis of related groups (Marciniak et al., 1998). The acceptance of length of stay as an indicator of resource utilization has caused a surge of interest across the healthcare industry in the predictability of LoS.

Due to increased availability of EHR’s, in recent years an increasing emphasis is given to the effective mining of clinical data to obtain actionable insights for improving healthcare delivery, a concept often termed “data-driven healthcare” (Dat, 2014; Madsen, 2014). Data-driven health care practitioners have been addressing various problems aimed to improve healthcare quality (Xiang et al., 2012; Zhou et al., 2014; Ho et al., 2014a,b; Gligorijevic et al., 2015b). The overall objective is to build a stable framework for modeling different aspects of the healthcare systems, and to provide significant insights to healthcare institutions and patients alike. Some particularly important and impactful applications are aimed towards predictive modeling of health outcomes in terms of diseases, procedures, mortality, and other measures that may have a huge impact on quality of patient treatment. The models are used to improve detection of high-risk groups of patients, or detect

\(^9\) NHS Choices. [http://www.nhs.uk], accessed June 2018
important effects not taken into consideration in prior medical treatments.

However, the modeling process is very challenging, as healthcare observational data are often sparse, heterogeneous, and/or incomplete due to different hospital and insurance policies, further aggravated by non-standardized physician practices (Hripcsak and Albers, 2013). The existing data mining tools are not fully capable of addressing the important task of healthcare modeling (Chowriappa et al., 2014), and, in order to make use of multifaceted, noisy healthcare data sources, development of novel efficient and effective machine learning approaches is required.

In this study we address this important problem, and propose a novel approach that makes use of the latest advances in the representation learning for the task of predicting inpatient length of stay, pricing, and survival rates, with the objective of modeling the quality of healthcare services.

3.6.1 The proposed approach

In this section we present a novel approach for learning low-dimensional, distributed representations of patient EHRs. As a first step, we describe how to apply state-of-the-art, unsupervised neural language models for learning embeddings of diseases and applied clinical procedures from the EHR data of individual patients. Then, the obtained embeddings are employed to find useful inpatient feature vectors, used to train predictive models of the healthcare quality indicators in a supervised manner. The entire pipeline of the proposed methodology is illustrated in Figure 3.12 and each step is presented in more details in the following sections.

Problem definition

Assume we are given a set $\mathcal{R}$ of $N$ hospital inpatient discharge records (representing a single hospital visit) and sets $\mathcal{D}$ of possible diseases and $\mathcal{P}$ procedures. Then, a discharge record $r_i = [(d_{i1}, \ldots, d_{iD_i}), (p_{i1}, \ldots, p_{iP_i})] \in \mathcal{R}$, $i = 1, \ldots, N$, of the $i^{th}$ patient is defined
as a sequence of diseases $d_i \in D$ and procedures $p_i \in P$ at the end of a hospital stay. Here, $D_i$ is the number of diagnosed diseases and $P_i$ is the number of applied procedures in the sequence, so that $D_i + P_i = H_i$ and that record is represented as $r_i = (h_{i1}, \ldots, h_{iH_i}) \in \mathcal{R}$, where $h_{il}$ can be a disease or a procedure in the sequence. Then, using the set $\mathcal{R}$, the objective is to find $M$-dimensional real-valued representations $v_d \in \mathbb{R}^M$ for every disease $d$ and $v_p \in \mathbb{R}^M$ for every procedure $p$, such that similar diseases and procedures lie nearby in the joint $M$-dimensional vector space and to use them to build a patient vector representation $x_i \in \mathbb{R}^M$ for training predictive models of the healthcare quality indicators.

Once again, the proposed model will be built on top of distributed language models described in Section 3.2.

**disease+procedure2vec method**

In this section we propose disease+procedure2vec (dp2v) approach for learning diseases and procedures representations (step 1 in Figure 3.12) that extend models of the disease2vec algorithm. Low-dimensional representations for diseases and procedures are learned by maximizing the objective function $\mathcal{L}$ over the entire set $\mathcal{R}$ of records as follows,

$$
\mathcal{L} = \sum_{r \in \mathcal{R}} \sum_{h_i \in r} \sum_{-b \leq m \leq b, m \neq 0} \log \mathbb{P}(h_{i+m} | h_i).
$$

(3.15)
Probability \( P(h_{i+m}|h_i) \) of observing some “neighboring” disease/procedure \( h_{i+m} \) given the current disease/procedure \( h_i \) is defined using the soft-max function as

\[
P(h_{i+m}|h_i) = \frac{\exp(v_{h_i}^T v'_{h_{i+m}})}{\sum_{h=1}^H \exp(v_{h_i}^T v'_h)},
\]

(3.16)

where \( v_h \) and \( v'_h \) are the input and output \( M \)-dimensional vector representations of disease/procedure \( h \) and hyper-parameter \( b \) represents the length of the context for disease records. Note that \( h \) can represents either \( d \) or \( p \), with \( H = |D| + |P| \).

![Figure 3.13](image)

**Figure 3.13:** Graphical representations of the disease+procedure2vec model. The model uses central disease/procedure \( h_i \) to predict \( b \) diseases/procedures (colored yellow and blue, respectively) that come before and \( b \) that come after it in the discharge record.

As illustrated in Figure 3.13 and equation 3.16, disease+procedure2vec uses central disease/procedure \( h_i \) to predict \( b \) diseases/procedures that come before and \( b \) diseases/procedures that come after it in the discharge record, an architecture known as the SkipGram. As a result, diseases and procedures that often co-occur and have similar contexts (i.e., with similar neighboring diseases and procedures) will have similar representations as learned by our model. Additionally, we have considered a continuous bag of words architecture (CBOW), that uses context diseases and procedures to predict a central disease or procedure, as described in Section 3.5.3, however, the SkipGram architecture was consistently more accurate than the CBOW (as shown in Figure 3.14) and as such was the one used in disease+procedures2vec model.
Table 3.13: Number of inpatient stays and number of diagnoses and procedure codes used for different healthcare providers

| Provider          | N     | |D|   | |P|   | |D|+|P|   |
|-------------------|-------|---|---|---|---|
| Medicare          | 11,300,025 | 11,636 | 3,649 | 15,285 |
| Medicaid          | 9,134,840  | 12,237  | 3,668  | 15,905  |
| Private insurance | 12,344,355 | 12,458  | 3,737  | 16,195  |
| Self-pay          | 1,247,209  | 10,640  | 3,230  | 13,870  |

The *disease+procedure2vec* model was optimized using stochastic gradient ascent, suitable for large-scale problems. However, computation of gradients is proportional to the number of unique disease and procedures in the datasets, which may be computationally expensive in practical tasks. As an alternative, we used negative sampling approach (Mikolov et al., 2013a), which significantly reduces the computational complexity.

**Patient visit representation**

Having learned the disease and procedure vectors, we aim to exploit them for the purpose of predicting total charges, length of stay, and mortality. For this purpose, we generate a data set $\mathcal{M} = \{(x_i, y_i), i = 1, ..., N\}$, where for each record $r_i$ the value of $y_i \in \mathcal{Y}$ represents one of the target variables: LoS, total charges (TOTCHG), or binary mortality indicator, and $x_i \in \mathbb{R}^M$ is a patient’s feature vector calculated by summing vectors of diseases and procedures that appear in that record (Grbovic et al., 2015) (step 2 in Figure 3.12),

$$x_i = \sum_{j=1}^{D_i} v_{dij} + \sum_{l=1}^{P_i} v_{p_{il}}, \quad (3.17)$$

Once the data set $\mathcal{M}$ is generated, the learning task is to find a prediction function $f : \mathbb{R}^M \rightarrow \mathcal{Y}$, which maps each patient visit into one of the three variables of interest depending on the task (step 3 in Figure 3.12). When predicting LoS and TOTCHG this results in a regression problem, while for mortality prediction the problem can be viewed as a classification task.
**The analysis of model parameters**

In Figure 3.14 results obtained by varying vector dimension and window size for both CBOW and SkipGram models are shown for the task of predicting total charges. The SkipGram model was consistently more accurate than the CBOW model, thus we opted to use this model in disease+procedures2vec approach. Varying parameter $b$ did not introduce much variation in the results for SkipGram, thus we chose to set context neighborhood size to $b = 40$, such that model captures larger context and most of the diseases and procedures in that record. From Figure 3.14 we can see that increasing parameter $M$ improves the accuracy, however dimensionality is increased, leading to a more complex model that is more difficult to train. Dimensionality of the embedding space was set to $M = 200$, the parameter $M$ was chosen in such a manner as to avoid larger dimensionality of the learned model while obtaining good predictive accuracy. Finally, we used 25 negative samples in each vector update for negative sampling. Similarly to the approach presented in (Mikolov et al., 2013a), the most frequent diseases and procedures were sub-sampled during the training phase.
Figure 3.15: Distribution of California inpatient hospital admissions by the primary payer (for a 2003-2011 period)

Table 3.14: Association of procedures to two high-mortality diseases discovered by measuring cosine distance on features obtained using dp2v embedding model

<table>
<thead>
<tr>
<th>Neighbors of respiratory failure</th>
<th>Neighbors of congestive heart failure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insertion of endotracheal tube</td>
<td>Insertion of implantable heart assist system</td>
</tr>
<tr>
<td>Tracheostomy toilette</td>
<td>Implantation of cardiac resynchronization defibrillator total system (CRT-D)</td>
</tr>
<tr>
<td>Other lavage of bronchus and trachea</td>
<td>Implantation of cardiac resynchronization defibrillator pulse generator (CRT-D)</td>
</tr>
<tr>
<td>Bronchoscopy through artificial stoma</td>
<td>Insertion of percutaneous external heart assist device</td>
</tr>
<tr>
<td>Other oxygen enrichment</td>
<td>Heart transplantation</td>
</tr>
<tr>
<td>Other repair and plastic operations on trachea</td>
<td>Excision destruction or exclusion of left atrial appendage (LAA)</td>
</tr>
<tr>
<td>Fiber-optic bronchoscopy</td>
<td>Aquapheresis</td>
</tr>
<tr>
<td>Infusion of vasopressor agent</td>
<td>Automatic implantable cardioverter-defibrillator (AICD) check</td>
</tr>
<tr>
<td>Replacement of tracheostomy tube</td>
<td>Noninvasive programmed electrical stimulation (NIPS)</td>
</tr>
<tr>
<td>Replacement of gastrostomy tube</td>
<td>Removal of lead(s) [electrode] without replacement</td>
</tr>
<tr>
<td>Complete glossectomy</td>
<td>Endovascular removal of obstruction from head and neck vessel(s)</td>
</tr>
<tr>
<td>Other intubation of respiratory tract</td>
<td>Replacement of automatic cardioverter-defibrillator lead(s) only</td>
</tr>
</tbody>
</table>

EHR discharge database - insurance cohorts

The detailed description of the data was given in Section 3.3. In this particular study, we split the dataset into four cohorts by insurance type to provide detailed characterization of the learned embeddings with respect to type of insurance patients had.

In Figure 3.15 we plot the distribution of inpatient admissions by primary payer (i.e., type of insurance). Histograms of diagnoses and procedures counts per visit are shown in Figures 3.1 and 3.2, respectively. Additionally, we show the number of records \(N\), unique
diseases $|\mathcal{D}|$, and procedures $|\mathcal{P}|$ for four types of health insurance in Table 3.13. To address different practices of health insurance providers, we built non-overlapping cohorts for each of four insurance groups and trained separate embedding models for each of them. The experimental setup and results are presented in the following section.

3.6.2 Empirical evaluation

In this section we first explore the embedding space learned using the proposed method, validating that the vector representations are meaningful and insightful. Then, we discuss linear predictive models used in the experiments, and describe baseline approaches for low-dimensional embedding. Lastly, we discuss experimental setup, give evaluation metrics, and present the obtained results.

Exploring associations in the embedding space

The dp2v model maps each disease and procedure into a common low-dimensional space, and in this section we provide evidence that such learned mappings are indeed medically relevant. In particular, we explored the embedding space by retrieving the nearest procedures to diseases found in the SID California database. This is done by choosing most similar procedures for a query disease via calculating cosine similarity of their vectors.

As examples of learned associations between diseases and procedures we selected to find nearest procedures for respiratory failure and congestive heart failure (CHF), two conditions that exhibit high mortality among patients. We retrieved 12 nearest procedures for each query disease, and show the results in Table 3.14. We can see that for the respiratory failure the method retrieved several procedures that serve to aid in breathing of the patient, such as insertion of endotracheal tube, tracheostomy toilette, repair and plastic operations on trachea, replacement of tracheostomy and gastrostomy tube, intubation of respiratory tract, and oxygen enrichment. We also see procedures that are commonly applied prior to bronchus examination and for bronchus cleaning, such as bronchoscopy for
throat, trachea examination, and lavage of bronchus and trachea.

For the *congestive hearth failure* disease discovered associated procedures also confirm that dp2v embeddings are medically relevant. Several procedures in the top 12 list include different implants aimed to assist the heart (e.g., CRT, AICD) or electro method performed to stimulate heart pumping (e.g., NIPS). Other procedures include heart transplantation, aquapheresis (which treats fluid overflow that can be caused by CHF), or endovascular removal of blood clots that can be caused by a heart attack. The results validate the quality of the learned representations, where medically relevant diseases and procedures were found to be nearby in the embedding space.

### Predictive models

Several penalized linear models for regression and classification tasks are used in our experiments, as suggested in the relevant literature (Tierney et al., 1995; Moran et al., 2007). In particular, for regression problems we apply linear regression,

\[
y_i = f(w, x_i) = w^T x_i + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2),
\]

where \(\varepsilon\) is a zero-mean Gaussian noise with variance \(\sigma^2\). On the other hand, for the classification problem we use the logistic regression model,

\[
y_i = f(w, x_i) = I\left(\frac{1}{1 + \exp \left( - (w^T x_i) \right)} > 0.5\right).
\]

<table>
<thead>
<tr>
<th>Penalty</th>
<th>Optimization problem</th>
<th>Model name</th>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lasso</td>
<td>(\min_w f(w, x) + \lambda</td>
<td>w</td>
<td>_1)</td>
<td>LeastR</td>
</tr>
<tr>
<td>Group Lasso</td>
<td>(\min_w f(w, x) + \lambda</td>
<td>w</td>
<td>_1)</td>
<td>gLLeastR</td>
</tr>
<tr>
<td>Fused Lasso</td>
<td>(\min_w f(w, x) + \lambda</td>
<td>w</td>
<td><em>1 + \lambda \sum</em>{i=1}^{M-1}</td>
<td>w_i - w_{i+1}</td>
</tr>
<tr>
<td>Sparse group Lasso</td>
<td>(\min_w f(w, x) + \lambda</td>
<td>w</td>
<td><em>1 + \sum</em>{i=1}^{G} \lambda_{G_i}</td>
<td>w_{G_i}</td>
</tr>
<tr>
<td>Overlapping group Lasso</td>
<td>(\min_w f(w, x) + \lambda</td>
<td>w</td>
<td><em>1 + \sum</em>{i=1}^{G} \lambda_{G_i}</td>
<td>w_{G_i}</td>
</tr>
</tbody>
</table>
Vector \( \mathbf{w} \) is an unknown set of weights for both prediction models, and \( I(\cdot) \) is an indicator function equal to 1 if the argument is true and 0 otherwise.

In addition, for both models we explored a number of regularization approaches, ranging from \( \ell_1 \) Lasso to overlapping group Lasso penalizations. We summarized the training objectives of five penalized linear models in Table 3.15, where \( \ell_1 \) indicates Lasso norm and \( \ell_q \) is norm of the non-overlapping groups, \( w_i \) and \( w_{Gi} \) indicate a single dimension of the weight vector and a group of dimensions defined by the index set \( G_i \), respectively. For the sparse group Lasso, the index sets \( G_i \) do not overlap (i.e., \( G_i \cap G_j = \emptyset, \forall i \neq j \)), which is not the case for the overlapping group Lasso. The index sets \( G_i \) for group Lasso models were defined in groups of ten consecutive features, indexed from 1 to 10, 11 to 20, and so on until \( M - 9 \) to \( M \) (smaller groups showed better performance). For the overlapping group Lasso the index sets were defined as 1 to 20, 11 to 30, and so on. All \( \lambda \) parameters were set to be equal and chosen from range \([0.01, 0.1]\), determined through cross-validation. In the conducted experiments, an implementation from the efficient SLEP package (Liu et al., 2009) is used for training the models.

**Low-dimensional embedding baselines**

As the objective of our work is to find meaningful representations of diagnoses and procedures in a low-dimensional space, we compare the proposed embedding approach to a number of state-of-the-art alternatives. More specifically, we considered Latent Dirichlet Allocation (LDA) (Blei et al., 2003), as a representative of topic learning models, as well as spectral clustering (Tang and Liu, 2011) and modularity (Newman, 2006) approaches used for low-dimensional representations of nodes in an undirected graph representing co-occurrence of diagnoses and procedures. In addition, we examined binary encoding in the original \( \mathbb{R}^{|\mathcal{D}|+|\mathcal{P}|} \) space and applied PCA on such sparse representation. In the following sections we briefly describe the baseline embedding methods.
**Binary coding with dimensionality reduction (dPCA)**

A high-dimensional representation of EHR records is obtained by creating a binary vector of \(|D| + |P|\) entries corresponding to the total number of unique diagnoses and procedures found in the SID California database (the values of \(|D|\) and \(|P|\) can be found in Table 3.13). Each entry in the extended representation is either 0 or 1 depending whether that particular diagnoses or procedure occurred in that discharge record. As the dimensionality of this problem is large, we apply PCA (van der Maaten et al., 2009) to reduce dimensionality of the problem to \(M\) dimensions (in our experiments we set the dimensionality of the embedding space to \(M = 200\) for all methods).

**Spectral clustering (Spec)**

If we consider an undirected network \(G\) of co-occurrences of diagnoses and procedures in hospital discharge data, we can use advanced tools to learn node representation in \(\mathbb{R}^M\) space using the information from the graph. The spectral clustering method generates a representation in \(\mathbb{R}^M\) space from the first \(M\) eigenvectors of \(L\), a normalized graph Laplacian of graph \(G\) (Tang and Liu, 2011). The Laplacian is defined as \(L = D - A\), where \(D = \text{diag}(d_1, d_2, \ldots, d_N, p_1, p_2, \ldots, p_N)\) and \(A\) is the adjacency matrix of \(G\). The normalized Laplacian \(L\) is then defined as

\[
L = D^{-1/2}LD^{-1/2}. \tag{3.20}
\]

Then, we find the first \(M\) eigenvectors of the normalized Laplacian and treat them as latent dimensions of nodes from the graph \(G\), thus inferring low-dimensional representations for both procedures and diagnoses.

**Modularity (Mod)**

This method generates a representation in \(\mathbb{R}^M\) space from the top \(M\) eigenvectors of \(B\), the modularity matrix of \(G\). For two nodes \(i\) and \(j\) in the graph \(G\) with degrees \(d_i\) and \(d_j\), respectively, the expected number of edges between these two nodes in a uniform random graph model is \(\frac{d_i d_j}{2m}\), where \(m\) represents the total number of edges in
the graphs. Modularity matrix $B$ measures the deviation of adjacency matrix $A$ from a uniform random graph with the same degree distribution,

$$B = A - \frac{1}{2m}dd^T. \quad (3.21)$$

While in many real graphs the adjacency matrix $A$ is typically very sparse, the modularity matrix $B$ is typically dense. The matrix $B$ is then decomposed using SVD method and the obtained eigenvectors of $B$ encode information in $\mathbb{R}^M$ space about modular partitions of the graph $G$ (Newman, 2006), which are used to represent the nodes in a lower-dimensional space.

*Latent Dirichlet Allocation (LDA)*  
LDA is a popular latent topic model (Blei et al., 2003), shown to obtain a state-of-the-art performance in many tasks both within and outside of the domain of the natural language processing. Assuming a fixed number of topics that generated the data, the model learns a topic distribution over the diseases and procedures, effectively embedding them in the topic space. Then, the found topical representations can be used as feature vectors in the classification and regression models.

*Evaluation metrics*

For evaluation of the proposed regression methods we use a goodness-of-fit metric $R^2$ defined as follows,

$$R^2 = 1 - \frac{\sum_i(y_i - \hat{y}_i)^2}{\sum_i(y_i - \mu)^2}, \quad (3.22)$$

where $y_i$ and $\hat{y}_i$ are true and predicted values of the target variable for the record $r_i$, respectively, and $\mu$ is the mean value for all records in the set $\mathcal{R}$.

For evaluation of patient survival analysis we use an accuracy measure defined as follows,

$$\text{accuracy} = \frac{tp + tn}{tp + fp + tn + fn}, \quad (3.23)$$
Table 3.16: Average total charges, length of stay in days, and survival rate for four datasets from SID California database

<table>
<thead>
<tr>
<th>Provider</th>
<th>TOTCHG</th>
<th>LoS</th>
<th>Survival rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medicare</td>
<td>$50,878.02</td>
<td>5.94</td>
<td>0.96</td>
</tr>
<tr>
<td>Medicaid</td>
<td>$30,264.11</td>
<td>4.51</td>
<td>0.99</td>
</tr>
<tr>
<td>Private insurance</td>
<td>$29,412.26</td>
<td>3.71</td>
<td>0.99</td>
</tr>
<tr>
<td>Self-pay</td>
<td>$31,824.64</td>
<td>3.97</td>
<td>0.98</td>
</tr>
</tbody>
</table>

where \( tp \) and \( tn \) denote true positives and true negatives, respectively (i.e., correctly classified cases), while \( fp \) and \( fn \) denote false positive and false negative test examples, respectively (i.e., mistakenly classified cases).

Results

In this section we provide experimental results of three predictive tasks on four insurance data sets. Different representations of diagnoses and procedures were trained for each insurance data set, and learned using five competing approaches. In particular, four datasets were created for each of the insurance categories. From the first month of the observation period we sampled 100,000 records for training and testing predictive models, while the remaining data was used for learning the embedding models. From the 100,000 sampled examples, 80% were randomly chosen for regression and classification training, while 20% were used for testing. In addition, as hospitals currently report mean values for TOTCHG, LoS, and survival rate, shown in Table 3.16, we also use these values as a naïve baseline. We further comment on their performance in the following sections.

Prediction of total charges (TOTCHG)

In this section we address the problem of predicting total charges for a patient per hospital visit. As discussed previously, there are more than 100 factors that may influence hospital charges, making the estimation of the exact value a non-trivial problem. For example, Table 3.16 suggests that Medicare patients are charged almost twice as much as the other
three groups of patients (which are similar with respect to average charges). As Medicare patients are people of age, we can assume that they are diagnosed with more conditions and have more procedures performed compared to the other three insurance groups.

We first used the mean TOTCHG computed on the training data as a trivial baseline predictor and measured its accuracy on the test data for each provider. We observed that this trivial predictor underperformed and obtained $R^2 < 0$. The result indicates that the information provided by hospitals is of little value for an individual patient, and in the following we explore more involved approaches for this predictive task, where as an input we take into account diagnosed diseases for a specific patient and a list of procedures that might be applied.

In Table 3.17, we show the results in terms of $R^2$ measure obtained by five regression models for four insurance categories, making use of a 200-dimensional representations obtained by various embedding methods. We observe that the proposed dp2v model outperformed the baseline approaches in all 20 experiments (for all five regression models and for all four insurance categories). The $R^2$ improvement of using the proposed embedding over the best performing alternative is on average around 20%. The obtained results strongly suggest that the most useful representation for predicting total charges is learned using dp2v model. We also see that the LR regression model outperformed alternatives in this application, and that the most difficult task was to estimate costs for patients on Medicaid insurance.

**Prediction of length of stay (LoS)**

The length of stay is one of the most important indicators of quality of a hospital system, and is an important parameter considered when choosing a hospital. Therefore, providing LoS estimation for a specific visit is a very important task. Many hospitals are handling these predictions by reporting the mean length of stay. Similarly to the total charges, our experiments indicate that such a summary statistic is not informative for individual patients.
Table 3.17: $R^2$ results obtained for predicting total charges by five regression models for four insurance categories

<table>
<thead>
<tr>
<th></th>
<th>LR</th>
<th>gLR</th>
<th>fLR</th>
<th>sgLR</th>
<th>oLR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medicare</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.6454</td>
<td>0.6388</td>
<td>0.5846</td>
<td>0.3641</td>
<td>0.4204</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5584</td>
<td>0.5274</td>
<td>0.3487</td>
<td>$\leq 0$</td>
<td>0.02218</td>
</tr>
<tr>
<td>Mod</td>
<td>0.5635</td>
<td>0.5235</td>
<td>0.3628</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>LDA</td>
<td>0.2022</td>
<td>0.2040</td>
<td>0.1955</td>
<td>0.2141</td>
<td>0.2008</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.5059</td>
<td>0.4805</td>
<td>0.3300</td>
<td>$\leq 0$</td>
<td>0.0005</td>
</tr>
<tr>
<td>Medicaid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.5850</td>
<td>0.5805</td>
<td>0.5646</td>
<td>0.4550</td>
<td>0.4550</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5155</td>
<td>0.5138</td>
<td>0.4423</td>
<td>0.1892</td>
<td>0.2836</td>
</tr>
<tr>
<td>Mod</td>
<td>0.5163</td>
<td>0.5092</td>
<td>0.4490</td>
<td>0.0945</td>
<td>0.1769</td>
</tr>
<tr>
<td>LDA</td>
<td>0.2052</td>
<td>0.2046</td>
<td>0.1974</td>
<td>0.1630</td>
<td>0.1511</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.4112</td>
<td>0.4118</td>
<td>0.3094</td>
<td>0.0601</td>
<td>0.1166</td>
</tr>
<tr>
<td>Private insurance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.6553</td>
<td>0.6434</td>
<td>0.5930</td>
<td>0.2903</td>
<td>0.3773</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5744</td>
<td>0.5539</td>
<td>0.4401</td>
<td>0.1038</td>
<td>0.1801</td>
</tr>
<tr>
<td>Mod</td>
<td>0.5757</td>
<td>0.5516</td>
<td>0.4111</td>
<td>0.0196</td>
<td>0.0374</td>
</tr>
<tr>
<td>LDA</td>
<td>0.1936</td>
<td>0.1932</td>
<td>0.1692</td>
<td>0.1610</td>
<td>0.1516</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.5688</td>
<td>0.5438</td>
<td>0.4967</td>
<td>0.0768</td>
<td>0.1875</td>
</tr>
<tr>
<td>Self-pay</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.6093</td>
<td>0.5954</td>
<td>0.5575</td>
<td>0.3281</td>
<td>0.3375</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5246</td>
<td>0.4989</td>
<td>0.4100</td>
<td>0.0686</td>
<td>0.1491</td>
</tr>
<tr>
<td>Mod</td>
<td>0.4756</td>
<td>0.4672</td>
<td>0.3680</td>
<td>0.0194</td>
<td>0.0879</td>
</tr>
<tr>
<td>LDA</td>
<td>0.0939</td>
<td>0.0945</td>
<td>0.0864</td>
<td>0.0787</td>
<td>0.0455</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.6048</td>
<td>0.5706</td>
<td>0.4390</td>
<td>0.1057</td>
<td>0.1689</td>
</tr>
</tbody>
</table>

($R^2 < 0$).

In this study we consider a patient that is diagnosed with several diseases, and we account for procedures suggested for this patient in order to estimate the patient’s length of stay. The results of five regression models learned on latent features projected by the competing models are shown at Table 3.18. We observe that the proposed dp2v model was the best choice in 18 out of 20 experiments, obtaining average accuracy improvements up to 34% for Medicare, 19% for Medicaid, and 20% for self-pay patients over the best
Table 3.18: $R^2$ results obtained for predicting LoS by five regression models for four insurance categories

<table>
<thead>
<tr>
<th></th>
<th>LR</th>
<th>glLR</th>
<th>fLR</th>
<th>sgLR</th>
<th>olLR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Medicare</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.4356</td>
<td>0.4260</td>
<td>0.3872</td>
<td>0.2687</td>
<td>0.3411</td>
</tr>
<tr>
<td>Spec</td>
<td>0.4092</td>
<td>0.3989</td>
<td>0.2840</td>
<td>0.0598</td>
<td>0.0935</td>
</tr>
<tr>
<td>Mod</td>
<td>0.4136</td>
<td>0.3955</td>
<td>0.2569</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>LDA</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.3337</td>
<td>0.3149</td>
<td>0.2538</td>
<td>$\leq 0$</td>
<td>0.0005</td>
</tr>
<tr>
<td><strong>Medicaid</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.3220</td>
<td>0.3178</td>
<td>0.3089</td>
<td>0.1876</td>
<td>0.1964</td>
</tr>
<tr>
<td>Spec</td>
<td>0.2691</td>
<td>0.2571</td>
<td>0.1906</td>
<td>0.0392</td>
<td>0.0818</td>
</tr>
<tr>
<td>Mod</td>
<td>0.2910</td>
<td>0.2641</td>
<td>0.1813</td>
<td>0.0093</td>
<td>0.0259</td>
</tr>
<tr>
<td>LDA</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.2715</td>
<td>0.2575</td>
<td>0.1703</td>
<td>0.0253</td>
<td>0.0423</td>
</tr>
<tr>
<td><strong>Private insurance</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.3657</td>
<td>0.3599</td>
<td>0.3874</td>
<td>0.0493</td>
<td>0.1230</td>
</tr>
<tr>
<td>Spec</td>
<td>0.3463</td>
<td>0.3507</td>
<td>0.2528</td>
<td>0.0155</td>
<td>0.0321</td>
</tr>
<tr>
<td>Mod</td>
<td>0.3508</td>
<td>0.3574</td>
<td>0.2404</td>
<td>$\leq 0$</td>
<td>0.0125</td>
</tr>
<tr>
<td>LDA</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.2893</td>
<td>0.3448</td>
<td>0.2342</td>
<td><strong>0.0702</strong></td>
<td><strong>0.1254</strong></td>
</tr>
<tr>
<td><strong>Self-pay</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.2402</td>
<td>0.2383</td>
<td>0.2137</td>
<td>0.0766</td>
<td>0.0945</td>
</tr>
<tr>
<td>Spec</td>
<td>0.1402</td>
<td>0.1279</td>
<td>0.0813</td>
<td>$\leq 0$</td>
<td>0.0026</td>
</tr>
<tr>
<td>Mod</td>
<td>0.1459</td>
<td>0.1290</td>
<td>0.0743</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>LDA</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
<td>$\leq 0$</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.0876</td>
<td>0.0774</td>
<td>0.0432</td>
<td>$\leq 0$</td>
<td>0.0015</td>
</tr>
</tbody>
</table>

performing alternative. Interestingly, for private insurances the proposed model did not provide improvement for all predictive models. Nevertheless, the model that performed the best on this dataset used features learned by the dp2v embedding method. We can conclude that the proposed embedding approach provides the best features for prediction of length of stay among the considered models overall.
**Prediction of inpatient survival**

Lastly, we turn our attention to estimating patient’s mortality, which we use as an ultimate quality indicator of hospital care considered in this study (Pfister et al., 2015). More specifically, the prediction task was to estimate patient’s survival probability, taking into consideration diagnosed conditions and conducted procedures.

Table 3.19: Mortality prediction accuracy by five classification models for four insurance categories

<table>
<thead>
<tr>
<th></th>
<th>logR</th>
<th>gLogR</th>
<th>fLogR</th>
<th>sgLogR</th>
<th>olLogR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medicare</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.6256</td>
<td>0.6131</td>
<td>0.5385</td>
<td>0.5433</td>
<td>0.5332</td>
</tr>
<tr>
<td>Spec</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
</tr>
<tr>
<td>Mod</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
<td>0.4923</td>
</tr>
<tr>
<td>LDA</td>
<td>0.4928</td>
<td>0.4928</td>
<td>0.4928</td>
<td>0.4928</td>
<td>0.4928</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.4825</td>
<td>0.4825</td>
<td>0.4825</td>
<td>0.4825</td>
<td>0.4825</td>
</tr>
<tr>
<td>Medicaid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.8289</td>
<td>0.8273</td>
<td>0.7796</td>
<td>0.7566</td>
<td>0.7928</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
</tr>
<tr>
<td>Mod</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
<td>0.5066</td>
</tr>
<tr>
<td>LDA</td>
<td>0.5164</td>
<td>0.5164</td>
<td>0.5164</td>
<td>0.5164</td>
<td>0.5164</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
</tr>
<tr>
<td>Private insurance</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.8714</td>
<td>0.8643</td>
<td>0.7405</td>
<td>0.7619</td>
<td>0.7524</td>
</tr>
<tr>
<td>Spec</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
</tr>
<tr>
<td>Mod</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
<td>0.5167</td>
</tr>
<tr>
<td>LDA</td>
<td>0.4881</td>
<td>0.4881</td>
<td>0.4881</td>
<td>0.4881</td>
<td>0.4881</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.5769</td>
<td>0.5769</td>
<td>0.5769</td>
<td>0.5769</td>
<td>0.5769</td>
</tr>
<tr>
<td>Self-pay</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dp2v</td>
<td>0.8435</td>
<td>0.8252</td>
<td>0.6125</td>
<td>0.6357</td>
<td>0.5391</td>
</tr>
<tr>
<td>Spec</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
</tr>
<tr>
<td>Mod</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
<td>0.4951</td>
</tr>
<tr>
<td>LDA</td>
<td>0.4792</td>
<td>0.4792</td>
<td>0.4792</td>
<td>0.4792</td>
<td>0.4792</td>
</tr>
<tr>
<td>dPCA</td>
<td>0.4764</td>
<td>0.4764</td>
<td>0.4764</td>
<td>0.4764</td>
<td>0.4764</td>
</tr>
</tbody>
</table>

From Table 3.16, we observe that data sets for this prediction task are highly imbalanced. Therefore, in order to make a fair comparison we drew a balanced sample for
each of the insurance categories and learned classification models on such data. From Table 3.19 we observe that survival for the Medicare group was the most difficult to predict, and that for the private insurance group classification models perform the best when compared to other insurance categories. Nevertheless, mirroring the result from the previous experiments, we can see that the features learned by the dp2v method resulted in the highest accuracy, outperforming the competing approaches by a significant margin.

3.6.3 Conclusions

In this study proposed a novel unsupervised approach for learning representations of inpatients, diseases and procedures from large hospitalization records database. We compared our approach to four competitive baselines on three different predictive tasks, where we applied five regression and classification models. Experiments on predicting important inpatient quality indicator values for a potential patient stay were conducted on a large-scale inpatient EHR database, with four cohorts defined according to insurance categories. Benefits of using the proposed embedding approach versus the alternatives were shown of a majority of conducted experiments, demonstrating the power of the proposed approach and its potential for modeling healthcare quality. However, the methodology still possesses drawbacks in terms of modeling diseases and procedures embeddings. For example, currently the model does not account for the concept of primary diagnosis and secondary diagnoses, heterogeneity of a disease is not captured well by the given approach and multiple visits of same patients, including readmission, are not included in the modeling process. Modeling longitudinal effects and addressing disease heterogeneity ought to be addressed in the future research endeavors.
CHAPTER 4

LEARNING INTERPRETABLE DEEP
ARCHITECTURES FOR HEALTH
INFORMATICS WITH UNCERTAINTY

4.1 Introduction

In this chapter we explore several problems of high importance in health informatics and
propose novel deep architectures that build on the assumption of high quality neural em-
beddings of medical concepts explored in detail in Section 3. To further explore the im-
portance and practical aspect of using such models in practice we develop interpretable
deep model described in Section 4.2.

4.2 Deep Attention Model for Triage of Emergency Department Patients

Optimization of patient throughput and wait time in emergency departments (ED) is an
important task for hospital systems. For that reason, Emergency Severity Index (ESI)
system for patient triage was introduced to help guide manual estimation of acuity levels,
which is used by nurses to rank the patients and organize hospital resources. However,
despite improvements that it brought to managing medical resources, such triage system
greatly depends on nurse’s subjective judgment and is thus prone to human errors. Here, we propose a novel deep model based on the word attention mechanism designed for predicting a number of resources an ED patient would need.

Our approach incorporates routinely available continuous and nominal (structured) data with medical text (unstructured) data, including patient’s chief complaint, past medical history, medication list, and nurse assessment collected for 338,500 ED visits over three years in a large urban hospital. Using both structured and unstructured data, the proposed approach achieves the AUC of $\sim 88\%$ for the task of identifying resource intensive patients (binary classification), and the accuracy of $\sim 44\%$ for predicting exact category of number of resources (multi-class classification task), giving an estimated lift over nurses’ performance by 16% in accuracy. Furthermore, the attention mechanism of the proposed model provides interpretability by assigning attention scores for nurses’ notes which is crucial for decision making and implementation of such approaches in the real systems working on human health.

4.2.1 Introduction to Triage of Emergency Department and Problem Statement

Due to patient overcrowding and increased acuity of waiting patients, the Emergency Departments (EDs) have been under ever-increasing pressure to improve utilization of their resources. For that reason, EDs of many hospitals in the US have been investing into improvements to their services by optimizing staff and resource requirements, patient wait time, and treatment outcomes (Gilboy et al., 1999; Stojanovic et al., 2017). The hospital management teams are highly motivated in improving the system efficiency, such that medical help can reach both the highest severity of illness and intensity of service groups, while also providing a separate queue for the least resource-intensive patients in an expedited manner\(^1\).

\(^1\) https://www.acep.org/Clinical---Practice-Management/Utilization-Review-FAQ/, accessed June 2018
An important aspect of the patient resource allocation in ED systems begins with the triage processes. When patients arrive at the emergency department, they are processed at the triage area by a nurse who listens to their complaint and assesses acuity (i.e., urgency), completes entry of discrete history items plus writes a note summarizing the findings (e.g., patient’s medical history and symptoms), and lastly measures basic body functions such as heart rate, blood pressure, or temperature. Then, once all relevant information is collected, it is used to assign each patient to a triage category. The practice of assessing patients’ acuity levels and predicting the amount of required resources for treatment is called the triage process. Rating systems for acuity triage have traditionally been based solely on the illness severity of a patient, determined through the nurse’s assessment of vital signs, subjective and objective information, past medical history, allergies, and medication (Gilboy et al., 1999; Gilboy, 2012). Then, a nurse would assign the acuity level by making a judgment regarding severity of illness and intensity of service to help determine the acuity queue each patient is assigned to. The acuity assignment impacts the waiting time for each patient to transition to the next phase of care, typically the treatment area. However, this process was inherently flawed due to high variance and subjectivity of the practitioners, and variation in ability to correctly predict needed resources (Kosowsky et al., 2001).

The Emergency Severity Index (Gilboy et al., 2012) estimation of acuity has become the standard triage method over the past 15 years, and is the most dominant implemented system in the emergency departments in the United States at the moment (McHugh et al., 2012). The index has five levels of acuity (Wuerz et al., 2000), where levels 1 and 2 are ranked as highly urgent and patients with such acuity are given the highest priority. On the other hand, the ESI levels of acuity 4 and 5 are considered non-urgent, and are often given lower priority or more commonly are placed in a separate queue. The ESI system was created with a goal to aid in patient triage and to help separate more complex (or resource-intensive) patients from those with simpler problems, and thus improve patient throughput and disposition decision (Gilboy et al., 2012). The ESI triage improved over
traditional systems by introducing nurses’ estimation of the number of resources that a less acute patient would need during his/her ED stay. Resources may include the number of lab tests (e.g., blood or urine), ECG, X-ray, CT, MRI, or therapeutic interventions like fluids hydration or medications. Even consultation with a specialist is considered as a resource of emergency department.

![Emergency Severity Index (ESI): A Triage Tool for Emergency Department](image)

**Figure 4.1:** Emergency Severity Index (ESI): A Triage Tool for Emergency Department

Note that resource prediction is only used for less acute patients. At decision steps I and II on the ESI algorithm (Figure 4.1), the nurse decides which patients meet criteria for ESI levels 1 and 2 based only on patient severity of illness. However, at decision step III, the nurse assigns ESI levels 3 to 5 by assessing both acuity and predicted resource needs. Thus, the triage nurse only considers resources when the answers to decision Step I and II

---


120
are "no." Correct estimation of resource consumption has significant consequences, as it was shown that patients who require two or more resources have higher rates of hospital admission and mortality, as well as longer lengths of stay in the ED (Eitel et al., 2003; Tanabe et al., 2004). As such, the ESI system was created with an assumption that triage nurses would be able to accurately estimate the number of resources for an ED stay using the flow algorithm shown in Figure 4.1 and discriminate high acuity patients from low acuity ones.

However, the data collected from 338,500 ED visits shows that assessing the number of resources needed for patients’ treatments is not an easy task for the nurses. Namely, Figure 4.2a shows that 55% of patients were placed into Level 3 ESI category, even though they actually required less than 2 resources in 35% of the cases (Figure 4.2b). In addition, despite the size and variety of cases belonging to Level 3 ESI category, this group is processed solely in a first-come, first-serve manner. The order of processing does not take into account within-group resource allocation nor the propensity to have more severe illness, leading to suboptimal organization and utilization of resources.

Therefore, this study focuses on making an efficient and effective system for predicting the number of resources patients would need, in order to provide the ranking among them and assist in patient prioritization and disposition. Specifically, our goal is to utilize routinely available structured data, together with the unstructured textual triage notes and devise a binary-classification model capable of discriminating resource intensive patients from less demanding patients, as well as multi-class classification model for estimating the actual category of number of resources patient would need with better accuracy than triage nurses.

Thus far, natural language understanding presented a huge challenge to convert nurse’s notes into structured data useful for prediction of patient visit outcomes (Chapman et al., 2011; Wang et al., 2015). However, given the amount of unstructured textual data that ED information systems contain, utilizing such information can be a crucial step towards
obtaining satisfactory results. Recent advances in the field of Deep Learning and Natural Language Processing (NLP) have provided powerful approaches for extracting discriminative features from unstructured textual data. The novel advances proposed different approaches such as neural word embeddings (Bengio et al., 2003; Mikolov et al., 2013a) or recurrent neural networks (RNNs) (Jagannatha and Yu, 2016; Sutskever et al., 2014) that have shown excellent results in many NLP tasks.

To solve this problem, we propose a novel deep architecture, named Deep Attention Model (DAM). It is built upon bi-directional recurrent neural network (bi-RNN) (Schuster and Paliwal, 1997) suitable for modeling sequence data, such as free text data. Bi-RNN’s are not dependent on single direction sequence of information, which is of high importance for texts that are written rapidly and by different persons. To further mend the issues of triage nurses’ noisy text data, we propose the attention based mechanism on top of the bi-RNN word representations to learn attention scores for each word in the notes,
which would allow interpretability of the deep architecture outputs. We conduct a thorough experimental evaluation using three years of emergency admission data to examine how the proposed model compare to nurses’ performance, as well as current state-of-the-art approaches for text classification. We further investigate interpretability of the model by analyzing attention weights on short medical texts. Our results suggest potential for substantial improvement of the triage system performance over the current practices.

4.2.2 Related Work

Firstly, we describe problems and some existing approaches in mining medical text data and then present some notable existing methodological advances used for building highly discriminative features from text which we will exploit for predicting ED admission outcomes.

Medical free-text data mining approaches

Learning from clinical text is a long-standing challenge, and analysis of such data has become a major focus of research community recently. However, the problem persists as a difficult one, as there are no clear standard methods or tools for analyzing medical text data yet (Chapman et al., 2011). In (Nguyen et al., 2010) authors predict lung cancer stages from free-text pathology reports. These reports are analyzed using symbolic rule-based approach that uses SNOMED clinical terms to extract key lung cancer characteristics from free-text reports. Approach in (Coden et al., 2009) predicts cancer and its disease progression from pathology reports using another rule based system for automatic conversion of unstructured pathology reports into a structured data. These models require handcrafting features from text data which is a long and tedious work, and often requires revisiting existing model to improve features.

To automatically learn features from text data (Jagannatha and Yu, 2016) proposed to

4 http://www.snomed.org/snomed-ct, accessed October 2017
use the bi-RNN algorithm for detection of medical events based on texts from medical records. In their experiments it was shown that bi-RNN significantly outperforms existing tools for medical text analysis. Approaches for automatic feature extraction are more convenient, do not require handcrafted features and have shown to yield representations of highest quality. Our approach builds on the results of (Jagannatha and Yu, 2016) as we use the bi-RNN as one of the building blocks of our model which aims to improve waiting room patient disposition.

**Neural embedding models for text**

Many advances in modeling sequence data were made in the field of natural language processing, where models for mathematical characterization of language were proposed. Namely, models for distributed low-dimensional representations of words or tokens were initially proposed in (Rumelhart et al., 1986) and were successfully applied more recently in (Bengio et al., 2003). Distributed embedding approaches take advantage of word order, and follow the assumption of $n$-gram language models that neighboring words are statistically more dependent.

Typically, these models learn the probability distribution of the next word given a number of preceding words, which act as the context. The p.d. $\Pr(w_t | w_{t-n+1} : w_{t-1})$ is typically approximated using a neural network (Bengio et al., 2003) trained to predict a word $w_t$ by projecting the concatenation of vectors for context words $(w_{t-n+1}, \ldots, w_{t-1})$ into a latent representation with multiple non-linear hidden layers and the output softmax layer (Bengio et al., 2003). More recently, novel architectures (such as continuous bag-of-words (CBOW) and SkipGram that observe both preceding and later words in the sequence) have shown great improvements in representational power and training speed compared to the traditional neural embedding models (Mikolov et al., 2013a). These approaches have been discussed in greater length throughout Chapter 3.

Furthermore, we discuss deeper architectures capable of learning very discriminative
and abstract representations of texts.

Recurrent Neural Networks for text modeling

The Recurrent Neural Networks (RNNs) models are popular for modeling sequence data. Their power lies in the fact that they maintain an internal state that is updated sequentially which learns representations of word sequences that are used as a proxy for predicting the target, while in previously described approaches, word sequence was often modeled by an order-oblivious sum. Ability to stack multiple layers generates higher order representations that yield great improvements of the model on many tasks. Particular success was achieved using long-short term memory (LSTM) cell as an architecture of RNNs (Hochreiter and Schmidhuber, 1997). More recently, popular sequence-to-sequence paradigm for RNNs was proposed, where input sequence is encoded using the “encoder” network, and output sequence is generated using the “decoder” network (Sutskever et al., 2014). This paradigm has been successfully used for translating sentences from one language into another.

Word Attention Models  Attention models build upon sequence-to-sequence paradigm of RNNs (encoder-decoder networks) by dynamically re-weighting (i.e. focusing attention) on various elements of the source (text) representation during the decoding process, and they have demonstrated considerable improvements over their non-attention counterparts (Bahdanau et al., 2014). Attention mechanism was developed as a separate neural network that takes sequence of word embeddings and learns attention scores for each word, with higher attention assigning to more “important” words in the document leading to more focused higher order representation of the sequence. Additionally, this mechanism is not limited to learning scores of words but can be also applied to sentences or other segments of text as well (Gligorijevic et al., 2018). Attention models have more recently been adapted for the general setting of learning compact representations of documents (Zhai
et al., 2016).

**bi-LSTM**  Another interesting paradigm are bi-directional recurrent neural networks, where two RNNs (i.e. LSTM, thus bi-LSTM), encode the text as a forward and backward sequence, respectively (Schuster and Paliwal, 1997). Final words representation is obtained by concatenating representations of the two LSTMs and it was observed that bi-LSTM’s perform well on datasets without the strict order in sequences, such is the case with triage text data.

**Convolutional Neural Networks for text**

Recently, architectures for sequence modeling increasingly include temporal convolutions as building blocks. A good example of such approach is ConvNet for text classification (Zhang et al., 2015) and Very Deep CNN (VDCNN) model (Conneau et al., 2016), both of which are using temporal convolutions to model sequence of words (or characters) with task to perform classification. These models have been successful to outperform RNN based models. In this study, we will use word-level VDCNN as our primary baseline.

It should be noted that the above mentioned convolutional approaches are designed for character level modeling, while we use them for word level modeling. Our reasons are that medical notes do not generate sufficient amount of data for character level models to learn proper mappings, even though advantages of such approaches can be very useful in our setting: Nurses use many abbreviations, make many typos during triage process, etc. Our initial experiments on character level modeling were unsatisfactory, and further pursue for such approach will be left for future work.

**4.2.3 Proposed Model**

The architecture of the triage acuity prediction model, which we refer to as Deep Attention Model (DAM), is presented in Figure 4.3. Before exploring its architecture, we first
explain how the input is represented.

**Text embedding block**  Let us assume that we are given a set $\mathcal{P}$ of patient records, where each patient record $p_n$ contains an unstructured portion of data $p_{un}^n$, and a structured portion of data $p_{sn}^n$ and an outcome category $c_n$. Unstructured portion of the data contains text on chief complaint, medical history, home medications and nursing notes, while structured
portion of the data includes routinely available continuous and nominal data such as hearth rate, blood pressure, temperature, patient age group and other.

In order to extract meaningful features from unstructured portion of the data, we consider $p_n^u$ of length $l_n$ as a document containing a sequence of words $(w_1, w_2, \ldots, w_{l_n})$. Document length $l_n$ is maintained via null character padding and cropping. We first embed each word in the document $p_n^u$ into a $d_w$ dimensional vector using word embedding layer. To learn different interaction between words as they might repeat or correlate in the sections of the triage nurses’ notes we pass our embeddings through a dense layer to obtain $d_m$ dimensional word vectors. These cross learned word vectors are then passed through bi-LSTM layer that learns sequence dependency of words in both directions. bi-LSTM layer has the capability to mend the bias in taking the notes, as some nurses may ask for chief complaint first, some might start from previous medical history and some might ask when did the symptoms occur first before asking for chief complaint if symptoms are visually identifiable, etc. Learning sequence from both directions can capture relations between keywords across text. Finally, one more fully connected layer is used to embed words to capture higher order nurses’ notes embeddings. It should be noted that without using this dense layer and using only bi-RNN embeddings, model persistently yielded poorer results. Last two layers in text embedding block have an unchanged embedding size $d_m$ in our experiments.

**Attention block**  The output of previous layers is a matrix $h_n$ of size $l_n \times d_m$, where we embed every word in a $d_m$ dimensional space. It is desirable to compress this matrix representation into a single vector to make it easy to build a loss function and facilitate training.

This can be achieved in multiple ways, which we refer to as pooling, i.e. one can take the sum of all vectors in the document ($\sum_i h_n^{(i)}$) for sum pooling, take average ($\frac{1}{l_n} \sum_i h_n^{(i)}$) for average pooling or take maximum ($\max(h_n^{(1)}, \ldots, h_n^{(l_n)})$) for max pooling. However,
these pooling strategies do not take into account “importance” of different words, where it is desired to give more weight to words that are more important (or provide more information) in the text. In our experiments sum-pooling was always the best performing strategy and we will use it as a baseline.

To evaluate the importance of different words in document, we adopt the techniques from machine translation, namely sequence to sequence learning (Bahdanau et al., 2014) and adapt them to a more general case where compact representations are needed (Zhai et al., 2016). The advantage of attention models, is that word attention scores $a^{(i)}$ are generated dynamically based on the given context, and as such are independently obtained for each document. These scores are obtained using a separate neural network architecture $s(h_n^{(i)}; \theta)$ that simply learns to score words based on their embeddings using softmax function:

$$a^{(i)} = \frac{\exp(s(h_n^{(i)}; \theta))}{\sum_{i=1}^{l_n} \exp(s(h_n^{(i)}; \theta))}.$$  \hspace{1cm} (4.1)

Neural network $s(h_n^{(i)}; \theta)$ (two fully connected layers with ReLU nonlinearities in our experiments) learns real valued scores, normalized across the document, given document representation $h_n$. Learning embeddings is coupled with the entire network in our model, allowing for an end-to-end training. The final vector projection of nurses’ notes is then obtained as $v_n = \sum_i a^{(i)} \ast h_n^{(i)}$.

Finally, learned attentions allow the model to focus on more important words, as we will evaluate in Section 4.2.5. This enables interpretability of the scores provided by the model, which is mandatory for decision making in the emergency departments.

**Final block** Obtained vector representation of the notes $v_n$ is then fed into three fully connected layers with ReLU activations as shown in Figure 4.3. Final layer is used for scoring of classes and sigmoid function is used for binary classification and softmax for multiclass classification, which provide scores $p_n$. 

129
We choose logistic loss function to optimize the model:

\[ \mathcal{L} = \sum_{i \in |C|} \left( \sum_{n : \xi_n^i = 1} \log p_n^i + \sum_{n : \xi_n^i = 0} \log (1 - p_n^i) \right), \]  

(4.2)

where \(|C|\) is the set of possible classes that depends on whether we want to split patients into two cohorts of higher and lower acuity patients (binary classification) or want to predict actual resource category (multiclass classification).

Adding handcrafted features As mentioned before, triage systems collect vast amount of structured data \(p_n^s\) as well. To incorporate such data into our model, we append vector of features collected for the patient to the final layer of embeddings. This creates a deep and wide (Cheng et al., 2016) architecture of the DAM model.

4.2.4 Experiments

Here we show evaluations of the proposed approach. We first present how we construct the data set to conduct the experiments, describe different baselines that are relevant for this study, and provide parameters used in our experiments. We then dive into research questions raised in the introduction and discuss the results obtained from the related experiments.

Experimental set-up

Triage Data

The data used in this study is retrospective review of ED data over a 3 year period 2012–2015 at an urban academic medical center. Routinely available continuous and nominal data includes: ED assigned location, gender, age range, method of arrival, hour of arrival, number of prior ED visits, insurance group, heart rate, systolic blood pressure, and temperature. Data are binarized to create binary vector \(p_n\). Text data included chief complaint, past medical history, medication list, and free text initial nursing assessment. The
response variable represents category of number of resources (0-5) for multi-class classification task, and it is binarized as positive class for the patients who consumed 3, 4 or 5 resources vs. those who consumed 0 through 2 resource categories. The training population in our experiments consists 250,000 ED visits, while 20,500 patient cases were extracted as a validation set. The test set is comprised of 68,500 patients out of which 36,883 are ESI level 3.

**Baselines**

In the experiments we evaluate quality of predictions when using different data sources. Thus, our first choices of baselines are the logistic regression (LogReg) and the multi-layer perceptron (MLP) models that use only handcrafted structured features.

In order to learn features from text data, we employ a basic word embedding model (embed) where we learn word embeddings through classification framework in an end-to-end manner. As suggested in the literature (Jagannatha and Yu, 2016) we compare to bi-LSTM model which was successfully applied for analysis of medical texts in the past. For a representative of very deep text models we employ the word-level VDCNN model described in Section 4.2.2. Finally, to evaluate improvements of attention layer we employ sum pooling on our model (annotated as DSMP) as discussed in Section 4.2.3. All deep models are evaluated with and without using handcrafted features to investigate whether using them provides lift in accuracy.

**Experimental platform**

We implemented all the algorithms using Tensorflow platform and run them on a machine with two Nvidia Tesla P100 GPUs with 16 GB RAM and Intel Xeon CPU with 512GB RAM. Adam optimizer is used to minimize loss, with 0.001 starting learning rate, and batch size of 512 examples. We use a held out validation set to monitor the training progress, and all the models are trained till the validation loss stops decreasing. Dimen-
Table 4.1: Comparison of models that utilize only structured data, against DAM models trained on only unstructured, and on both types of input data.

<table>
<thead>
<tr>
<th>Models</th>
<th>Binary</th>
<th>Multi-class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.</td>
<td>AUC</td>
</tr>
<tr>
<td>LogReg</td>
<td>54.91%</td>
<td>0.5277</td>
</tr>
<tr>
<td>MLP</td>
<td>56.13%</td>
<td>0.5689</td>
</tr>
<tr>
<td>DAM-(P_u)</td>
<td>79.25%</td>
<td>0.8763</td>
</tr>
<tr>
<td>DAM-(P_uP_s)</td>
<td>79.21%</td>
<td>0.8797</td>
</tr>
</tbody>
</table>

The dimensionality of notes word embeddings is set to 300, while for bi-LSTM, first and final linear projection embeddings is set to 200. For VDCNN model, we used 64 filters in convolutional layers.

4.2.5 Experimental Results

In this section we report the performance of the proposed and baseline approaches for the number of resources category prediction tasks as binary and multi-class classification problems in terms of ROC AUC and accuracy for all the models. We aim to answer the following research questions:

Research Question 1:

*Can we automatically learn representations from the nurses’ notes textual content, without any feature handcrafting, in order to predict the number of resources category? Are additional structured features helpful for this task?*

Our objective is to test to what extent the text data representations learned by the introduced deep models, are effective for the number of resources category prediction task and how they compare to the models that use only structured data. In addition, we examine how they perform when using both structured and text data.

We compare learning models that can be implemented using the existing structured data, and models capable of utilizing additional unstructured text data. Table 4.1 shows that the DAM outperforms LogReg (AUC 53% for binary and 50% for multi-class), as well
as MLP model (AUC 57% for binary and 50% for multi-class) learned on structured data only. From the results, we see that DAM was able to learn representation from the triage notes textual content, and predict the number of resources category much better (AUC 88% for binary and 67% for multi-class classification task) than the models that learned from structured data only (31% and 17% lift in AUC, respectively). Even though representations learned from text bring majority of predictive performance improvement, we note, however, that DAM model learned on both text data and structured data performed slightly better than DAM model learned on text data only, showing that both of the data sources are informative and should be used in decision making.

**Research Question 2:**

*How do the DAM models perform for the task of identifying more resource-intensive patients vs. less? How do they compare to the baseline models on this binary classification task (more than 2 resources need as positive class or 2 or less resources needed as negative class)?*

To answer this question we compare our proposed DAM model with baselines able to learn from both structured and unstructured data simultaneously. Table 4.2 presents the predictive performance on binary classification task, and it can be seen that our proposed DAM outperformed the alternatives. Slightly less accurate is the DSMP model, which is the version of DAM where the attention block is replaced with the sum pooling strategy. Even though attention block appears to bring a small improvement in generalization ability, its main benefit is that it introduces interpretability, which we will investigate in research question 5. Primary baselines VDCNN and bi-LSTM models are few percent less accurate than the DAM, while linear embedding approach performs considerably worse.
Table 4.2: DAM vs baselines performances for the binary classification task (more than 2 resources, or less).

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
<th>ROC AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$p_u$</td>
<td>$(p_u,p_s)$</td>
</tr>
<tr>
<td>embd</td>
<td>55.30%</td>
<td>64.33%</td>
</tr>
<tr>
<td>bi-LSTM</td>
<td>76.59%</td>
<td>77.08%</td>
</tr>
<tr>
<td>VDCNN</td>
<td>76.81%</td>
<td>77.70%</td>
</tr>
<tr>
<td>DSMP</td>
<td>78.79%</td>
<td>78.63%</td>
</tr>
<tr>
<td>DAM</td>
<td>79.25%</td>
<td>79.21%</td>
</tr>
</tbody>
</table>

*Research Question 3:*

*How do the DAM models perform for the task of exact number of resources category prediction? How do they compare to the baseline models on this multi-class classification task (classes are number of resources category - 0,1,2,3,4,5)?*

We evaluate all models on the task of multiclass classification, where classes are number of resources used by the patient (where 5 or more resources are assigned to the same category). Results are provided in Table 4.3, with accuracy and average AUC (averaged over one-vs-all evaluation approach). As in the binary case, we observe that DAM is the best performing model with second best model is sum pooling alternative to attention layer. bi-LSTM and VDCNN approaches are best performing baselines, VDCNN being slightly better, and are consistently outperformed by the DAM. Furthermore, we can see that attention persistently provides improvement over next best pooling strategy, yielding best performing model that is also interpretable. As in the binary case, we can see that using structured data stably helps in improving prediction quality.

We next examine how does the DAM model compare to human performance on multiclass classification, as this is the exact task triage nurses are ask to perform (in the step III).
Table 4.3: DAM vs baselines performances for the multi-class classification task (number of resource category).

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
<th>Average AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$p_u$</td>
<td>$(p_u,p_s)$</td>
</tr>
<tr>
<td>Embd</td>
<td>14.74%</td>
<td>14.74%</td>
</tr>
<tr>
<td>bi-LSTM</td>
<td>39.16%</td>
<td>38.50%</td>
</tr>
<tr>
<td>VDCNN</td>
<td>39.68%</td>
<td>41.33%</td>
</tr>
<tr>
<td>DSMP</td>
<td>39.61%</td>
<td>40.67%</td>
</tr>
<tr>
<td>DAM</td>
<td>43.30%</td>
<td>43.80%</td>
</tr>
</tbody>
</table>

**Research Question 4:**

*How does the DAM model performance compare to nurses’ performance in the number of resources category prediction task?*

As ESI levels represent a surrogate for actual number of resources category, from the available data we can only approximate nurses’ performance on this task. We approximate nurses’ prediction accuracy by analyzing how many patients fall in a particular (*First Acuity Level, Number of Resources Category*) group. This is a reasonable approach to evaluate their performance because of the way ESI levels are assigned (Figure 4.1). Namely, in the step III nurses should predict how many resources a patient would require, and assign that patient to ESI Level 5 if it doesn’t require any resources, Level 4 if they require 1 resource, Level 3 if they require more resources, or consider Level 2 if the vitals are in danger zone (and they might require even more resources). Having this procedure in mind, we group number of resources categories to match ESI acuity levels as in Table 4.4.

Table 4.4: Approximating number of resources category by first acuity level prediction

<table>
<thead>
<tr>
<th>Number of Resources Category</th>
<th>First Acuity Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Level 5</td>
</tr>
<tr>
<td>1</td>
<td>Level 4</td>
</tr>
<tr>
<td>2 or 3</td>
<td>Level 3</td>
</tr>
<tr>
<td>4 or 5</td>
<td>Level 2</td>
</tr>
</tbody>
</table>

Then we approximate the nurse performance for the task of number of resources category prediction via their first acuity level assessment. Table 4.5 presents these measure-
ments on the whole dataset. We observe that majority (55%) of the patients were predicted to fall within Acuity Level 3 category, even though they needed only 1 (10%) or even 0 (8%) resources, and that only 10% of patients that did not need any ED resources (recall=0.10) were assigned Level 5 category. On the other hand, patients that required 4 or more resources (14.5%) were possibly misplaced within lower category as they required more resources than most of the patients within Level 3 category and should have higher priority. All of the above leads to total approximated accuracy of 43.6%.

By using the DAM algorithm, we were able to reduce off-diagonal patient counts (Table 4.6). The distribution of predicted resources groups better matches true distribution of the test data (ie. 15% to 16% for 0 resources, 20% to 21% for 1 resource, 36% to 32% for 2 or 3 resources and 29% to 31% for 4 or more resources) giving us the total accuracy of ~ 60% and a potential lift of 16% in accuracy compared to nurses’ assessments.
Research Question 5:

*Can the proposed DAM model be leveraged to improve the existing triage system flows?*  
*How can the attention weights be utilized by medical practitioners to understand predictions?*

Finally, we provide comments on how a system like this can be implemented in actual hospital emergency department systems. The ability of the DAM to discriminate high from lower priority patients as well as to accurately predict number of resources a patient will need allows it to be used for improving patient disposition in the waiting rooms. Such tool can aid triage nurses and doctors to prioritize patients accurately and with minimal bias, especially in uncertain cases, providing the service to the patients that need it the most. In order for computer assisted triage to be accepted by experienced triage staff, having simple estimation without useful clinical feedback will make its deployment more difficult. Thus, being able to provide some intuition on how was the estimation obtained is mandatory. In the DAM, attentions learned for words in notes can act as a proxy for such intuition. Higher weights can tell triage staff what was factored in the given prediction, thus allowing control over potential errors model can make. Ultimately, if everything is rendered satisfying by the staff, actionable decision can then be taken. To evaluate attentions of the DAM, we generate several examples of patients complaint and run trained algorithm to evaluate the attentions (Figure 4.4). We can see that model is capable of focusing on the key words of patients complaint such as: “vehicle “collision”, “pain” and “gunshot”, “chest pain”, “syncopal”, etc., while assigning less of attention to other keywords that might be repeating: “mvc”, which stands for motor vehicle, or non-critical like “pt”, which is abbreviation for patient. It is difficult to properly quantify the quality of obtained attentions, however in the deployed system, triage staff can be allowed to grade attentions for each case thus allowing for supervision in retraining model to obtain higher quality attention mechanism. This will be pursued in the future work.
4.2.6 Conclusions

In this study we addressed the problem of high variance subjective resource utilization outcomes prediction in triage rooms. For this task we show that utilizing nurses’ notes can provide a significant improvement in accuracy compared to standard continual and nominal data. We proposed a novel model to exploit medical texts and obtain state-of-the-art predictive accuracy, finally outperforming reported accuracies of triage staff. Attentions
the proposed model learns can be very useful in providing clear feedback on what guided the predictions aiding interpretability and clinical acceptance of the model. We further aim to address several more issues that nurses’ notes data have, such are common typos, big variety of abbreviations and human bias.
CHAPTER 5

CONCLUSIONS

To conclude, in this thesis we proposed several approaches that tackle predictive model facets that go beyond predictive quality, such are predictive uncertainty and interpretability. We first proposed an extension to the powerful probabilistic graphical model GCRF, discussed its bias problem and problem of estimating the overblown aleatoric uncertainty and proposed a solution for it and we evaluated the model on the single-step-ahead prediction task. We then discussed a potential way of incorporating the epistemic uncertainty in the model, and evaluated it on the multi-steps-ahead prediction task. In the second chapter we proposed novel approaches for learning distributed representations of medical concepts that are based on highly successful advances in the field of natural language processing. We have demonstrated how these distributed representations can be of high quality as evaluated on a series of tasks, such is phenotyping, gene-disease association discovery, disease type discovery, and predicting hospital quality indicators when representing patients visits. We have extensively analyzed the interpretability of obtained representation by examining discovered phenotypes or group of associated diseases as given by the model with medical literature. Results were showing that the model is capable of capturing underlying disease mechanisms that are not provided in the data simply from disease co-occurrences in elec-
Electronic health records. Finally, in the third chapter we proposed a novel deep architecture for analyzing free text and nominal data collected during the triage process in emergency departments. The proposed approach is based on the attention mechanism which provides facet of interpretability while providing predictions of patient’s acuity or number of emergency department resources needed to treat the patient.

In addition to modeling predictive uncertainty and interpretability, all the proposed models were compared against several alternative models, under a variety of scenarios, manifesting surpassing performance over its alternatives.


Dokic, T., Dehghanian, P., Chen, P.-C., Kezunovic, M., Medina-Cetina, Z., Stojanovic, J.,
down due to Lightning and Sever Weather,” in *Hawaii International Conference on System Sciences, HICSS-49*.

learning.”


human metabolic network topology for disease comorbidity,” *Proc. of the Nat. Acad. of Sc. of the U.S.A., PNAS*.


of genomics on drug discovery and clinical medicine,” *QJM*, 93, 391–423.


pregnancy,” *Critical care medicine*, 33, S286–S293.

and reporting,” *Baltimore, CMS and NCHS, 2008Centers for Medicare and Medicaid Services (CMS), the National Center for Health Statistics (NCHS), Baltimore CMS and NCHS*.

Ghalwash, M. F., Radosavljevic, V., and Obradovic, Z. (2014), “Utilizing temporal pat-
terns for estimating uncertainty in interpretable early decision making,” in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 402–411, ACM.

(DIAMOnD) algorithm derived from a systematic analysis of connectivity patterns of

Health Sci.
Gilboy, N., Travers, D., and Wuerz, R. (1999), “Re-evaluating triage in the new millen- 
nium: a comprehensive look at the need for standardization and quality,” *Journal of 

Index (ESI): a triage tool for emergency department care, version 4,” *Implementation 
handbook*, pp. 12–0014.


and application to iterative multiple-step ahead forecasting of time-series,” in *Switching 
and Learning in Feedback Systems*, Springer.


154


