A Probabilistic Graphical Model for Structured Prediction over Heterogeneous Data

Author:
Ye, Pengjie

Publication Date:
2017

DOI:
https://doi.org/10.26190/unsworks/19920

A Probabilistic Graphical Model for Structured Prediction over Heterogeneous Data

by

Pengjie Ye

B.Eng. Computer Software,
Tsinghua University, Beijing, China.

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Faculty of Engineering
School of Computer Science and Engineering
The University of New South Wales

September 2017

Abstract

Advances in sensor and instrumentation technology, together with cost reductions and capacity increases in computing and communication technologies, have led to the rapid accumulation of large amounts of data, additional to that collected by traditional methods. These sources form data called *heterogeneous* since it does not conform to a single type of data structure. A notable example is Electronic Health Record (EHR) data. Given the size and complexity of heterogeneous data there is a growing need to apply machine learning to predict, for example, patient outcomes from EHR data. Such data is inherently uncertain, so learning algorithms based on the framework of probabilistic graphical models for classification are appropriate.

Despite the popularity of structured prediction, its capability in utilising domain knowledge and modelling on the source of structure is limited. This thesis identifies the connection between the mechanism of abstract domain knowledge and the structural setting of a graphical model. A clique-based mapping method is proposed to develop a structural-binding and knowledge embedding set of feature functions.

A general discriminatively-trained probabilistic graphical model, the transitional random field (TRF), is proposed for modelling heterogeneous input data without the locality preserving property, which is widely seen in conditional random field(CRF) problem settings.

We also introduce a novel ontology-based probabilistic similarity measurement for heterogeneous data which simplifies probabilistic computation in TRFs and enables efficient inference. The TRF framework identifies and maps information from the input structure to the non-isomorphic format determined by the output structure, while at the same time utilising structurally embedded existing knowledge implicit in the structure of the input and output. This ability to represent dependencies as features denoting transitional relations between input and output gives TRF the potential to learn models from a wide range of heterogeneous data and make predictions about structured domain knowledge.

Our experiments on a large real-world data set demonstrate that TRF can be successfully applied to a demanding structured prediction problem over heterogeneous EHR data, with the proposed TRF training and inference algorithms obtaining good accuracy and efficiency.

Publications

Parts of the work presented in Chapter 2 were published in:

[1] P. Ye, Y. Xie, Q. Zhang, and C. Pang, "Predicting Stroke Outcomes from Physiological Patterns" in COINFO-2012 (2012).

[2] Q. Zhang, Y. Xie, P. Ye, and C. Pang, "Acute ischaemic stroke prediction from physiological time series patterns", in AIH2012 (2012), pp. 45–54.

[3] Q. Zhang, Y. Xie, P. Ye, and C. Pang, "Acute ischaemic stroke prediction from physiological time series patterns", Australas. Med. J., vol. 6, no. 5, pp. 280–6, Jan. 2013.

A separate publication on related work conducted during the course of this research is:

[4] Zhang, Q. and Ye, P. and Lin, X. and Zhang, Y. "Skyline probability over uncertain preferences" in Proceedings of the 16th International Conference on Extending Database Technology, 2013, pp. 395–405.

Contents

List of Figures

List of Tables

My sincere gratitude
to my supervisor Mike Bain
for your wisdom,
to my wife
for your love,
to my parents
for your unconditional support.

1 Introduction

The digitalization and development of medical equipment of life-care facilities in hospitals brings about possibilities of generating and collecting a huge volume of heterogeneous Electronic Health Record (EHR) data. The development and spread of wearable electronic devices could also provide possibilities in collecting high-density and everyday health data for a much larger population. Notably, the greatly enhanced data availability to computer systems may well exceed the level that unaided doctors can possibly have.

Under the natural assumption that these manageable data pools are extremely information-rich, machine-based analysis and prediction techniques are in great need to automate the diagnosis and prediction process. However, the current techniques for handling the heterogeneous input data and utilising the domain knowledge are far from sufficient, particularly on structured prediction model-related techniques.

The strong belief on the potential of applying state-of-the-art machine learning techniques to EHR prediction problems motivates us to develop novel techniques in this area. In this thesis, we systematically illustrate the novel ontology-assisted structured status prediction framework we developed for successfully making structured prediction over the overall health status of a patient, as will be examined in later chapters[1].

The main contributions are:

[1]Note that in this thesis we use the term "status" rather than the more usual "state" to avoid the possibility of confusion with the class of probabilistic graphical models known as "state-space models". We fully define the meaning of the term status in Chapter 3.

- exploratory data analysis and prediction learning on health record data (Chapter 2)
- ontology representation and domain knowledge embedding for structured prediction models (Chapter 3)
- heterogeneous input data modelling in structured prediction models (Chapter 4)
- an analysis of the *discriminative* learning approach to probabilistic graphical models highlighting the implicit structure on *both* the input and output components of the models (Chapter 4)
- the introduction of a new form of discriminatively learned probabilistic graphical model, the Transitional Random Field (TRF), that relaxes this implicit structural restriction (Chapter 4)
- in terms of inference for TRFs, a key insight is how to relax the implicit *locality* constraint of Conditional Random Fields (CRFs) during inference, by making use in TRFs of available structure in the data in a *general* way (Chapter 5)
- the derivation of a new training algorithm for TRFs, based on similarity, which avoids the computation of complex partition functions (Chapter 5)
- the implementation of an algorithm to learn TRFs using this framework (Chapter 6)
- results from an application of this algorithm to the challenging task of structural prediction of codes from the standardised medical International Classification of Diseases (ICD) ontology on a real sample of heterogeneous EHR data (Chapter 6).

The rest of this chapter is organised as follows. We first describe the characteristics of EHR data in the real-world healthcare environment and then discuss our motivation relative to this type of data. After examining the related methodologies for prediction, we then formalize the prediction problem for EHR data in the context of random field theories. We develop our machine learning techniques with special effort on the heterogeneous input data modelling. We cover different aspects of this ambitious structured prediction problem in corresponding chapters throughout this thesis.

1.1 The Heterogeneous Electronic Health Record (EHR) Data

We are in an era of data explosion, especially in healthcare-related fields [Gar13]. Hospitals and other healthcare organizations are keen to advanced high-tech equipment to improve their healthcare quality. Heavy investments have been made in the recent years by both the resourceful public and private sectors to deliver their vision, whereas the practitioners in the healthcare industry tend to be more conservative to sticking to the way they have been used to during the long time before the digital age.

While expensive digitalised equipment is continuously contributing huge amounts of high density data, the paper-pen weaponed humans are still often required to provide information for medical records. It makes sense, though, for the data collector to get every piece of information both from the highly automatic computer-based devices, such as are found in Intensive Care Units (ICUs) as well as the old school golden-age machines, such as thermometers, and even from natural language, e.g., as clinical narrative. The different types of output and various data sources make the resulting EHR data almost always heterogeneous.

Researchers from different sectors have realised that these information-rich data are forming a new research base which could result in deep impacts on many industries, including healthcare, financial and education, etc. [MD13, HY14, KKV13]. Governments of several countries have been dedicated to building centralised systems to store the electronic health record (EHR) data, in a way that is comprehensive as well as exchangeable [GT05, SBH$^+$07].

Despite the fast speed of deploying the infrastructure for EHR data, the ability to extract information from these heterogeneous observations is limited [OLSH12, HY14]. Thus, a novel machine learning framework is needed to model the latent structure behind the data and to further exploit the potential ability for structured prediction, given relevant human knowledge as the prior.

1.2 Motivation

Structure, as an abstraction of relationships, plays a vital role in the whole process of learning, both by humans and computers. The human brain uses structures to describe the characteristics of perceived information, and later uses them to understand new signals. Similarly, machine learning algorithms use structures to build the learning model

and, in some cases, uses complicated structures to describe the output of prediction. We are motivated to develop novel machine learning techniques that can fully utilise the embedded information from structure.

1.2.1 The Absence of Clear Observation Structure

This learning process becomes much more difficult when it is not possible to use one or a small number of well-defined structures to describe the observations. The reason could be that the observations may have many heterogeneous sub-structures from different logical perspectives, so an overall structure is unable to capture most of the structural information. The reason could also be that the observation does not have any structure that could be formally described by the current mathematical modelling language at all.

Such heterogeneous observations not only bring difficulties to the learning process but also present challenges to the modelling of the output of the predictions. Heterogeneous data encodes a lot of information, both structurally and semantically, but it is often the case that we are not even sure about what to predict to fully reveal the information latent in the observation data.

1.2.2 The Non-Isomorphic Output Structure

Many graphical models have been studied extensively in the machine learning community to model the dependencies among factors behind observations. Similar structures which are isomorphic to the dependency graph, or its elementary subgraph, naturally become the output of the prediction. The similarities between structures of the latent variables in the observations, and the structures of the prediction output, benefit the training and inference algorithms greatly because these two can be represented by the same or very similar structures.

However, the homogeneity of the mappings between the latent structure behind the observations and the outputs of prediction can also greatly limit the extent to which we can learn from the data, and the expressive power of the output. Thus, for machine learning practitioners it would be more desirable to model the outcome of prediction as a rigid structure with a full set of prior information from existing knowledge and then learn from the heterogeneous latent structure in the observations. Note that this process is actually a transition from a heterogeneous structure to a well-defined

rigid one for the output, which allows much more potential to make use of available knowledge.

1.2.3 The Target Problem in General

Any EHR prediction algorithm will need to solve the structure transition problem discussed above because (1) of the heterogeneous observations making up the input data, and (2) it will need at the same time to make full use of the dependency information embedded in available output structure.

Formally, the general target problem can be described as: Let $G = (V, E, I)$ be a graph structure such that the output random variables $\mathbf{Y} = (\mathbf{Y}_v)_{v \in V}$, so that $\mathbf{Y}$ is indexed by the vertices of G. I is the invariant knowledge embedded in G and E is the set of edges of G. Find a target model M such that given the observation random variables $\mathbf{X}$, M is able to calculate the distribution $\Pr(\mathbf{Y}|\mathbf{X}, G)$.

1.3 Temporal Modelling

1.3.1 Stochastic Process

The following is a definition of a stochastic process: Given a probability space $(\Omega, \mathcal{F}, P)$ and a measurable space (S, Σ), an S-valued stochastic process is a collection of S-valued random variables on Ω, indexed by a totally ordered set T ("time"). That is, a stochastic process X is a collection $\{X_t : t \in T\}$, where each X_t is an S-valued random variable on Ω. The space S is then called the state space of the process.

This is a strictly defined type of random variable sequence, rather than a probability distribution model. However, it is difficult to model heterogeneous input data with the state space S shared among $\{X_t : t \in T\}$.

1.3.1.1 Problems with the Gaussian Process Random Variables and the Feature Functions

The assumption that a feature function's value is a random variable which draws values from a Gaussian distribution or is a sum of a subset of such random variables is often too strong for applications where very little can be assumed. In most cases of structured prediction, the feature functions are often little more than indicator functions, which

means (1) these functions are often not continuous, and (2) we have very limited prior knowledge about the possible connection between the feature appearing in the input data and the corresponding change in the distribution over the domain-specific ontology underlying the output structure.

1.3.2 State Space Models

A state space model (SSM) [Mur12] is an extension to an HMM, with continuous hidden states. The temporal relation between $\mathbf{z_t}$ and the previous hidden state $\mathbf{z_{t-1}}$ is modelled as:

$$\mathbf{z}_t = g(\mathbf{u}_t, \mathbf{z}_{t-1}, \boldsymbol{\epsilon}_t)$$

The current observation $\mathbf{y_t}$ is related to the current hidden state $\mathbf{z_t}$ by:

$$\mathbf{y}_t = h(\mathbf{z}_t, \mathbf{u}_t, \boldsymbol{\delta}_t)$$

where $\mathbf{z}_t$ is the hidden state; $\mathbf{u}_t$ is an optional input or control signal; $\mathbf{y}_t$ is the observation; g is the transition model; h is the observation model; $\boldsymbol{\epsilon}_t$ is the system noise at time t; $\boldsymbol{\delta}_t$ is the observation noise at time t; $\boldsymbol{\theta}_t = (\mathbf{A}_t, \mathbf{B}_t, \mathbf{C}_t, \mathbf{D}_t, \mathbf{Q}_t, \mathbf{R}_t)$ are parameters of the model, which becomes stationary if $\boldsymbol{\theta}_t$ is independent of time.

Clearly the model setting is very similar to an HMM, yet with more flexibility in describing continuous valued hidden states $\{\mathbf{z}_t\}$ and observations $\{\mathbf{y}_t\}$. Nevertheless, the hidden state dependencies are restricted to the pairwise ones between $\mathbf{z}_{t-1}$ and $\mathbf{z}_t$. Similarly, the current observation $\mathbf{y}_t$ is only dependent on variables for the current time t ($\mathbf{z}_t$, $\mathbf{u}_t$ and $\boldsymbol{\delta}_t$), without further direct modelling of the dependencies between $\mathbf{y}_t$ and $\mathbf{y}_{1:t-1}$ or $\mathbf{z}_{1:t-1}$ ($t > 1$).

The prediction target for SSM is the probability distribution of the belief state $p(\mathbf{z}_t|\mathbf{y}_{1:t}, \mathbf{u}_{1:t}, \boldsymbol{\theta})$. Thus, one direct approach for this is to make use of existing continuous distributions. One of the major cases used in real world applications is the linear-Gaussian SSM (LG-SSM), or linear dynamical systems (LDS).

An SSM is a LG-SSM if:

- the transition model is a linear function: $\mathbf{z}_t = \mathbf{A}_t\mathbf{z}_{t-1} + \mathbf{B}_t\mathbf{u}_t + \boldsymbol{\epsilon}_t$
- the observation model is a linear function: $\mathbf{y}_t = \mathbf{C}_t\mathbf{z}_t + \mathbf{D}_t\mathbf{u}_t + \boldsymbol{\delta}_t$

- The system noise is Gaussian: $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t)$
- The observation noise is Gaussian: $\boldsymbol{\delta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t)$

The LG-SSM has many applications in the real world because it supports exact inference. For more details on the training and inference related algorithms see [Mur12].

1.4 Heterogeneous Input Data

Heterogeneous input data brings extra challenges to structured prediction models. Normally no fixed vocabulary can be assumed for this kind of data. Thus, it is very difficult to model the dependencies between different observed variables in the data, even if we know how to define them.

Moreover, due to the lack of structural information, it is often impossible to know the mapping relation between the local features in the observation and those on the latent variable side. Thus, *locality* cannot be preserved. In that sense, any sub-graph of the output structure is also conditioned on the global observation. Consequently, the ability to capture complex features is a necessity, e.g., features defined over relatively large sub-graphs. Thus, the resulting structured prediction model should be able to handle large cliques.

1.5 Ontology in Structured Prediction

Ontology as a representation of the domain knowledge has great potential in assisting structured prediction. This is because the output structure is generally a description of the belief state for the observation. Thus, an ontology could enhance structured prediction methodologies in two ways:

1. Ontology provides a comprehensive concept vocabulary for describing the status implied by the current observation; and

2. A structured prediction model could increase the effectiveness of the framework by incorporating the domain knowledge embedded in an ontology, e.g., co-occurrence indicators that are hard-wired into feature functions.

1.5.1 Ontology as Domain Knowledge

Ontology is a standard approach to the definition, organisation and, often, formalisation of domain knowledge. An ontology is a "specification of a conceptualization" [Gru93], where a conceptualization is the set of objects and relations defined on them in a domain [GN87]. Currently there is a move to standardize widely-used domain knowledge structures, such as the medical terminologies used in electronic health record data, as formal ontologies.

1.5.2 Information Transition with Ontology

Ontology is important to structured prediction in terms of information mapping. The general assumption is that there is useful information residing in the "raw" and sometimes heterogeneous input data. Thus, the learning and prediction process in ontology-based structured prediction is actually an information transition from the "raw" form in heterogeneous data to a converted and regularised form given by a rigid ontology structure. This requires that the concepts together with their interdependencies in an ontology can be properly represented and such structural information takes part in the learning and predicting process.

The potential of ontology has not yet been fully addressed in the machine learning community, except for in some specific applications, e.g., sentiment analysis or classification and user profiling in social networks [FPM16, GBH09, Jan16, OLC+16, BHML16].

1.5.3 Knowledge Graphs

Essentially, an ontology represents structured knowledge in a specific domain. Knowledge graphs are multi-relational structures that have attracted interest from many communities, particularly industry, e.g. Google [Sin12]. Discovering facts about entities and constructing structured knowledge bases, on the other hand, is a major goal of information extraction and knowledge base population technologies [BEP+08, JG11, RCR16]. For more details of some initial work on domain-specific information extraction, see [Mil95, RLM+06]. Due to the extremely rapid growth in the amount of text-based information available on the web, there is a strong need for gathering and organising knowledge for further storage and query. The concept of knowledge graph was proposed in [Sin12].

Although both knowledge graph and ontology consist of a group of concepts and their relations, the emphasis of each is different. Firstly, knowledge graph related techniques address the problem of how to gather information and build the connectivity for the graph, while ontologies are normally and ideally built up by and for professionals with existing domain knowledge. Thus, the form and source of relations are normally different. Secondly, the structure of an ontology is often more hierarchical, with relatively clearer divisions between different levels of abstraction. Knowledge graphs, on the other hand, often have fairly simple structures with various logical relations between entities.

A knowledge graph is a symbolic and logical system and many prediction tasks on it involve aggregating global knowledge over it, thus computational efficiency is often the major concern. A new trend of techniques named embedding, aiming at solving this type of problems, is to project the high-dimensional space to a low-dimensional continuous vector space, while preserving the needed topological properties [WZFC14b]. More follow-up work on the methodologies of extending a knowledge graph and aggregating information from it can be found in [LLS+15, PMGC13, WZFC14a, Zha02].

1.5.4 Ontology for CRF

1.5.4.1 Validity

Although normally ontology has already the form of structured knowledge, it has to satisfy several requirements from both its semantic setting and the structural setting with respect to target prediction models, such as CRFs.

1. An ontology for CRF needs to be valid in terms of the semantic meaning and relationships of its concepts. Generally speaking, an ontology is valid for being used in a prediction model, e.g., a CRF, only when it is with a valid concept arrangement as an ontology by itself in the first place. This already implies the completeness and accuracy for concepts involved. Nevertheless, the validity also requires the following.

(i) Concepts in an ontology are normally arranged in a hierarchical manner, with different levels of abstractions. There should exist one or several spanning tree(s) whose paths represent the hierarchical relations between concepts, such that the spanning tree(s) provide a complete cover over all the concepts from the ontology.

(ii) The concept arrangement should have an appropriate level of granularity, and semi-continuous changes. More concretely, any change in the position of a concept in the ontology that is determined by the domain knowledge should be consistent with the

graphical position of that concept given by the ontology. In other words, the ontology should be such that position changes along the path of a spanning tree indicate smooth semantic progression, which can be the basis for modelling the adjacencies from a hierarchy in the probabilistic graph model.

2. An ontology for a CRF needs to be a Markov Random Field (MRF). As the output structure of a CRF, the random variable set Y defined by the ontology should obey the Markov property. This will be discussed in more detail later.

Note that as pointed out in [Cli90], constructing random mosaics which are spatially Markov is a challenging problem. However, for the purposes of probabilistic modelling, in this thesis we require only that an ontology to be used as the output structure in a model should be a general graph such that one or more spanning trees may be constructed on it.

1.5.4.2 A Fixed Output Structure for Various Instances

One direct implication from using ontology in a predictive model is that, for any prediction task in the same domain, the output structure is the same for the whole category of structured prediction problems. There is an asymmetry between the output and the unstructured or semi-structured heterogeneous inputs. More specifically, for a particular category of prediction problems in some domain, with a well-defined ontology, the fixed output structure can be used as a common descriptive vocabulary to describe *different* input instances. In that sense, it is complete and adequate to use a fixed ontology-based structure as the basis of induced knowledge to describe different possible heterogeneous inputs. This motivates a robust yet powerful probabilistic framework to solve this prediction problem, while relaxing the local structural mapping relation presented in all previous MRF/CRF-type models. For the formalisation of the local structural relation between for the input and output, see Section 4.4.

1.5.4.3 Random Field Construction for Ontology

If we put the semantic meaning of an ontology aside and only consider the structural information, an ontology[2] can be reduced to a simple graph $G = (V, E)$, where every node V_i in G represents a concept explicitly defined in the specific domain. If we assign a continuous random variable $Y_i \in [0, 1]$ to V_i, representing the confidence value

[2] At least for ontologies from the biomedical domains considered in this thesis; in general this may not apply for ontologies defined using fragments of first-order logic, such as Description Logics.

of V_i to appear in the result set of the current status described by the ontology, the configuration $\mathbf{y}$ of the set of random variables $\mathbf{Y} = \{Y_i\}$ is a complete description of the current status based on the predefined domain-knowledge. Thus, the probability of a configuration $P(\mathbf{y})$ is an important measurement in obtaining the Maximum Likelihood Estimator (MLE).

In order to make the induced random field Markovian, it requires that all the sensible dependencies lead to edges in E. With this completeness in modelling dependencies, the resulting random field has only local dependencies. This confidence value-based setting results in a model with strictly positive distribution and only local dependencies. Thus, the induced graph is an MRF.

1.5.5 Continuous MRF/CRF

Following the description of the continuous confidence distribution of an ontology as the desirable prediction space, in this section we consider MRFs with continuous random variables for modelling the probability of a given distribution.

The Hammersley-Clifford theorem builds the bridge between the Markov property and clique factorisation in a general setting. So, its conclusion is also applicable to cases with continuous output variables.

Continuous random variable output does not bring much change to the evaluation of feature functions. However, it greatly influences the setting and the semantic meanings of feature functions. We fully examine these in Section 4.3.3. Here we first examine the relation between a multivariate Gaussian distribution and its corresponding MRF. Then the previous normalisation scheme is transformed to set up the partition function for the MRF with continuous random variable output.

1.5.5.1 Non-parametric Models

The continuous output model has a clearer joint probability representation when we know the abstract form of the distribution of individual random variables. Given the joint distribution, the dependencies between connected nodes can be inferred.

We take the Gaussian family as an example. Given a multivariate normal distribution, it forms a Markov random field with respect to a graph $G = (V, E)$ if the missing edges

correspond to zeros on the precision matrix:

$$X = (X_v)_{v \in V} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$$

such that

$$(\Sigma^{-1})_{uv} = 0 \quad \text{if} \quad \{u, v\} \notin E$$

For more details about the proof, see [RH05].

A variation to the traditional usage of the feature function is briefly mentioned in [QLZ+09], where the C-CRF model tries to combine the similarity relation and the parent-child relation by adding the parent-child term $R_{i,j}(y_i - y_j)$ and the similarity term $\frac{S_{i,j}}{2}(y_i - y_j)^2$. However, the potential of variational features in inference and relational learning was not identified in this work.

1.5.6 Challenges to Existing Models

Using an ontology as the output structure for a CRF requires a learning framework able to:

1. handle generally or densely connected CRFs, including training and inference
2. break the locality preserving property, as will be fully examined in the corresponding chapters, which results in a "semantic" preserving property
3. perform efficient feature function-based computation with clique size larger than 2, such that the dependencies appearing in an ontology have presence and real influence.

In this thesis the central problem will be seen as that of characterising a general information transition process to enable machine learning, as we have begun to outline above and will be made clear below. The challenges posed by this general information transition process are from both the input and output structures, and their dependencies. We first consider the variations to the output and set up the model in the following Section 1.5.5. The modelling for the input data is discussed in Section 1.4 and Section 3.4.1.2. The general connection between the input and output structure is discussed in Section 4.2.5 and further examined in more details in Section 4.4.

1.6 Example Prediction Problems

In this section, we consider several classic prediction problems which are applicable to be adapted with structured output. By doing this, we demonstrate the possible settings and difficulties of these problems, without loss of generality. Therefore these problems will be used as running examples in this thesis, to illustrate our solutions to all the motivating problems.

1.6.1 Locality Modelling in Applications

In this section, a range of prediction problems are discussed to demonstrate structure settings for the different types of dependencies formalised in Section 4.5 on the input $\mathbf{X}$ and the output $\mathbf{Y}$. The nature of the dependencies, and the way the random variables are inter-dependent, are determining factors for the fundamental structure of the network, and hence the subsequent probabilistic graphical model.

At the end of this section, we identify the dual-locality assumption hiding in the mapping between $\mathbf{X}$ and $\mathbf{Y}$ of almost all structure settings in the literature. Relying on this property has greatly limited the range of problems to which the existing prediction frameworks can be efficiently applied.

1.6.2 MRF Applications

Markov random fields find application in a variety of fields, ranging from computer graphics to computer vision and machine learning. MRFs are used in image processing to generate textures, as they can be used to generate flexible and stochastic image models. In image modelling the task is to find a suitable intensity distribution for a given image, where suitability depends on the kind of task, and MRFs are flexible enough to be used for image and texture synthesis, image compression and restoration, image segmentation, surface reconstruction, image registration, texture synthesis, super-resolution, stereo matching and information retrieval. They can be used to solve various computer vision problems which can be posed as energy minimization problems, or problems where different regions have to be distinguished using a set of discriminating features, within a Markov random field framework, to predict the category of the region. Markov random fields were a generalization over the Ising model and have, since then, been used widely in combinatorial optimizations and networks.

1.6.3 The Sequence Segmenting and Labelling Problem in Text Processing

Text-based sequence segmentation is actually a tagging problem, which has been studied extensively in the natural language processing community. Typical examples are applications in speech recognition/language models [ZN09].

Due to the format of human language, in such applications the input $\mathbf{X}$ and output $\mathbf{Y}$ almost always have a linear-chain structure. In text processing, the input $\mathbf{X}$ normally represents a sentence and $\forall\, \mathbf{X}_i \in \mathbf{X}$ represents a word in $\mathbf{X}$. Thus a configuration $\mathbf{x}$ of $\mathbf{X}$ is a specific sentence and the generating random value $\boldsymbol{x}$ has a one-to-one mapping relation between $\boldsymbol{x}_t$ and $\mathbf{X}_i$.

Similarly the output $\mathbf{Y}$ represents a tag sequence, which is typically seen in the part-of-speech (POS) tagging problem [LMP01]. $\mathbf{y} = \{\mathbf{y}_0, \ldots, \mathbf{y}_T\}$ is a configuration of $\mathbf{Y}$, with each $\mathbf{y}_t$ being the POS tag for the position t in the given input sentence $\mathbf{x}$.

This linear-chain structure in the POS tagging problem is solely based on the temporal dependencies assumed in the text processing scenario, because the index variable in describing the current state/stage is t, which could be an analogy to time.

Although this is a general-purpose probabilistic model, text-based application, e.g., part-of-speech (POS) tagging, is a classic structured prediction task for this discriminative model [MS99].

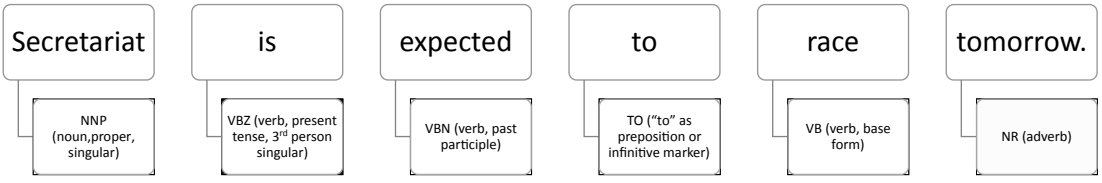

FIGURE 1.1: The linear CRF in POS Tagging

As we can observe in Figure 1.1, the CRF for POS tagging is actually a linear CRF, with both linear input and output structures. Furthermore, there is a strong one-to-one mapping between the two groups of random variables.

1.6.4 The Area Tagging Problem in Computer Vision

CRFs have also found their way to the field of computer vision [NGL10]. Here the edges between nodes of the output random variables are used to model adjacent areas in the figure. Similarly, CRFs can be used to tackle the area tagging problem.

We can also note that, although MRFs are applicable in the computer vision field, the ability to model many other features of the input variables makes CRF a more popular choice.

1.6.5 Specific-Purpose vs. General-Purpose Disease Progression Model

Considering the specific disease progression prediction problem, it could either be focusing on a specific disease (specific-purpose progression model) or on a cohort of diseases (Hawkes process). For coverage of these descriptions see [CBS+16a].

We focus on the general-purpose progression model here.

1.6.5.1 The Basic Problem Setting

The general scheme is to make predictions by leveraging the large amount of historical data in EHR. Thus, the model should be able to handle the longitudinal information within. Intuitively the prediction outcome should be a description of the health status of the patient, similar to the diagnoses and medication orders from a physician in future.

The basic setting for the disease progression prediction model is: given a large set of EHR data for a cohort of patients, predict the most-likely distribution of the confidence values of all the possible diseases as a comprehensive description of the patient's current health status.

The basis of prediction, or the raw input data, is the longitudinal patient records, including biographical information for the patients, admissions, operations and wards movements, etc., together with descriptive codes, e.g. diagnosis codes, medication codes, procedure codes, or even comments from physicians. There is an underlying assumption that, given all the information available, the outcome of a patient's health status is, to some extent, predictable. We hold this to be true because the same argument for the whole development of medical science applies here. Although it is probably optimistic to say that disease progression is understandable by humans, the purpose of developing prediction algorithms is to at least achieve the level of expertise of the best doctor, or consensus of doctors, available. The output of the prediction is the physician diagnosis and medication order of the next visit.

1.6.5.2 Main Challenges

The main challenges for the previous models are, firstly, scalability and secondly, the lack of domain-specific knowledge.

Given the importance of time as an index variable in all the disease progressions, temporal models are a necessity.

A naive but straightforward method is to set up individual classifiers for every possible disease (or ICD code for a disease). However, the hierarchical structure of such disease predictions means strong inter-dependencies and conceptual overlapping, resulting in great difficulties in inferring the result at a fine-grained granularity.

Recurrent neural networks (RNN) are used to solve this problem in [CBS+16a] in an approach called "Doctor AI". However, this method only considers previous ICD codes as input, and thus only makes predictions based on the inertia existing in the ICD progression. Ignoring all the heterogeneous data by only examining the previous ICD codes as input implies a huge information loss. Moreover, purely relying on the inertia of the ICD codes without systematically making use of the domain knowledge makes it difficult to further understand disease progressions. To see this, without considering the structure of ICD codes defined by an ontology, the model in Doctor AI represents the output labels as a plain vector composed of indicators of the occurrence of individual ICD codes.

1.7 Related Literature

1.7.1 Prediction over EHR Data

The prediction problem over healthcare related data is actually not new in the biomedical or informatics literature. Before the boom in data collection from electronics devices, some health-related data was collected by enterprise systems, only in a relatively simpler form. This was partly motivated by proposals for an intelligent knowledge agent system for healthcase predictions. These have yet to accomplish their goal.

The efforts to model medical knowledge and to make basic inference and prediction can be traced back to the days of expert consultant systems and computer-based clinical decision support systems (CDSSs)[WHB+88, HHHS98, WS08, KHBL05, GAM+05]. As a matter of fact, the initial systems were designed with a relatively bold, all-inclusive philosophy. However, it was found that the overall knowledge representation problems

are extremely difficult and, unfortunately, these are the basis of all medical expert systems.

An examples is the expert system designed for education of medical students to practise problem-solving skills where normally a good physician is needed [WHB+88]. The system is supposed to mimic an expert diagnostician, which requires the ability to recognize a selected set of diseases. The relations between some specific disease and all its manifestations are stored as a knowledge frame. This is an interesting attempt to model the relationships between different diseases, because the list-style representation of a densely connected two-dimensional disease graph will result in a memory overflow quickly when the number of diseases increases, especially given the hardware specifications of computers in the 1980s. A hierarchy as a representation of the disease topology was used in this design.

Despite many medical expert systems having been designed (diagnostic systems, reminder systems, disease management systems, drug-dosing or prescribing systems, etc.) and put into the professional working environment[GAM+05], the ability of CDSSs is still fairly limited. They can only demonstrate enhanced clinical performance in a few specific fields, e.g., drug dosing, preventive care, etc. Systems may actually work better in patient matching than making clinical suggestions. Results show that many CDSSs improve practitioner performance. However, the effects on patient outcomes are sometimes inconsistent[GAM+05].

To further extend the ability of CDSSs, a four-phase approach has been developed to further enhance the evolution of clinical decision support architectures [WS08]. Although each of them has been proven to be effective, the authors concluded that there is a common limitation in the knowledge representation system that all the approaches cannot overcome.

Actually, this conclusion applies universally to all the medical diagnosis oriented expert systems from the early days. The proposed ideal performance of the applications cannot be achieved without a comprehensive and accurate knowledge representation mechanism.

Having experienced difficulties in modelling and representing the overall knowledge in general, the approach of predicting one or several specific indicator variables have often been often adopted. Also, statistical association has been preferred to the traditional logical inference based on abstract knowledge. Various machine learning techniques e.g. support vector machines (SVM) and some regression models, etc. have been deployed [WRS10].

The indicator of heart failure in 6 months' was chosen as the output random variable in [WRS10].

The relationships depicted in Figure 1.2 between the indicator variable and a small set of supporting variables of different types were carefully studied by applying the SVM, Boosting, and logistic regression separately.

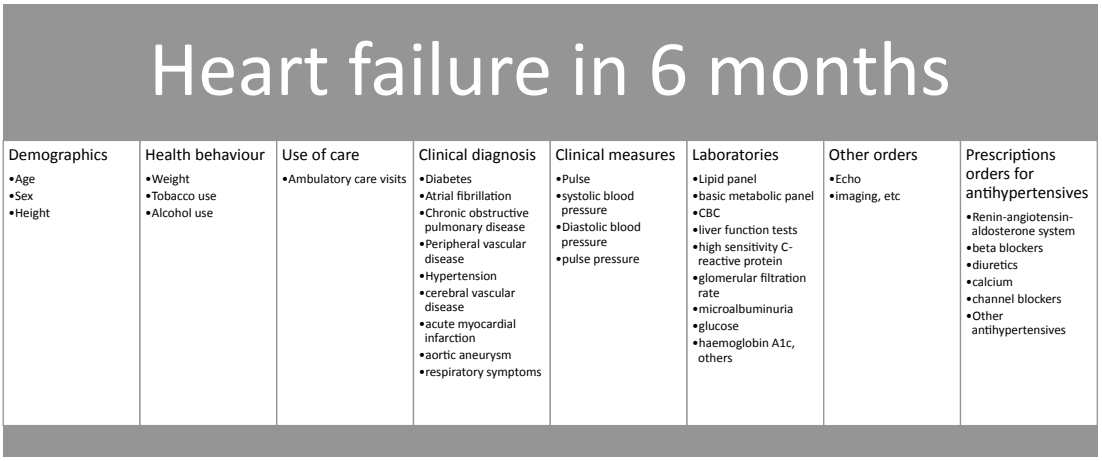

FIGURE 1.2: The heart failure indicator and its supporting variables

There are also several studies in the literature adopting similar structures to Figure 1.2 in order to model the relationships between different indicator variables and other groups of supporting variables, e.g., hospital mortality [vWEGF10, CSC$^+$13, ACZ$^+$13, TSNJ14], readmission[CBWB12, SGM13, GRB13], cardiopulmonary arrest[ACZ$^+$13], and so on.

More advanced generative models have also been devised to study the relationships between different layers of features and temporal patterns in terms of the value change of the indicator variables [CPCC$^+$16]. Results show that more complicated structures such as these can fit the dependency relations of several different indicator variables at the same time, albeit with parameter differences.

1.7.2 The Need for Prior Knowledge in EHR Prediction

The limited power of models based on associative dependencies motivates the need to incorporate more prior-defined domain knowledge to improve EHR prediction. Recall the embedded information I in the structure G as described in subsection 1.2.3: a well-defined medical-ontology graph structure should help to utilize the corresponding domain information.

The ICD-10 code is the 10th revision of the International Classification of Diseases (ICD)[3]. The main part is a tree-like hierarchy to represent all human diseases. Five different types of dependencies among the whole set of nodes are defined to make the hierarchy a general graph. The rigorous structure and rich information in the ICD-10 hierarchy provide a good expressive tool to embed human knowledge about diseases. In this thesis a distribution of well-defined random variables corresponding to the whole hierarchy will be adopted in chapter 4 to give a structure for prediction of a patient's health status.

1.8 The Similarity Problem in the EHR Literature

The first problem when dealing with the similarity between patients is how, based on EHR data, to define and compute the similarity itself. Intuitively, this should be possible based on the current in-hospital electronic health data, as well as the medical history, including the demographic information, for any pair of patients. Based on pair-wise similarity over a large group of patients, finding the most similar patients has great potential in many clinical aspects, for example, for predicting the eventual discharge diagnoses by exploiting similarities between patients along multiple dimensions.

How can such a similarity measure be developed ? One possibility is to apply machine learning. The prospective outcome of similarity measure learning is a similarity function $S(P_i, P_j)$ between any two patients P_i, P_j, which is capable of providing normalized similarity measurements in a metric space.

1.8.1 Problem Definition

The similarity measure that we will introduce later in this thesis will actually provide a way of defining the distance in the heterogeneous data space. Once defined, such a similarity measure can have a number of uses. For example, a typical distance based querying method is the method of k-NN (k-Nearest Neighbours). The particular role of such queries in the EHR scenario would be to find similar patients to a selected patient, which can provide the basis for advice on medical treatments.

[3] http://www.who.int/classifications/icd/en/

1.8.1.1 A Naive Solution

A naive solution is calculating all the pairwise distances or similarities for all the data points with respect to the query point, then picking the top-k nearest neighbours to determine the "peers" of the patient. However, this algorithm is likely to be too inefficient for practical applications as the number of patients increases.

1.8.1.2 The Peer-Based Prediction Problem

Given two patients with observations $\mathbf{x}_a$ and $\mathbf{x}_b$, calculate the probability distribution of $p(\mathbf{y}_a^{(n+1)}|\mathbf{y}_b, \mathbf{x}_a, \mathbf{x}_b)$, where $\mathbf{y}_a^{(n+1)}$ stands for the next state after $\mathbf{y}^{(n)}$.

There is some existing work in dealing with similarity-based methods in real world electronic health records (EHR). Several methods have been published for predicting certain patient outcomes using large cohorts of patients. The method in [WRS10] addresses detection of heart failure more than six months before the actual date of clinical diagnosis, and [FJJ+12] discussed the inference of patient prognosis based on patient similarities. In [GSR+13], the authors presented a similarity-combining approach for computing the similarity between patients. Apparently, the authors encountered the problem of heterogenous data structure, including multi-dimensional and temporal structure. Their solution is based on a simplified structure for the EHR data for each patient, by filtering out frequent information. The individual similarity computations are followed by a combining formula ($\text{Score}(H, I) = \max_{H', I' \neq H, I} \sqrt{S(H, H') \cdot S(I, I')}$) to get the maximal similarity to a known, gold-standard EHR history set.

1.8.2 Similarity Measure Construction

Similarity measure construction plays an important role in defining the overall patient similarity.

In [GSR+13], two ICD code-based similarity measures (1-2) and eight similarity measures (3-9) between hospitalizations were defined. All similarity measures were normalized to the range[0,1].

1. ICD code similarity: ICD codes c_i and c_j as: $S(c_i, c_j) = \frac{\text{NCA}(c_i, c_j)}{\#\text{levels}}$, where NCA is the level of the nearest common ancestor and #levels are the number of levels in the ICD hierarchy.

2. Empirical co-occurrence frequency: use the HCUP data to compute empirical co-occurrences between ICD codes, across all patients. First compute the Jaccard score between each pair, then transform the Jaccard score into a similarity measure.

3. Medical history: each patient may possess medical history from three sources: (1) past encounters with local health providers; (2) discharge codes of past hospitalizations; and (3) personal history ICD codes provided in the current hospitalization. The union of these three sources constitutes the patient medical history profile. To compute the similarity of two such profiles, a bipartite graph is formed over the member ICD codes, connecting two codes in the two profiles by an edge whose weight is the similarity between the codes. The similarity score is the value of a maximal matching in this graph, normalized by the smaller history set size.

4. The maximal matching computation is performed using either of the two ICD similarity measures, resulting in **two** similarity measures.

5. Blood test similarity: The authors used only the chronologically first blood test of each type, performed upon admission for each hospitalization, retaining only blood test results obtained during the first three days of hospitalization. They also filtered many other blood tests to fit the traditional array similarity model.

6. The authors formed two other types of similarities: (1) using the entire set of common blood test array between any two hospitalizations, they computed the Euclidean distance between the z-score vectors, normalized by their length; and (2) the average of differences in absolute values between the blood tests with the highest z-score for each patient. The distance D_{ij} between patients i and j was converted to a similarity value by linear transformation.

7. ECG similarity: the ECG values included eight interval values as well as the heart rate. Similar to the blood tests, the authors used only the chronologically first measurement, performed upon admission for each hospitalization, obtained during the first three days of hospitalization. Each ECG measurement had undergone the same normalization and similarity construction as the blood tests. So, both the blood test and ECG values selected for the similarity computation have lost the characteristics of the temporal pattern.

8. Age similarity: in order to give precedence to age differences at younger ages, the similarity between two patients p_i and p_j is computed as $S(p_i, p_j) = 1 - \frac{|p_i - p_j|}{\max(p_i, p_j)}$.

9. Gender similarity: defined as 1 if the two patients have the same gender and 0 otherwise.

In [CCCM10], different similarity measures are defined according to different data types.

For those categories consisting of a single scalar data field (age, AFP value, number of lesions and tumour size), the similarity measure between the i^{th} and j^{th} patients is given by the following expression.

$$\text{SimAFP}(i,j) = \frac{1}{1 + |\,\text{afp}(i) + \text{afp}(j)|}$$

For those categories consisting of two mutually exclusive binary data fields (gender, hepatitis, portal vein invasion, degree of liver damage), the patient similarity is given by the following expression (we use degree of liver damage as an example)

$$\text{SimDamage}(i,j) = \text{damage}(i) \cdot \text{damage}(j)$$

For those categories consisting of multiple binary data fields (treatment before TACE and other image findings), the patient similarity is given by the following expression (we use treatment after TACE as an example)

$$\text{SimLtreat}(i,j) = \frac{\text{ltreat}(i) \cdot \text{ltreat}(j)}{|\,\text{ltreat}(i)| \cdot |\,\text{ltreat}(j)|}$$

For those categories consisting of two independent binary data fields (locations of lesions), the patient similarity is given by the following expression.

$$\text{SimLocLesion}(i,j) = \frac{\text{side}(i) \cdot \text{side}(j)}{|\,\text{side}(i)| \cdot |\,\text{side}(j)|}$$

For those categories consisting of multiple scalar data fields (complete blood picture, liver function test and renal function test), the patient similarity is given by the following expression (we use liver function test as an example).

$$\text{SimLFT}(i,j) = \frac{\text{lft}(i) \cdot \text{lft}(j)}{|\,\text{lft}(i)| \cdot |\,\text{lft}(j)|}, \text{where}\, \text{lft}(k) = [\text{lft}_1(k), \ldots, \text{lft}_4(k), 1]$$

1.8.3 Graph based Methods

In [KSS+14], a graph-based semi-supervised learning (SSL) algorithm is used as a classification method. The SSL graph proposed is a simple similarity graph, with edges to represent the similarity between k-neighbours. Furthermore, a multigraph, which has multi-omics information with the same node structure, could be exploited by this algorithm as well. This structure requirement can make the computation of

SSL over the multigraph much more efficient. This work demonstrates the possibility of making predictions based on similarity between patients with multi-level information in multigraphs, however, in a much-simplified manner.

1.8.4 The Learning Approach

The patient similarity algorithm SimSVM in [CCCM10] is proposed for classification on EHR data. With 14 similarity measures as input, SimSVM outputs the predicted class and the degree of similarity or dissimilarity by training a support vector machine with a linear kernel. However, this method also needs to do a lot of filtering on the raw data to produce a simplified data structure, and hence this is likely to be inadequate for large-scale heterogenous EHR data in the real world.

1.9 The Multi-Label Problem

In a vaccine efficacy study [SLF+10], several different labelled data sets $\mathcal{D}_1, \ldots, \mathcal{D}_k$ from the previous years' flu are used to help to understand the current new flu data set $\mathcal{D}_0$. Auxiliary data sets $\mathcal{D}_1, \ldots, \mathcal{D}_k$ from multiple sources $\mathcal{S} = \{S_1, \ldots, S_k\}$ ($\mathcal{D}_i \in S_i, i \in [1, \ldots, k]$), with various distributions of $\mathbf{x}_{\mathcal{D}_i}, i \in [1, \ldots, k]$, and possibly different output label alphabets $\mathcal{Y}_{\mathcal{D}_i}, i \in [1, \ldots, k]$, are considered heterogeneous. In order to make use of $\mathcal{D}_1, \ldots, \mathcal{D}_k$, the learning framework extracts $(\mathbf{x}, \mathbf{y})$ pairs similar in distribution to the ones in $\mathcal{D}_0$, and transforms the output $\mathbf{y}_{\mathcal{D}_i}, i \in [1, \ldots, k]$ to $\mathbf{y}_{\mathcal{D}_0}$ by mapping between the label alphabets $\mathcal{Y}_{\mathcal{D}_i}, i \in [1, \ldots, k]$ to $\mathcal{Y}_{\mathcal{D}_0}$. However, the validity of such transformations relies on a hidden assumption that $\mathcal{D}_0, \mathcal{D}_1, \ldots, \mathcal{D}_k$ can fit into the same model, and the corresponding $\mathbf{X}_{\mathcal{D}_i}, i \in [0, \ldots, k]$ are homogeneous. Actually, the definition proposed for heterogeneous data is limited to different label alphabets $\mathcal{Y}_{\mathcal{D}_i}, i \in [0, \ldots, k]$ and the miscellaneous mapping relationships between $\mathbf{x}$ and $\mathbf{y}$. The lack of properly modelling heterogeneous structures on $\mathbf{x}$ makes such a definition incomplete.

The multi-source problem addressed in [SLF+10] is based on the multi-view learning problem motivated by classifying web pages based on two separated feature sets [BM98, LPZ08]. The formation of a structured feature vector space then provides a rigid framework to perform various analyses [OKMI07].

Further considerations are needed for situations where the random vectors $\mathbf{X}_{\mathcal{D}_i}, i \in [0, \ldots, k]$ can have different numbers of components, that is, a dynamic-in-length but

still normalized feature space. The shared instances among a subset of $\mathbf{X}_{\mathcal{D}_i}, i \in [0, \ldots, k]$ become the basis for the overall relational network — a weighted graph [SPGP12].

The model above can be formally described as multiple data sources $\mathcal{D}_1, \ldots, \mathcal{D}_k$ providing features for objects $o_1, \ldots, o_n$. For every $o_p, p \in [1, \ldots, n], \exists\, i \in [1, \ldots, k]\, s.t. \exists\, \mathbf{x}_{o_p,\mathcal{D}_i} \in \mathcal{D}_i$ as an instance for the features of o_p in $\mathcal{D}_i$. Moreover, if there also exists a $j \in [1, \ldots, k]\, s.t. \exists\, \mathbf{x}_{o_p,\mathcal{D}_j} \in \mathcal{D}_j$, it is possible to have $|\mathbf{x}_{o_p,\mathcal{D}_i}| \neq |\mathbf{x}_{o_p,\mathcal{D}_j}|$. Thus, for every $\mathcal{D}_i, i \in [1, \ldots, k]$, all it has is the instances of one random vector $\mathbf{X}_{\mathcal{D}_i}$, fully indexed by $p \in [1, \ldots, n]$. Clearly, over-simplifying by using the representation of a single random vector for one data source greatly limits the range of heterogeneous structures this model could handle.

Although the multi-source setting partially reflects EHR data's heterogeneous alphabets, we note that for $\forall j \in [1, \ldots, N], \exists\, i \in [1, \ldots, k], s.t.\, \mathbf{x}_j \in \mathcal{D}_i$, all the random variables in $\mathbf{X}$ can be indexed by a real number, the time t. This property of EHR data naturally leads to a stochastic process model [PP02].

In order to capture the complex interdependencies between random variables at a fine-grained representational level a probabilistic graphical model is often used. As a generalization of a stochastic process, a *random field* is a collection of random variables indexed by nodes in a topological structure [Van10]. By adding restrictions to obey the Markovian property, the Markov random field (MRF) is obtained, which has been widely applied to computer vision problems, due to its simplicity and the ability to incorporate spatial information in its indices [CJ83, Li01].

The traditional learning framework over probabilistic graphical models addresses the problem of modelling the structure of the input $\mathbf{x}$, mostly either as a generative model based on a factored representation of the joint probablity of the input and output $\mathbf{y}$, or as a discriminative model where the output is conditioned on the input. However, often the output $\mathbf{y}$ is simply a single variable as the class label, probably due to the prevalence of classification problems in machine learning and thus the difficulty in obtaining labelled data of a different type [Mur12]. In the hidden Markov field (HMRF) model proposed in [ZBS01], although designed as indexed by the same set of indexes and the one-to-one mapping as in the HMM, full topological structures were considered for the first time for both the input $\mathbf{x}$ and the output $\mathbf{y}$.

The discriminative conditional random field(CRF) extends the MRF by conditioning on the input $\mathbf{x}$ as a whole [LMP01, SP03]. This is actually a great leap towards spending the most modelling effort in the structure of output $\mathbf{y}$, while keeping the possibilities of having a totally different topological structure of $\mathbf{x}$. The fact that CRF often suffers the

from the same inference complexity as the MRF does, and that its major application is in natural language processing, has so far limited further exploiting the expressive power in learning between heterogeneous structures. However, we will demonstrate later that the CRF-based model has great advantages in correctly modelling possible dependencies within the structural transition between $\mathbf{x}$ and $\mathbf{y}$, along with superior efficiency in relational learning.

1.10 Summary

The work presented in this thesis aims to propose a general framework and methods to tackle the problem of structured prediction over heterogeneous data, which has not previously been solved. A typical class of heterogeneous data — EHR data — will provide an ideal test domain. The heterogeneous nature and the complex inter-latent-variable dependencies of this data, as well the definition of a structured output to be predicted based on an encoding of human knowledge in the medical field as an ontology (although to a relatively limited extent as sourced from the ICD-10 hierarchy) gives our approach great potential for theoretical and empirical significance.

The EHR structured prediction problem requires an ontology based on a semantic vocabulary — the ICD hierarchy. The methodologies for constructing a representation of this ontology for machine learning should be able to have comprehensive coverage over all the pre-defined ICD codes, together as much of their complex inter-dependencies as possible. More importantly, the ontology should be in a suitable form such that the information embedded can be fully utilised by the probabilistic prediction model in training and prediction.

In this thesis we consider a complete ontology built on the whole ICD hierarchy, thus it has comprehensive coverage over existing human diseases. Our representation for this ontology takes a framework of measuring ICD labels and their inter-dependencies into the probabilistic model. Meanwhile, given its good coverage of the disease taxonomy, it is also potentially useful for the output structure. Thus, the ontology plays an important part not only in assisting learning and prediction, but also in constructing the output of the whole model. We will discuss this ontology-based structured prediction problem in the context of multi-label problem, and will demonstrate later that the existing models and related algorithms fail for this type of problems.

Nevertheless, to our knowledge, there is no work where the whole ICD hierarchy is used as the ontology for machine learning. We propose algorithms to transfer the ICD

hierarchy into the output structure, meanwhile utilising the embedded knowledge to assist model training and prediction.

The rest of this thesis is organized as follows. We first discuss some traditional ways of capturing and selecting features in the EHR domain, mainly based on temporal patterns. After that we fully examine the possible settings for EHR prediction problems. Then a special class of CRF called the Transitional Random Field (TRF) is proposed. A learning framework is developed in the next chapters to allow TRF the ability to conduct efficient similarity measuring, training and inference. This is then empirically evaluated on a large-scale real-world EHR dataset. Finally, some future work based on the proposed techniques is discussed.

2

Feature Construction and Selection Techniques for the EHR Data

In this chapter, we address the problem of predicting patient outcomes from EHR data in an empirical study. The selected problem is that of prediction of stroke outcome, which is a major disease with potentially serious medical consequences. Our basic hypothesis will be that physiological data patterns over the 48 hours following the occurrence of ischaemic stroke can be used to predict the outcomes for patients three months later. We investigate several approaches to the issue of data representation, both for input and output of the prediction model. We compare a number of machine learning algorithms in terms of their prediction performance. A key finding is the importance of using *temporal* features in the representation to enable improved prediction performance.

2.1 Introduction

Stroke is one of the most important diseases that can cause human death. However, despite the frequency and importance of stroke, there are only a limited number of evidence-based acute treatment options currently available. Therefore, a relatively accurate prediction of stroke outcome based on justifiable determinants could be important for decisions on the medical treatment that should be appropriate at the very beginning of the stroke.

Unfortunately, the domain and effects of the potential determinants are still unclear. Although some typical physiological variables have already attracted some attention

from scientists because of their correlation with the outcomes, relatively accurate methods for such determinants' identification are still yet to be developed.

In this chapter, we address the problem of identification of such determinants based on an extended candidate feature set including not only physiological variables but also their temporal trends. Our approach employs linear classifiers, logistic classifiers and Parzen classifiers and we use leave-one-out cross-validation to evaluate prediction accuracy. We demonstrate the efficiency and the accuracy of our approach empirically on a data set collected from the Royal Brisbane and Women's Hospital.

Stroke is a major cause of death, particularly following ischaemic heart disease (Australian Institute of Health and Welfare 2006). The World Health Organization (WHO) defines stroke as "rapidly developing clinical signs of focal (or global) disturbance of cerebral function, with symptoms lasting more than 24 hours or leading to death, and with no apparent cause other than of vascular origin" [Inv88]. The outcome for a patient, therefore, is typically the very thing that people care about most.

Although stroke is a dreadful cause of death, outcomes vary tremendously among patients. At the same time, currently physicians possess only a few therapies that can improve the outcome. These therapies produce benefit either by abbreviating the duration of ischaemia, preventing further stroke, or preventing deterioration due to post-stroke complications. However, huge differences can be observed in the health of stroke patients after three or six months, with some recovering while others may die. The majority of stroke victims survive. But in five years post-stroke, about half of all stroke survivors will be unable to function independently and will rely on others for assistance with some of all aspects of daily life. Therefore, a relatively accurate prediction of stroke outcome could potentially be important to the decision on medical treatment at the very beginning of the stroke.

2.2 Problem Setting

Our target is to identify the critical physiological determinants to achieve an accurate stroke outcome prediction. This problem is very challenging, but, on the other hand, it is potentially of great importance. This is because our understanding of the changes in the main modifiable physiological parameters, namely blood pressure, body temperature and blood glucose, and the impact these changes have on stroke outcomes, remains incomplete. In particular, threshold levels for instituting treatment to modify

these parameters, targets to be achieved with treatment, and the effectiveness of such treatments remain uncertain.

Therefore, research on the determinants and the indicators of the level of damage to the brain brought by ischaemia of the heart and the correlation between the physiological variables and the various outcomes could be of important prognostic value, and thus have attracted interest from the research community. It is reasonable to expect an improvement in the ability of the stroke unit to achieve better outcomes once any associations between the physiological variables and the outcomes of acute ischaemic stroke are adequately modelled.

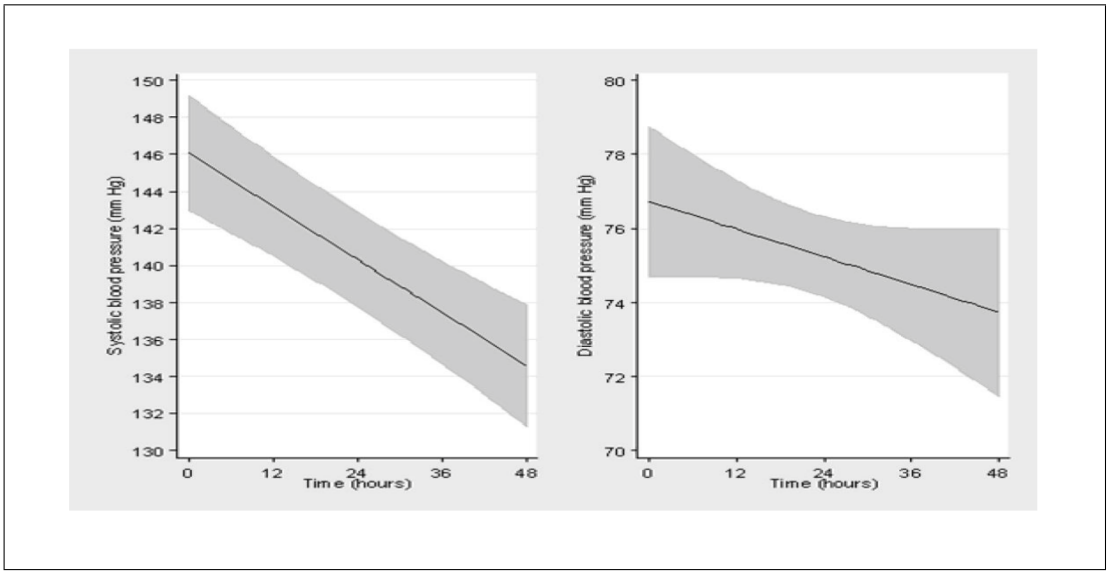

FIGURE 2.1: Mean systolic and diastolic blood pressure over the 48-hour period following ischaemic stroke [Won].

2.2.1 Modelling for the Stroke domain

There are two main types of stroke: ischaemic stroke, caused by occlusion of a cerebral artery depriving a part of the brain of blood supply and leading to infarction, and haemorrhagic stroke, caused by rupture of a cerebral artery with damage caused by bleeding into the brain. Ischaemic stroke is more common, in Australia accounting for more than 75% of all stroke occurrences [Won].

The relation of ischaemic stroke outcomes to measures of blood pressure (BP) and BP variability (BPV) has been reported in the literature. In [DMR+00] evidence was found to support the hypothesis that beat-to-beat systolic BP (SBP), diastolic BP (DBP), and mean arterial pressure (MAP) levels are associated with a greater incidence of

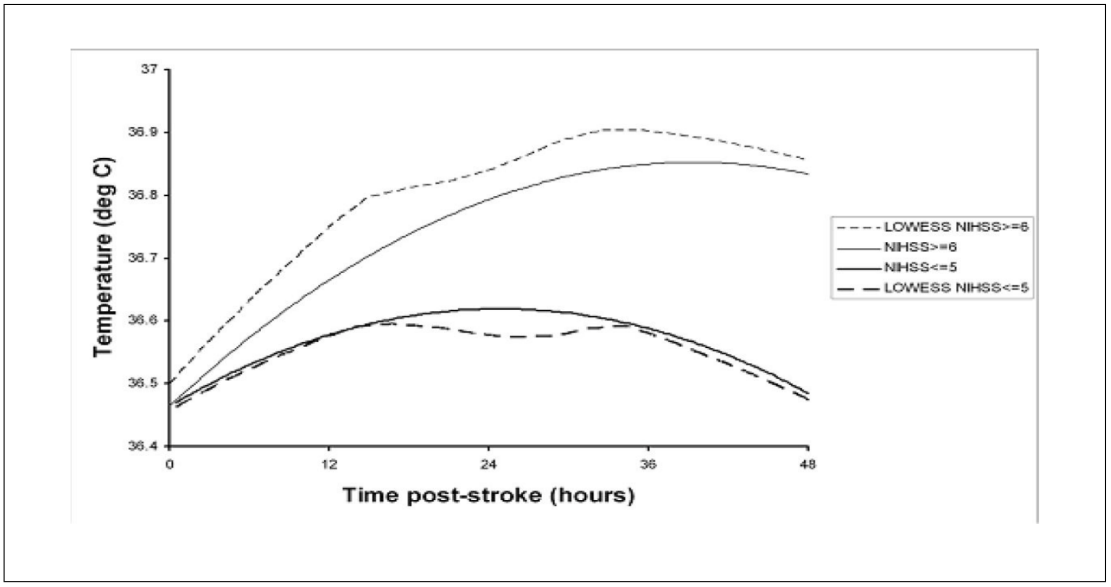

FIGURE 2.2: Mean temperature measurements over the 48-hour period following ischaemic stroke [Won]. Temperature rise and fall over the period shown for patients separated by thresholded National Institutes of Health Stroke Scale (NIHSS) scores.

target organ damage. Furthermore, it was observed that those patients with a high MAP and DBP but not SBP variability within each BP quartile had a worse prognosis compared with those with a low BPV [DMR+00].

In the literature, the relation of blood pressure to outcomes for stroke patients is complex. The effect of blood pressure reduction in the first 24 hours of acute stroke onset was found to correlate with poor outcomes [OFST+03]. Blood pressure monitored periodically within the first 72 hours after admission demonstrate that extreme values correlate with unfavourable outcomes [RKH+09]. Several statistical properties (e.g., maximum, mean, variability, etc.) of periodically monitored blood pressure within 24 hours of stroke onset have been investigated and found to have strong association with the outcome at 30 days after ischaemic stroke [YK08]. For example, variability of systolic BP is inversely associated with favourable stroke outcomes. Other blood pressure based physiological variables such as the Pulsatility Index have shown strong associations with stroke outcomes [MFBG+03].

Additionally, other physiological variables have also been demonstrated to have strong relations with acute intracerebral haemorrhage. For example, abnormalities of blood glucose [Won], heart rate variability [GSN+04], ECG [CFB05] and temperature [BC01, RLF09, HHS00] may be predictors of 3-month stroke outcomes. Since changes in such physiological variables appear to be important in determining stroke outcome this suggests that they should form part of the representation for outcome prediction

modelling. Taken all together, therefore, one of the greatest challenges of our problem setting is how to identify and quantify all the relevant physiological variables of any potential prognostic value.

The initial work of [Won] further extended the range of the relevant physiological variables considered, indicating that early changes in some common variables such as blood pressure (Fig. 2.1) or temperature (Fig. 2.2) represent potential therapeutic targets. Some systolic BP patterns have also been observed on some subgroups of patients, e.g., those suffering from infection. Such results can give us hints for selection of the physiological variables for the candidate feature set.

2.2.2 Related Work

Most of the above analyses, if not all, are based on statistical properties of periodical snapshots of physiological parameters, hourly or daily, up to three months. However, it is a natural question whether continuous patterns of physiological stream data, such as data trends, have a similar predictive role. Although it is clear that elevated blood pressure levels within 24 hours after stroke have predicted poor outcomes, few studies have investigated the predictive ability of more sophisticated trends (e.g., combined trends of several physiological parameters). This could be an effective way to readily obtain important prognostic information for acute ischaemic stroke patients.

The relationship between beat-to-beat blood pressure (BP) and early outcome after acute ischaemic stroke was described in [DMR$^+$00]. The authors also raised the question of which parameters of BP, or its variability (BPV), had the most power in predicting the outcome. The correlations between BP, BPV and the outcome were examined in different subgroups of patients (e.g., cortical infarction, subcortical and posterior circulation infarction patients). The evidence was that a poor outcome at 30 days after ischaemic stroke was dependent on stroke subtype, beat-to-beat DBP (diastolic BP), and MAP (mean arterial pressure) levels and variability. However, this study was limited to the parameters of blood pressure (BP) levels and BP variability (BPV). A further investigation on BP in [OFST$^+$03] examined the detrimental effect of blood pressure reduction in the first 24 hours of acute stroke onset. BP reduction is regarded to have the possibility to worsen an already compromised perfusion in the brain tissue, and so not lowering BP in the early stage after the stroke onset is suggested, although discussion on the relation of higher BP to outcomes is lacking. In [RKH$^+$09] blood pressure variation was represented by counting threshold violations. Significant

differences in the frequency of upper threshold violation occurrences were observed between different time points after stroke. The authors also indicated that the history of hypertension and higher National Institutes of Health Stroke Scale (NIHSS) scores on admission were independent predictors. In [Won] some temporal patterns from changes of some physiological variables were observed, and they also attempted to employ such temporal patterns to explain and predict the early outcomes. Unfortunately, the success of this prediction result was limited, due to the lack of a systematic method. However, from this and the other research mentioned above it is evident that the possibility of using machine learning to construct a predictive model for stroke outcomes based on temporal patterns in the patient's physiological data is a reasonable hypothesis.

Pre-processing time-series data to construct new features has attracted extensive study in the data mining research community, e.g., for frequent temporal pattern mining. Time series "shapelets" is one such approach, proposed in [YK09] to describe local patterns that are highly predictive of a class. Further investigation on the ability of logical combinations of shapelets to be used as an expressive primitive for time series classification is in [MKY11]. However, such techniques are unlikely to be feasible for physiological variables because of the intrinsic noise and variance of clinical data. This is due to the different intervals at which nurses collect data from patients, and also by the various lengths of patients' time in hospital. Since the assumptions behind such general-purpose pre-processing techniques as shapelets are not met in clinical data their simple application is likely to distort the actual shape of the temporal patterns in the clinical data and make them unsuitable for learning a good predictive model.

2.2.3 Classification for Stroke prediction

Our primary hypothesis is that post-stroke blood pressure, temperature and glucose exhibit time-dependent trends after ischaemic stroke and that these trends can be quantified by analysing the timing and magnitudes of serial measurements of these physiological variables (as in, e.g., [Won]). In addition, it is reasonable to hypothesise that baseline factors that represent determinants of post-stroke physiology may be detected through their significant relationships with post-stroke physiology, and statistically significant relationships with post-stroke physiological variables would indicate that these factors are determinants of post-stroke physiology [Won].

Our overall approach will be to build an enhanced candidate feature set to make sure it contains as many as possible of the relevant physiological variables. In particular, to handle the complex time series aspect of our data, variables which describe the trend

of physiological variables (e.g., the slope, the absolute change value, the mean, the standard deviation, etc.) are derived from the original ones and incorporated into the candidate feature set. Then regression based classifiers are trained and tuned against a gold standard dataset. The leave-one-out cross-validation method will be adopted to achieve a satisfactory estimate of prediction accuracy.

The main contributions of this work are as follows:

- we characterise the problem of learning a model to predict stroke outcomes from physiological patterns in real world clinical scenarios;
- we develop novel data processing methods to deal with the uncertainty inherent physiological time series of varying length and time intervals;
- we propose an algorithmic approach which will be able to use machine learning to identify significant local patterns by incorporating temporal trends of the physiological time series evident in the data into a predictive model.

2.3 Empirical Study

In this section, we describe the process of constructing the stroke outcome prediction model and its evaluation on a real-world dataset of patients. The main steps in the process fall into one of two categories: data preparation and classifier learning.

2.3.1 Data Collection

A cohort of patients with acute ischaemic stroke was recruited. Patients presenting to the Emergency Department of the Royal Brisbane and Women's Hospital, a tertiary referral teaching hospital, within 48 hours of stroke or existing inpatients with an intercurrent stroke between June 1, 2002 and March 31, 2003 were enrolled prospectively. Systolic blood pressure (SBP), diastolic blood pressure (DBP), temperature and glucose were recorded at least every 4 hours from the time of admission until 48 hours after the stroke. Measurements from patients who died during these first 48 hours were included in the analyses. Furthermore, some demographic and other stroke-related data were also collected, such as age and gender. The age range of these 173 patients was 16 to 92 years, the median age being 75 years. This data collection was approved by an ethics committee from the relevant institution.

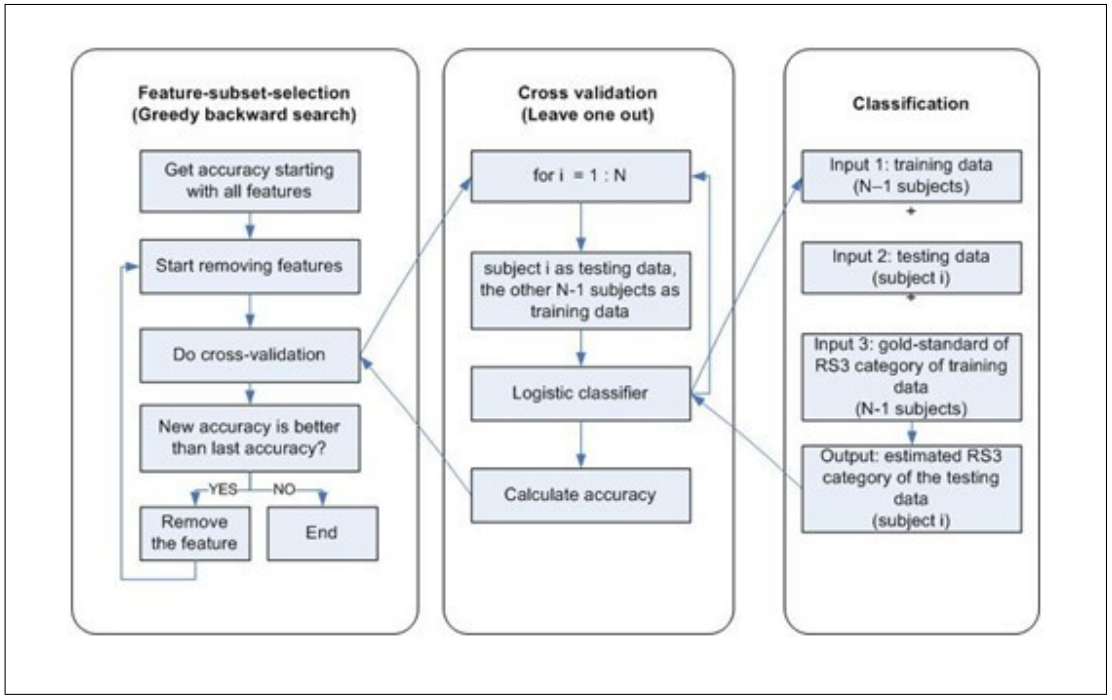

FIGURE 2.3: Flowchart for the construction of the stroke outcome prediction model.

2.3.2 Feature extraction

A number of timing and demographic features are extracted from the subjects' physiological measurements. A complete list of the 136 candidate features can be found in Table 2.1. The following paragraphs describe the generation of these features in three stages.

2.3.2.1 Feature Generation Stage 1.

This stage essentially consists of the basic features extracted from the patient records. It includes the three demographic features (features 1, 2, 3) - Age, Stroketia and AF. Additionally, features 4 - 21 are extracted from the measurements of six physiological parameters, which are blood sugar level (BSL), diastolic blood pressure (DBP), heart rate (HR), systolic blood pressure (SBP), and temperature and pulse pressure. Pulse pressure is the difference between systolic blood pressure and diastolic blood pressure:

$$P_{\text{pulse}} = P_{\text{sys}} - P_{\text{dias}}$$

Three statistical features - the mean value, median value and standard deviation, are extracted from each of the six physiological parameters.

2.3.2.2 Feature Generation Stage 2.

Another 35 features were added to the feature set at the second stage. The 35 new features were generated by extracting seven trend-pattern features from each of the five physiological parameters (BSL, DBP, HR, SBP, and Temperature). Using the MATLAB functions **polyfit** and **polyval**, one trend is applied to the time-series data of each physiological parameter of each subject in the dataset. Then the seven trend-pattern features are extracted from the trend, as follows:

1. xchange: the difference between the x value at the end of the trend and the x value at the beginning of the trend

$$\text{xchange} = x_{\text{end-of-trend}} - x_{\text{beginning-of-trend}}$$

2. absxchange: the absolute value of the xchange
3. slope: the slope of the trend
4. sign: the sign of the slope
5. bisign: the binary value of the sign of the slope
6. NumofMeasure: the number of the data points
7. FreqofMeasure: the average time interval between measurements:

$$\text{FreqofMeasure} = \frac{\Delta x}{\text{NumofMeasure}}$$

2.3.2.3 Feature Generation Stage 3.

At the third stage, the final feature set was generated by including another 80 features, thus expanding it to 136 features. The method used to obtain these new features is as follows:

- Chop the time sequence of the physiological parameter data and only use the first 48 hours' worth of data. The data after 48 hours are discarded.
- Segment the 48-hour-time-sequence data into two sub-sections: 0-24 hours and 25-48 hours.

- Fit one trend to each of the sub-sections using the MATLAB functions **polyfit** and **polyval**.
- For each of the two resulting sub-sections of each physiological parameter, ten features are extracted: mean, median, standard deviation, xchange, abschange, slope, sign, bisign, NumofMeasure, FreqofMeasure.
- Since there are many subjects who have limited data points in their blood sugar level (BSL) data, which makes it difficult to do trend segmentation, only four physiological parameters (DBP, SBP, HR, and Temperature) are used to extract these segmented-trend-pattern features.

Therefore, in total we have:

$$(10 \text{ features}) \times (2 \text{ sub-sections}) \times (4 \text{ physiological parameters}) = 80$$

which is the number of features added in stage 3.

2.3.3 Classification Criteria

Patient outcomes can be classified according to a range of different criteria. In this study, a standard seven-level scale was used. However, in order to use a two-class classifier learning algorithm the outcomes labelled according to this scale have to be converted into two classes. This partitioning was done in three different ways to avoid possible bias of the learning results.

2.3.3.1 RS3 score

A standard seven-point RS3 (Rankin Scale) scoring system for subject outcomes was used in this study [Ran57]. The interpretation of each level of this scoring system is shown in Table 2.2.

The RS3 score categorizes the outcome assessment after three months. The RS3 score of outcome assessment for the patients in this study was worked out from the follow-up data of the patients after three months. The RS3 score varies between 0 and 6. For example, patients with RS3 = 6 means the subject is dead after three months and RS3 = 0 means the subject recovers quite well after three months. The distribution of the subjects with different RS3 values over the 173 subjects is shown in Table 2.3.

TABLE 2.1: Complete set of extracted and derived features for stroke outcome prediction.

Feature No.	Feature Name	Feature No.	Feature Name
1	Age	69	DBPFeature.std2
2	Stroketia	70	DBPFeature.NumofMeasure2
3	AF	71	DBPFeature.FreqofMeasure2
4	BSLFeature.mean	72	DBPFeature.xchange2
5	BSLFeature.median	73	DBPFeature.absxchange2
6	BSLFeature.std	74	DBPFeature.slope2
7	DBPFeature.mean	75	DBPFeature.sign2
8	DBPFeature.median	76	DBPFeature.bisign2
9	DBPFeature.std	77	HRFeature.mean1
10	HRFeature.mean	78	HRFeature.median1
11	HRFeature.median	79	HRFeature.std1
12	HRFeature.std	80	HRFeature.NumofMeasure1
13	SBPFeature.mean	81	HRFeature.FreqofMeasure1
14	SBPFeature.median	82	HRFeature.xchange1
15	SBPFeature.std	83	HRFeature.absxchange1
16	TemperatureFeature.mean	84	HRFeature.slope1
17	TemperatureFeature.median	85	HRFeature.sign1
18	TemperatureFeature.std	86	HRFeature.bisign1
19	Pulse.mean	87	HRFeature.mean2
20	Pulse.median	88	HRFeature.median2
21	Pulse.std	89	HRFeature.std2
22	BSLFeature.xchange	90	HRFeature.NumofMeasure2
23	BSLFeature.slope	91	HRFeature.FreqofMeasure2
24	BSLFeature.absxchange	92	HRFeature.xchange2
25	BSLFeature.sign	93	HRFeature.absxchange2
26	BSLFeature.NumofMeasure	94	HRFeature.slope2
27	BSLFeature.FreqofMeasure	95	HRFeature.sign2
28	BSLFeature.bisign	96	HRFeature.bisign2
29	DBPFeature.xchange	97	SBPFeature.mean1
30	DBPFeature.slope	98	SBPFeature.median1
31	DBPFeature.absxchange	99	SBPFeature.std1
32	DBPFeature.sign	100	SBPFeature.NumofMeasure1
33	DBPFeature.NumofMeasure	101	SBPFeature.FreqofMeasure1
34	DBPFeature.FreqofMeasure	102	SBPFeature.xchange1
35	DBPFeature.bisign	103	SBPFeature.absxchange1
36	HRFeature.xchange	104	SBPFeature.slope1
37	HRFeature.slope	105	SBPFeature.sign1
38	HRFeature.absxchange	106	SBPFeature.bisign1
39	HRFeature.sign	107	SBPFeature.mean2
40	HRFeature.NumofMeasure	108	SBPFeature.median2
41	HRFeature.FreqofMeasure	109	SBPFeature.std2
42	HRFeature.bisign	110	SBPFeature.NumofMeasure2
43	SBPFeature.xchange	111	SBPFeature.FreqofMeasure2
44	SBPFeature.slope	112	SBPFeature.xchange2
45	SBPFeature.absxchange	113	SBPFeature.absxchange2
46	SBPFeature.sign	114	SBPFeature.slope2
47	SBPFeature.NumofMeasure	115	SBPFeature.sign2
48	SBPFeature.FreqofMeasure	116	SBPFeature.bisign2
49	SBPFeature.bisign	117	TemperatureFeature.mean1
50	TemperatureFeature.xchange	118	TemperatureFeature.median1
51	TemperatureFeature.slope	119	TemperatureFeature.std1
52	TemperatureFeature.absxchange	120	TemperatureFeature.NumofMeasure1
53	TemperatureFeature.sign	121	TemperatureFeature.FreqofMeasure1
54	TemperatureFeature.NumofMeasure	122	TemperatureFeature.xchange1
55	TemperatureFeature.FreqofMeasure	123	TemperatureFeature.absxchange1
56	TemperatureFeature.bisign	124	TemperatureFeature.slope1
57	DBPFeature.mean1	125	TemperatureFeature.sign1
58	DBPFeature.median1	126	TemperatureFeature.bisign1
59	DBPFeature.std1	127	TemperatureFeature.mean2
60	DBPFeature.NumofMeasure1	128	TemperatureFeature.median2
61	DBPFeature.FreqofMeasure1	129	TemperatureFeature.std2
62	DBPFeature.xchange1	130	TemperatureFeature.NumofMeasure2
63	DBPFeature.absxchange1	131	TemperatureFeature.FreqofMeasure2
64	DBPFeature.slope1	132	TemperatureFeature.xchange2
65	DBPFeature.sign1	133	TemperatureFeature.absxchange2
66	DBPFeature.bisign1	134	TemperatureFeature.slope2
67	DBPFeature.mean2	135	TemperatureFeature.sign2
68	DBPFeature.median2	136	TemperatureFeature.bisign2

TABLE 2.2: Rankin Scale for stroke outcome after 3 months.

RS3 Rankin Scale	
0	No symptoms
1	No significant disability
2	Slight disability
3	Moderate disability
4	Moderately severe disability
5	Severe disability
6	Dead

TABLE 2.3: Distribution of patients into each RS3 category.

RS3	No. of patients
0	42
1	23
2	16
3	23
4	22
5	14
6	33

2.3.3.2 Types of outcome grouping

To apply our method the distribution of seven RS3 outcomes as shown in Table 2.3 need to be partitioned into two groups so that classification learning methods can be used. This means that the RS3 scores are not analysed as continuous variables. Instead the RS3 score is divided into two groups in three different ways for comparison based on the values. The following are the three types of standards used to divide subjects into 'good' and 'bad' groups:

Type 1: RS3 0-1 (good) vs. 2-6 (bad)
Type 2: RS3 0-2 (good) vs. 3-6 (bad)
Type 3: RS3 0-3 (good) vs. 4-5 (bad)

The resulting distribution of patients in each of the partitions is shown in Fig. 2.4.

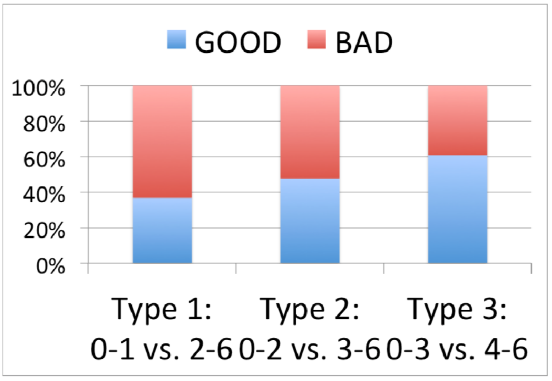

FIGURE 2.4: Distribution of patients with good or bad outcomes in 3 different groupings of RS3 codes (Y-axis: distribution percentage). Groupings into two outcome types enable the use of logistic regression (a two-class classifier).

2.3.4 Logistic Regression

In statistics, logistic regression is a type of regression analysis used for predicting the outcome of a binary dependent variable (a variable which can take only two possible outcomes, e.g. "yes" vs. "no" or "success" vs. "failure") based on one or more predictor variables. As in other forms of regression analysis, logistic regression makes use of one or more predictor variables that may be either continuous or categorical. Unlike ordinary linear regression, however, logistic regression is used for predicting binary outcomes rather than continuous outcomes.

An explanation of logistic regression begins with an explanation of the logistic function, which, like probabilities, always takes on values between zero and one: $f(z) = \frac{e^z}{e^z+1} = \frac{1}{1+e^{-z}}$. A graph of the function is shown in Figure 2.5. The input is z and the output is $f(z)$. The logistic function is useful because it can take as an input any value from negative infinity to positive infinity, whereas the output is confined to values between 0 and 1.

For subject i is usually defined as: $z = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \ldots + \beta_k x_{ik}$ $x_i = [x_{i1} x_{i2} \ldots x_{ik}]$ is the feature vector of subject i with the k features which are chosen from the 136 available features. $\beta = [\beta_1 \beta_2 \ldots \beta_k]$ are the coefficients. While calculating the optimal coefficients of a least-squares linear regression has a direct, closed-form solution, this is not the case for logistic regression. Instead, some iterative fitting procedure is needed, in which successive "guesses" at the right coefficients are incrementally improved. To do the logistic regression, the MATLAB function **glmfit** function is easy to apply.

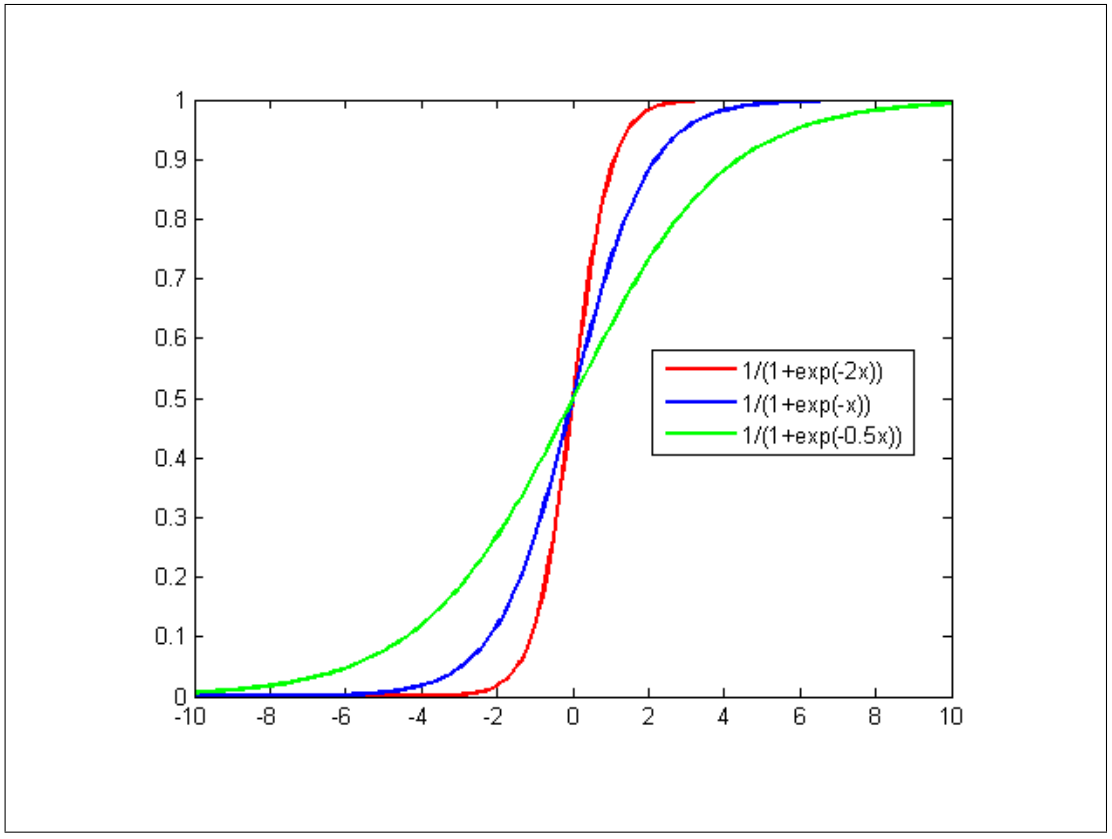

FIGURE 2.5: Plot of three functions captured in three univariate models with a single parameter or coefficient (red=2; blue=1; green=0.5) by a logistic classifier learning algorithm.

2.3.5 Leave-One-Out Cross-Validation

Cross-validation is used to obtain an unbiased assessment of classifier performance. $N = 173$ folds are employed, withholding a subject from the training set for each run, to later test with. Once a record has been withheld for testing, the classifier is trained using the remaining $(\mathrm{N} - 1) = 172$ subjects and the withheld subject is then reintroduced for classification.

2.3.6 Feature Subset Selection

In order to reduce the complexity of the model, potentially both to obtain insight into which features are important for correct patient outcome classification and to avoid overfitting, we applied feature subset selection (see Fig. 2.3).

2.3.6.1 Backward Search

A greedy backward search is performed to identify a near optimum subset of features from the 136 available, which the classifier model best fits and which provides the greatest discriminating information. Starting with all 136 features, in sequence, the feature which improves performance the most (or decreases it the least) is removed from the current set of features. This is repeated until all features have been removed. The intermediate feature subset which provides the maximum performance, compared to all other subset evaluated, is selected as the final feature set. The performance of a given feature subset is evaluated using accuracy of the classifier model used.

2.3.6.2 Forward Search

A sequential forward floating search (SFFS) algorithm was used for feature selection, in an attempt to discover the optimal subset of features from the pool of available candidate features. The optimal subset is defined as the subset of features that provides the best performance as indicated by the best accuracy value, as estimated using cross-validation. The SFFS algorithm begins with a forward-selection process, selecting from the pool of available features the single feature, which most improves the performance of the model. After selection of a feature, removal of a feature from the set of selected features is considered. The process of possible feature addition, followed by possible feature removal, is iterated until the selected feature set converges.

2.3.7 Estimation of Stroke Outcome Using Classification Learning

The following describes a procedure for estimating the subjects' stroke outcome after three months. Using the three types of two-class patient outcome labellings (see Fig. 2.4), a number of time domain features extracted from the time series data of the physiological measurements, and some supervised statistical classifiers, the stroke outcome after three months is estimated, following the descriptions above of the feature extraction and classification process.

2.3.8 Results

2.3.8.1 Classification criteria

As shown in Table 2.2 the RS3 score varied from 0 to 6 points with RS3=0 being the best outcome (no symptoms) and RS3=6 being the worst (death within three months). To simplify the prediction problem the RS3 categories were partitioned into two subsets forming "good" and "bad" outcomes. Figure 2.4 shows that this was done in three different ways, to test if any one split was better on the prediction outcome.

2.3.8.2 Prediction accuracy comparisons

With the logistic regression prediction technique previously described, we ran analyses on all above three types of grouping criteria to test our stroke outcome prediction algorithm. Trend pattern features were generated as described in Sections 2.3.2.2 and 2.3.2.3 We noticed that 'backward search' generated more accurate prediction results, and this was thus used as the default feature set search strategy.

In Figure 2.6, we also evaluated the benefits of adding trend patterns as new prediction features. Evidently, including trend patterns as prediction features improves results for all three types of grouping criteria, obtaining prediction accuracy at about 90%, compared to the previous 70%.

Figure 2.7 shows prediction accuracy comparisons under all three types of grouping criteria, broken down as 2 contingency tables. We also show the results for all three types of grouping criteria in terms of precision and recall. It is worth noting that Type 2 obtains the highest score in all the three values.

2.3.9 Discussion

We observe that, compared against predictions with only temporal and statistical features, on average these trend pattern feature based predictions can achieve a 20% increase in estimation accuracy.

The inclusion of trend patterns as prediction features in our algorithm achieved a higher precision rate as well as a good recall rate. However, the differences between precision, recall and accuracy over the three types of grouping criteria are quite small and may simply reflect the effect of less balanced class ratios for Types 1 and 3.

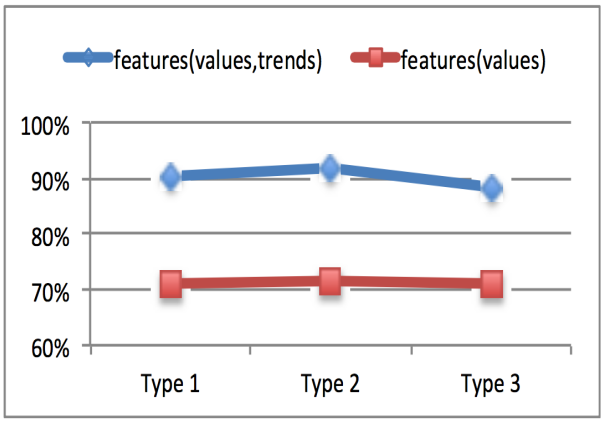

FIGURE 2.6: Including trend patterns as prediction features improves prediction accuracy (Y-axis: prediction accuracy).

Type 1

Actual	Predicted	
	Good	Bad
Good	56	9
Bad	8	100

Precision=86%,
Recall=88%,
Accuracy=90%

Type 2

Actual	Predicted	
	Good	Bad
Good	76	5
Bad	9	83

Precision=94%,
Recall=90%,
Accuracy=92%

Type 3

Actual	Predicted	
	Good	Bad
Good	97	7
Bad	13	56

Precision=93%,
Recall=88%,
Accuracy=88%

FIGURE 2.7: Prediction accuracy comparisons under three types of grouping criteria for RS3 scale values ($N = 173$).

Compared against traditional prediction methods that did not consider trend patterns of physiological parameters, we demonstrated that trend patterns play an important role in the improvement of prediction accuracy. However, our cohorts were relatively small. That is also why we only try our methods on dichotomous classification in this study. We anticipate clinical trials on larger cohorts to validate our prediction tool, to test prediction accuracy especially on RS3-score based classification.

We also anticipate new collaborations with healthcare professionals to determine the clinical truth behind those significant physiological trend patterns used in our prediction methods. We believe this will greatly benefit clinical treatments for acute ischaemic stroke.

2.4 Conclusion

In this chapter, we propose novel algorithms for digging physiological patterns to construct and select temporal features. Temporal trends were considered to enhance the completeness of the overall candidate feature set. We have demonstrated the regression based classifier can achieve reasonably accurate predictions based on the result of heuristic candidate feature searching algorithms. This approach also demonstrated its sound performance in empirical studies of clinical data sets we tested.

3 The Ontology-Assisted Structured Status Prediction

3.1 Introduction

The importance of the prediction problem over EHR data comes from the potential significant impact to human health and well-beings. Moreover, the fact that the EHR data is extremely information-rich greatly enhances the public interest from multiple sectors, including governments, academies and the whole insurance industry. It is widely accepted that EHR data contains sufficient information and the supporting argument is simple: clinicians generally can utilise the data effectively to make analysis and predictions, so the information completeness of the data source for the specific case is guaranteed[HA13]. Putting the background knowledge of professional clinicians aside, this assumption is the cornerstone for all the models for EHR prediction.

The study on statistical analysis for EHR data started well before the 1970s in epidemiology and biostatistics. Prognosis-based prediction model has also been studied by non-machine-learning researchers before the era of personal computers [Ran57, JTB+76, CAM76, BSM79, SSP+80].

The interest in automating the diagnosis process motivates the development in clinical decision-support systems since the 1980s [SKJ84, Inv88, WHB+88, JP90, LC95, HHHS98, LSF00, KHBL05, GAM+05, Don06, FS08, WS08, RS11, FJJ+12, BRR+13, MMMS+13, SGM13, CGB+15, FPY+16]. Reasoning related techniques were developed for this type of expert systems. These efforts motivate the formalisation and continuous research on modelling the problem of automatic standardised medical codes assignment in big data era [BKGOM11, BRR+13, WWH14, PPN+14, CGB+15, HDD15,

SLL+16]. Meanwhile, the importance of domain knowledge to prediction models has attracted wide attentions [SKJ84, WHB+88, JP90, WH03, HYBV05, WBK06, FS08, WS08, ZW11, CLSB11, MKE+11, SASS11, SHL+12, JJB12, SR13, BRR+13, GSR+13, MMMS+13, HA13, MD13, SRFL+14, GNDV+14, KSS+14, PSC+14, NKY+15, HDD15, PE15, CGB+15, DJW16, NZM+16, LAY+16, FPY+16, CPCC+16, JJQK+16, RKNP16, VWB+16, JGN+16, CBS+16a, LRPV16, CBS+16b, GNPI17]. Nevertheless, the prediction models developed in these works often have a very small number of output variables. We will cover more details on different aspects separately in corresponding sections later.

Despite the popularity of the prediction problem over EHR data among machine learning community, it is still an open problem. Both the output and its connection to the heterogeneous input data have not been sufficiently modelled and formalised. Moreover, we will demonstrate later that the existing machine learning training and prediction frameworks lack the capability in handling the complexity presented in this type of problems.

In this chapter, we formalise the structured status prediction problem for EHR data and identify the potential of domain knowledge in structured prediction. The contribution is as follows:

1. The ontology-assisted structured status prediction problem identification and formalisation for EHR data
2. Novel methodologies for constructing semantic descriptive hierarchy from ontology
3. Novel methodologies for domain knowledge abstraction and relation-based knowledge definition
4. Novel methodologies for embedded domain knowledge into probabilistic graphical models used for structured prediction
5. Proposed formal feature function construction techniques directly based on semantic clique-based configurations from relations
6. Examining the novel methodologies in general parallel real-world examples

3.2 The Prediction Problem over EHR Type Data

In this section, we discuss and generalize the prediction problem over EHR type data. Methodologies for analysing the relations and making predictions are widely studied by both machine learning and non-machine-learning communities in the literature. Despite the huge amount of literature on related topics and the great diversity of the prediction target, the types of machine learning models involved are somewhat limited. Approaches of directly applying existing machine learning models are often adopted, however, this often results in over-simplification to the complex nature of health prediction problems and hence implies obvious information loss. By examining the trajectory of the development of EHR prediction research, we drill down to the intrinsic requirements for modelling this type of problem and propose a novel prediction target which has not been addressed before due to limitations of existing machine learning techniques. We also propose in next sections new machine learning models and algorithms to address the modelling and computing issues after examining the existing related theories and algorithms.

We will demonstrate that the challenges from the EHR prediction problem actually generally exist in many classic machine learning problems. We use the EHR prediction problem as the major running motivation problem here to guide the discussion and development of ideas throughout this thesis. We will also apply these novel methodologies and techniques to two other parallel motivation problems to demonstrate the generality of the motivations.

3.2.1 The Prediction Task for EHR Problems

The prediction tasks in the EHR literature are hugely diverse. Obviously, the prediction task has a direct impact on the model setting. Thus, we discuss several types of prediction tasks here from different aspects accordingly.

3.2.1.1 Types of Prediction Target Variables

Early diagnosis of patient attracts the interest of medical researchers and professionals[Ran57, TKV10, MKR+16, YQF+03, WSW14, ZW11, SRFL+14, LZSI12, HLG12, KRN+91, NZM+16, ZLNY13]. Clearly, an early and accurate diagnosis has significant clinical meaning to decision-making during treatment and thus has a great influence on the prognosis. Prognostic variables, on the other hand, give estimations to the outcome

as the conclusion to a treatment or a progression of a specific disease[Ran57, SLW+15, SSP+80, FJJ+12, MFBG+03, TDP+16, OE16, YK08, KSS+14, FMG+12, CSC+13, PGAJ06, CBWB12, MKE+11, YIN+13].

A great advantage of these types of variables is that they are interpretable. Both of them have direct clinical meanings, thus the results and relevant features have clear medical explanations. Hence, the diagnostic and prognostic variables become the major source of traditional prediction tasks for EHR data.

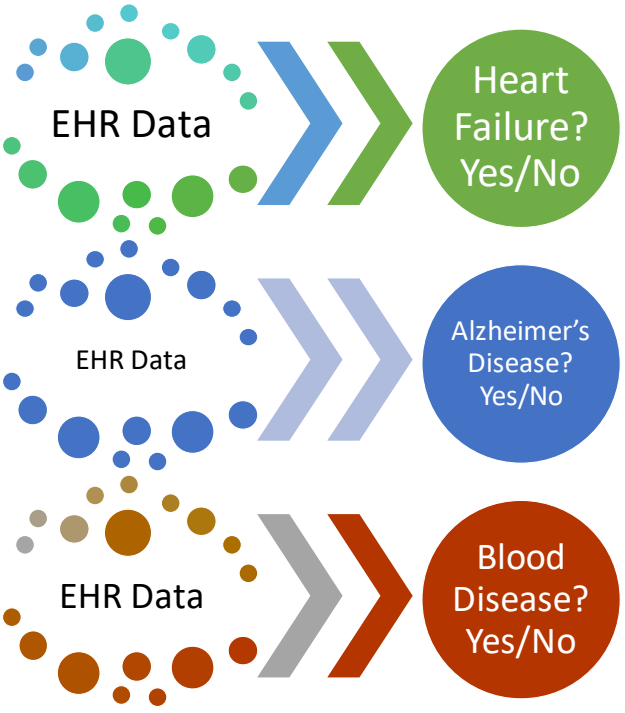

FIGURE 3.1: Disease-based prognostic variables for single-label EHR prediction.

Figure 3.1 depicts several disease-based prognostic variables for single-label EHR prediction. One of the most studied diseases for diagnosis prediction is heart failure (HF) [WRS10, ATH+13, WWH14, SHL+12, HDD15, DBA+15, FSCH16, ACZ+13, CGB+15, CBS+16a, PSC+14, CBS+16b, RS11, MMMS+13, GNPI17, KWHS13, VWB+16, SRFL+14, TSNJ14, GSR+13]. Prediction for Alzheimer's Disease (AD) is also studied extensively[LZSI12, NZM+16, ZLNY13, ENL+16, SR13, Gon16, FPY+16, ZLNY12, MKR+16, DJW16, WSW14, NKY+15, Fra16, SS15, GSRS11]. Despite medical professionals also pay attention to other diseases and ongoing research work continuously appears in recent publications, we only address the structural characteristics of the prediction models, while ignoring most of the clinical-semantic difference between different physiological output variables. Actually, as depicted in Fig 3.1, the diagnosis

prediction for some specific disease is a single-label classification problem. A prognostic variable has a clear advantage due to its aggregated semantic meaning and the remaining simplicity of being single variable.

The relation between the input EHR patient data and some specific prognostic variable was examined, e.g., the preoperative prediction of postoperative deep vein thrombosis[CAM76], the level of brain damage from severe head injury[JTB+76], the predictive capability of estrogen receptor values to patients' response to endocrine therapy[BSM79], and the relation between pre-treatment serum lactate dehydrogenase level and the prognosis of malignant lymphoma[SSP+80], etc. The set of prognostic variables is greatly extended by the development of machine learning theory and the particular interest from both the government and the industry. A group of surrogate variables have been studied intensively within modern probabilistic computing frameworks.

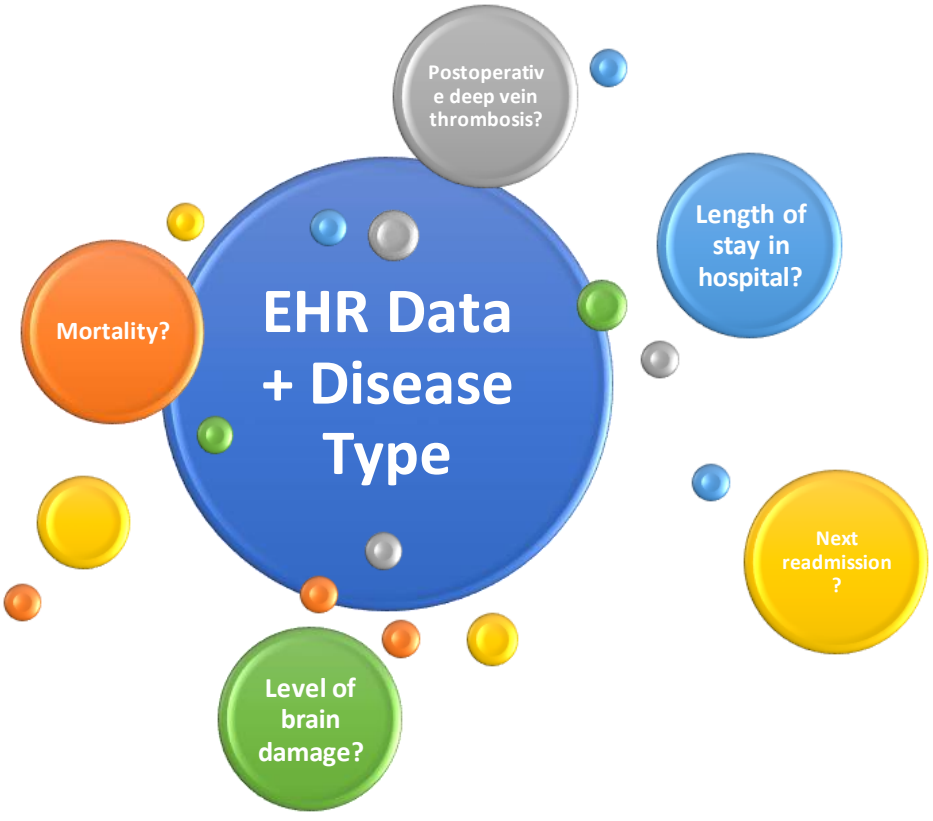

FIGURE 3.2: The single-disease model for prognostic variables prediction.

These variables normally semantically represent the outcomes or derivative outcomes of a disease or a medical treatment, e.g., short or long-term mortality[CSC+13, Hug09,

TSNJ14, GNDV+14, YIN+13, CPCC+16, NKY+15, Tu96, CA15, GNPI17, TDP+16, MFBG+03, JPS+16, LAY+16], length of stay(LOS) in hospital[GBZW93, RLF09, CPCC+16, GRB13, JPS+16, HDD15, GAM+05, PGAJ06, CBWB12, SR13, CA15, FPY+16, Tu96, JGN+16, LRPV16, vWEGF10, HHHS98, RC14, CAR97, MWHT14, PK11, SGM13, KCP11, MDLS+13, CR14, CAM76, LAY+16], readmission for hospitalized patients[CBWB12, SGM13, GRB13, CPCC+16, MDLS+13, LRPV16, SS13, DBA+15, CSC+13, GAM+05, RBS+15, SR13, FPY+16, YIN+13, KKZ12, GNPI17, HA13, Hug09, RRB13], the location of patient (in hospital, in ICU, at home or dead)[Hug09, KHBL05, CBS+16a, FMG+12, RC14], the probability for an individual patient will be developing a severe episode[BC01, SASS11, HHS00, RLF09, PGAJ06, WSW14, ZXYP13], etc. Among them the probability of mortality for a given time frame is the most widely studied outcome variable, which is treated as a direct reflection to the severity of patients' condition. Its clear clinical significance brings in the simplicity in interpreting the prediction results, thus, mortality has a long-lasting and continuous attraction to research interest. On the other hand, the other variables, e.g., Length of stay (LOS), readmission for hospitalized patients, etc., are regarded as indirect or derivative measurements of severity [PGAJ06].

3.2.1.2 Disease-Specific Vs. Multi-Task Models

The benefit of the previous settings is obvious - single variable output results in the simplicity of the probabilistic model, e.g., the direct application of logistic regression model in [HLS13, Tu96]. Moreover, often one particular setting could be used to model structures of multiple diseases. The reason is that the basic structures for the feature variables and the outcome variables are often invariant to the type of disease. Models are basically trained in the same way for different prediction targets as depicted in Fig 3.1. Thus, those different prediction target variables are in fact indistinguishable to the model in terms of structural settings. This brings about a huge number of disease-specific studies but with limited types of models. For a comparison of methodologies used for single disease predictions, see [WRS10].

As depicted in Fig 3.3, a well-motivated non-disease-specific scenario is the prediction problem over intensive care unit (ICU) data[Hug09, GNDV+14, YIN+13, ACZ+13], where multiple diseases coexist and disease-specific models do not perform well in the general prediction tasks for unlabelled patients [NKY+15]. Nevertheless, there are difficulties in combining disease-specific models for non-disease-specific scenarios, named as multi-task learning for joint diseases risks prediction [WWH14, NKY+15]. Concretely, for any target function f_t for an individual task t, a model combination

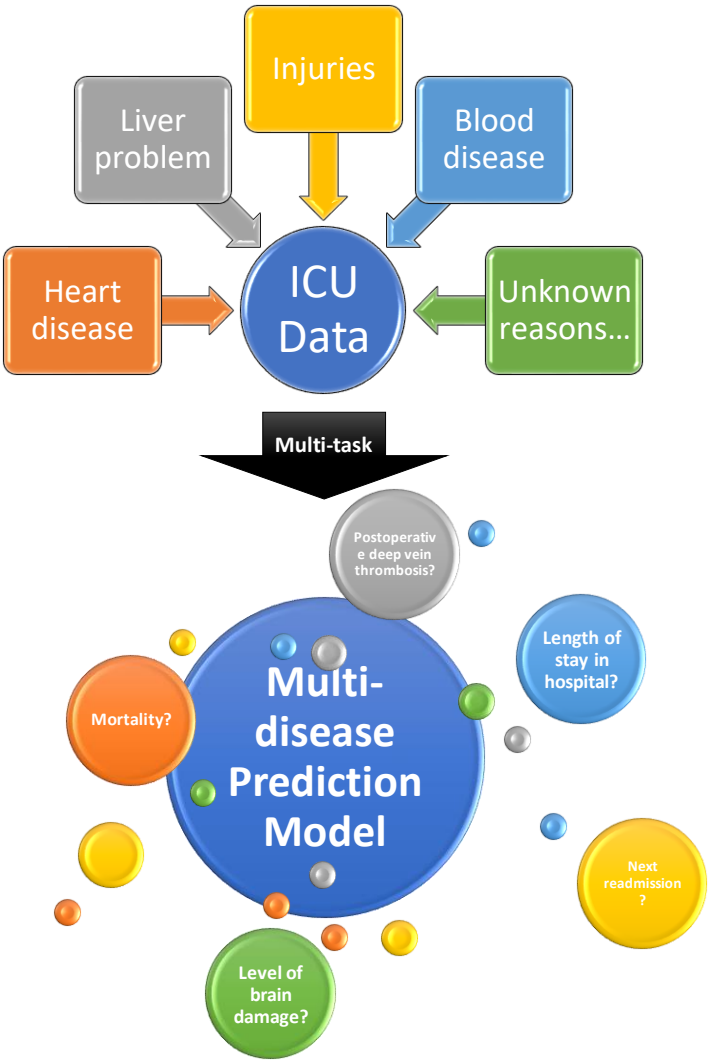

FIGURE 3.3: The multi-disease model for prognostic variables prediction.

across multiple tasks requires a sum over learning errors in all tasks, therefore, it increases the overall complexity dramatically. For a more comprehensive introduction to multi-task problems, see [CAR97, Gon16, AEP07]. It is also worth noting that it is possible to build prediction models for multiple diseases solely based on the temporal pattern of standardised diagnostic codes [CBS+16a]. Given that such model essentially eliminates the most part of information input by ignoring all the observed physiological data, it does not fit into the problem setting here.

3.2.1.3 Structured Prediction Task for Health Status

Given the previous discussion, there is a strong need for a more descriptive multi-variable structure as the prediction target. Naturally, the ultimate one is a patient's health condition. As a matter of fact, the surrogate variables discussed in the previous sections are essentially employed to describe the health condition, in a fairly simplified way. Clearly, it would be an over-simplification to the health status with a limited number of aggregated statistical variables or even values directly from physiological test results.

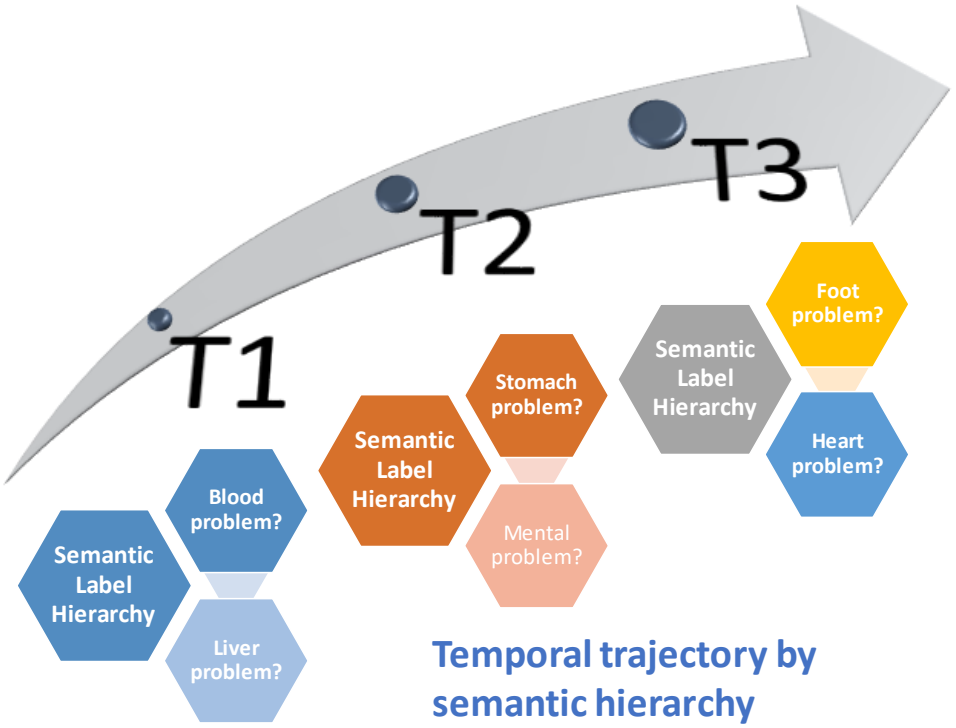

FIGURE 3.4: A simple semantic hierarchy description for temporal health status.

An alternative but more natural and desirable output setting is using health status directly as the output, instead of the surrogate statistical variables, as depicted in Fig 3.4. Despite the possible difficulty in defining health status, utilising aggregate variables to measure a patient's complete health condition is proven to be infeasible - existing methodologies fail in this task[RRB13, SGM13]. This is mainly due to the fact that "status description" is a highly subjective behaviour that inevitably involves the concept space[CBS+16b, CLSB11], where a status is normally described by concepts and properties, or more systematically, a vocabulary set of semantic variables. In light of this, a complete hierarchy of semantic labels can make up the skeleton of the

output descriptive structure. We leave the details on how to construct such descriptive structures in corresponding sections.

3.2.2 Domain Knowledge Assisted Structured Status Prediction

As discussed in previous sections, domain knowledge plays an important role in constructing the semantic output structure. Moreover, it is also critical to the prediction model itself. In the literature, the role of domain knowledge in machine-based diagnostic reasoning and prediction has drawn a wide attention from both medical and computing research communities [WH03, JP90, WBK06, BZ08]. Given the fact that the clinical and medical knowledge obtained from long term professional training is the basis of doctors' diagnosis, it is naturally assumed that machine-based methodology also needs a mechanism to abstract and utilise this domain knowledge to make reasonable predictions.

Although the domain knowledge can be in various forms, we only address the necessity of employing domain knowledge into the prediction model for EHR data here. Techniques for abstracting and utilising domain knowledge into model will be discussed in detail in corresponding sections.

3.2.3 Summary

In this section, we discuss the general modelling requirements for the prediction problem over EHR data. We summarise the challenges on an abstract level as follows:

Challenge 1. General Modelling Requirement for the Domain Knowledge Assisted Structured Health Status Prediction Problem:

The domain knowledge assisted structured health status prediction problem over EHR data requires a structured prediction target with the ability to comprehensively and semantically describe the overall health status of a patient. The model should also have the capability of projecting the output probabilistic distribution to such semantic structure, while fully incorporating the domain knowledge embedded in the ontology when doing training and prediction.

3.3 Parallel Motivation Problems

In order to demonstrate the generality of the challenges motivated by the EHR prediction problem, we list two classic problems which are widely studied in the machine learning literature as parallel motivation problems throughout this thesis. We will also demonstrate that the novel techniques proposed for the EHR prediction problem could also be applied to extend these classic machine learning problems to achieve more complicated real-world prediction tasks. The two running examples listed here will be used to address different aspects of structured prediction, which are critical to modelling the main motivation problem.

3.3.1 The Multi-Label Problem

In multi-label problem, every observation $\mathbf{x}_i \in \mathbb{R}^d$ (or $\mathbf{x}_i \in \mathbb{Z}^d$) is a d-dimensional instance to be labelled. The label space $\mathcal{L} = \{l_1, l_2, \ldots, l_q\}$ contains q possible class labels, out of which up to q labels could be associated with some $\mathbf{x}_i$. If a random variable $Y_i \in \{0, 1\}$ denotes an indicator to label $l_i \in \mathcal{L}$, we have a set of random variables $\mathbf{Y} = \{Y_1, Y_2, \ldots, Y_q\}$, with each $\mathbf{Y}_i$ associated to some label $l_i \in \mathcal{L}$. All the possible value configurations $\{\mathbf{y}_i\}_{0 \leq i < 2^q}$ to $\mathbf{Y}$ make up the output space $\mathcal{Y} = 2^{\mathbf{Y}}$. For simplicity, we define the input space as $\mathcal{X} = \mathbb{R}^d$. Thus, the learning target basically is a function $h : \mathcal{X} \mapsto \mathcal{Y}$, which maps any d-dimensional observation $\mathbf{x}_i \in \mathcal{X}$ to some $\mathbf{Y}$ value configuration $\mathbf{y}_i \in \mathcal{Y}$. According to the definition, clearly both $\mathbf{x}_i$ and $\mathbf{y}_i$ can be represented as plain vectors in $\mathbb{R}^d$ and $\{0, 1\}^q$, respectively.

Each label l_i can be viewed as a class or an assertion regarding a property. Despite the possible complex inter-dependencies between labels in $\mathcal{L}$, the whole label set $\mathcal{L}$ can be utilised as a semantic vocabulary to describe the observation $\mathbf{x}$. The output $\mathbf{y}$ is a configuration to random variable set $\mathbf{Y}$, where each $Y_i \in \mathbf{Y}$ corresponds to l_i. From a semantic point of view, the output $\mathbf{y}_i$ can be deemed as a descriptive variable (or super-variable) for $\mathbf{x}_i$ by making use of the label space $\mathcal{L}$ as a fixed vocabulary. Learning the target function $h : \mathcal{X} \mapsto \mathcal{Y}$ is equivalent to modelling $p(\mathbf{y}|\mathbf{x})$, which is the key to multi-label problem.

Despite having been studied extensively in different settings, the methodology of the multi-label problem for modelling and utilising the inter-dependencies in $\mathcal{L}$ is not sufficient. We will demonstrate later that there are still problems which cannot be solved by existing techniques. For a review of related algorithms, see [ZZ14, BK11, LZL+16].

3.3.2 Running Example 1: Text Classification

Text content classification is a classic application of multi-label problem. We take it as a running example for multi-label classification problem. In this problem, each observation $\mathbf{x}$ is a text object, e.g., an article or simply several paragraphs of text. The label space $\mathcal{L}$ consists of many topics from different abstract levels. It is often the case that a text object $\mathbf{x}_i$ covers different topics and several labels are suitable at the same time. The complexity of human language can easily embed multiple semantic meanings to a text object. We only consider the target problem here and leave the modelling details to corresponding sections.

Let's consider three possible label sets:

1. $\mathcal{L}_1$: The U.S., the U.K., Australia, Canada, France, Russia, China, Other, depicted in Fig 3.5.

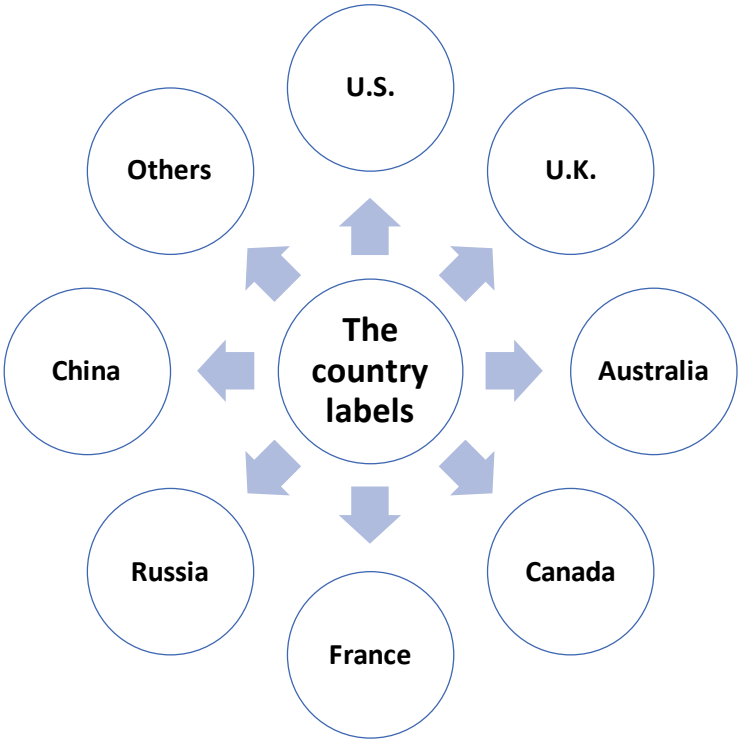

FIGURE 3.5: A mutually exclusive semantic label set.

2. $\mathcal{L}_2$: Company, start-up, high-tech, car company, traditional car company, Ford, Tesla, Toyota, Celebrity, CEO, People, entrepreneur, Elon Musk, Car, Engine Power, Max Speed, Price, Order Number, as depicted in Fig 3.6.

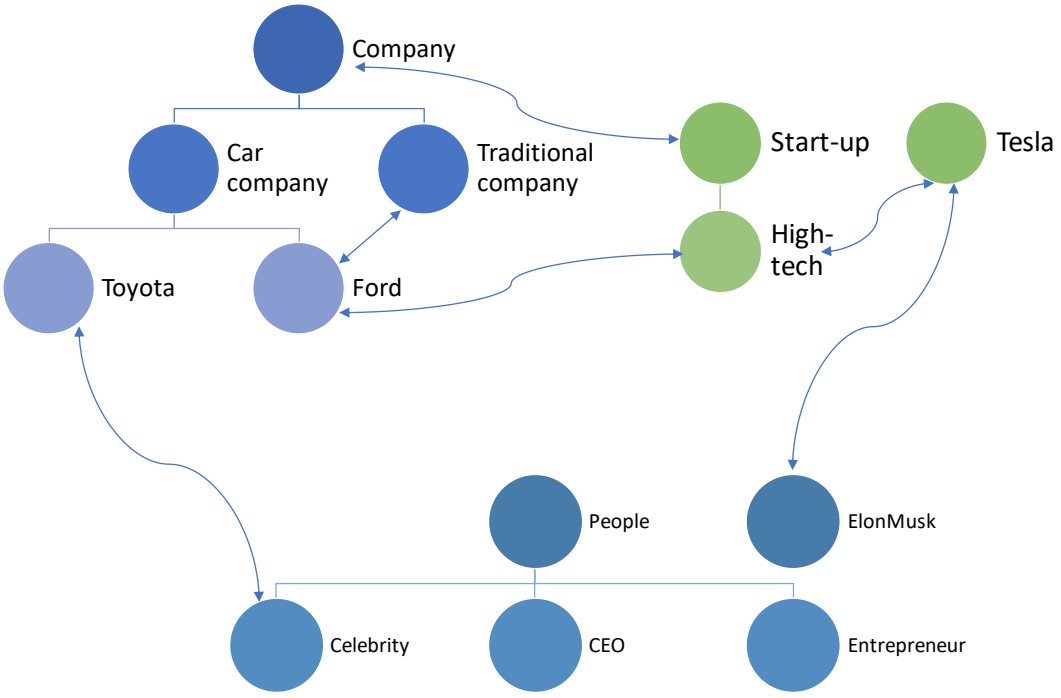

FIGURE 3.6: An inter-dependent semantic hierarchical label set.

3. $\mathcal{L}_3$: (The Plutchik's wheel for emotions): Affection Anger Angst Anguish Annoyance Anticipation Anxiety Apathy Arousal Awe Boredom Confidence Contempt Contentment Courage Curiosity Depression Desire Despair Disappointment Disgust Distrust Ecstasy Embarrassment Empathy Envy Euphoria Fear Frustration Gratitude Grief Guilt Happiness Hatred Hope Horror Hostility Humiliation Interest Jealousy Joy Loneliness Love Lust Outrage Panic Passion Pity Pleasure Pride Rage Regret Remorse Resentment Sadness Saudade Schadenfreude Self-confidence Shame Shock Shyness Sorrow Suffering Surprise Trust Wonder Worry, as depicted in Fig 3.7.

Obviously, it is safe to ignore interdependencies for $\mathcal{L}_1$, while ignoring label correlations could mean considerable information loss for $\mathcal{L}_2$ and $\mathcal{L}_3$.

Labels in $\mathcal{L}_2$ could have different types of values as measurements, e.g., Boolean values would suffice for most of them, however, numeric values are needed to describe Engine Power, Max Speed, price and order number. Moreover, the inter-dependencies are complex in $\mathcal{L}_2$, e.g., label engine power is a major decisive factor to *max speed* and *price*, while *price* itself has a strong influence on *order*. There is also possible inter-branch dependencies, e.g., *Tesla* is a brand owned by *Elon*, despite *Elon* is in the branch composed of People, CEO, celebrity, entrepreneur, etc.

[1]https://en.wikipedia.org/wiki/Robert_Plutchik

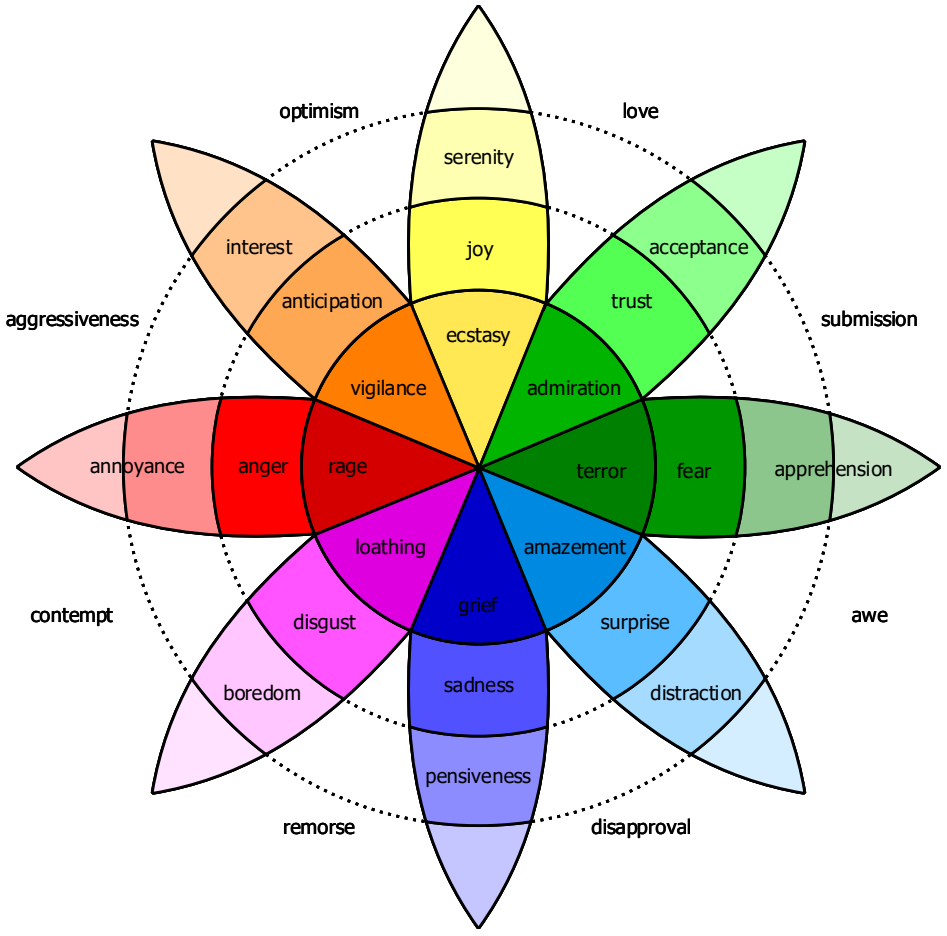

FIGURE 3.7: An ontology-based semantic hierarchy for emotion description (The Plutchik's wheel of emotions[1]).

$\mathcal{L}_3$ has a more rigid hierarchical structure, where an ontology could be built on it. For a related work on emotions prediction for text in a multi-class setting, see [ARS05].

Clearly, modelling the relations between different labels is important to the multi-label problem. The traditional second-order and higher-order approaches normally consider local dependencies but are insufficient to model the overall structure [JTYY10, EW01, FHLB08].

Challenge 2. General Modelling Requirements for a Special Text-Based Multi-Label Problem:

We consider a special type of text-based multi-label problem with a label set $\mathcal{L}$ which can be described as follows:

1. Members of $\mathcal{L}$ forms a hierarchy according to their abstraction level, such that a spanning tree exists for each label branch. The root of such spanning tree represents the most abstract level of concept and the level of abstraction decreases with travelling down along the edges.

2. Hierarchical relations cannot completely cover all the inter-dependencies between labels. If each pairwise dependency has a corresponding edge connected between the related labels, a generally connected undirected graph $G =< V, E >$ is formed, where V is the node set composed of all the labels and the edge set E represents all the pairwise inter-dependencies among $\mathcal{L}$.

General Modelling Requirement: The multi-label prediction with the previously described label set $\mathcal{L}$ problem requires methodologies to completely and accurately model the high-order inter-dependencies among $\mathcal{L}$ in a way that efficient training and predicting can be achieved.

3.3.3 Running Example 2: Image Tagging

The image segments tagging problem plays an important role in computer vision. In contrary to running example 1, where the structural modelling is mostly on the $\mathbf{y}$ side and the structural information of $\mathbf{x}$ is essentially ignored, the image tagging problem addresses the structural characteristics of $\mathbf{x}$ and its relation to the structure of $\mathbf{y}$. The observation $\mathbf{x}$ here is an image, while the output $\mathbf{y}$ is a configuration to the label indicator for a given label set $\mathcal{L}$.

The image tagging problem can have two possible settings, the segments(superpixel) level tagging and the pixel level tagging [HZCP04, LRKT09, Li01, RC95, SWS$^+$00]. This actually falls in the multi-label problem setting: given a segmentation of an image, the prediction task is to assign tags or labels to the individual segments, based on the features from texture, colour, position, etc. A direct derivation is the task of finding out the boundaries so that the computer knows what and where the segments are.

In Computer Vision, Image processing is one of the motivation problems for spatial statistical theories, e.g. MRF, whose classic applications are texture analysis and image segmentation, as depicted in Fig 3.8. A natural assumption for this type of modelling is that, adjacencies in a spatial distribution, e.g., between pixels, mega-pixels or segments in an image, imply dependencies in a probabilistic graph. There are huge amount of work on these topics, see [RC95, KPQ02, RB05, Liu15].

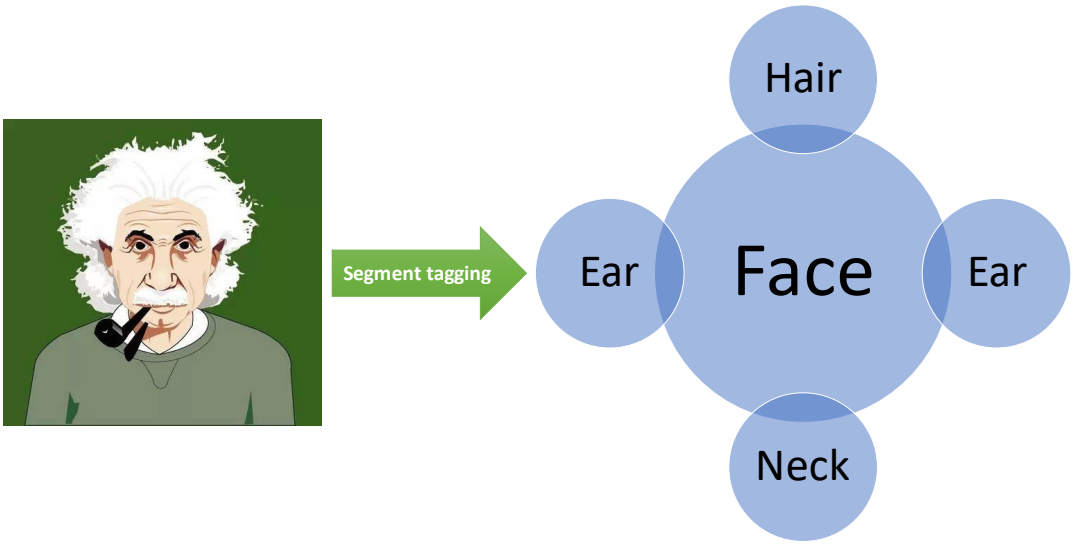

FIGURE 3.8: A simple locality-based image segmentation and labelling case.

Semantic features have been proved to be effective in computer vision [SWS+00]. This motivates the employment of semantic label set, which has its own dependency structure based the semantic relations among labels. For example, consider the following label sets as depicted in Fig 3.9:

1. $\mathcal{L}_1$: for painting styles: Modernism, Impressionism, Abstract Style, Expressionism, Cubism, Surrealism, None of the Others.

These semantic image labels are connected to the whole image instead of some local part and provide a comprehensive description to the nature of the examining image. Given that these labels actually represent a taxonomy in the domain of art, this is a precise way of describing complex objects by utilising the domain knowledge and the ontology-based descriptive hierarchy. Meanwhile, because these labels represent complicated overlapped concepts, its corresponding graph is fully connected, as depicted in Fig 3.10. In this figure, each node corresponds to a painting style.

Suppose the input is a painting and the prediction task is to assign a distribution over all the possible single-label assertions regarding the style. Apparently, we could have different reasonable estimations on the distribution, as depicted in the lower half of Fig 3.9, because the boundaries between painting styles are normally not very clear and the human-based classification is usually a very subjective matter.

Thus, the labels in $\mathcal{L}_1$ are all related to each other, even conditioned on an observation $\mathbf{x}$. Meanwhile, there is an unique label "None of the Others", indicating an intention of

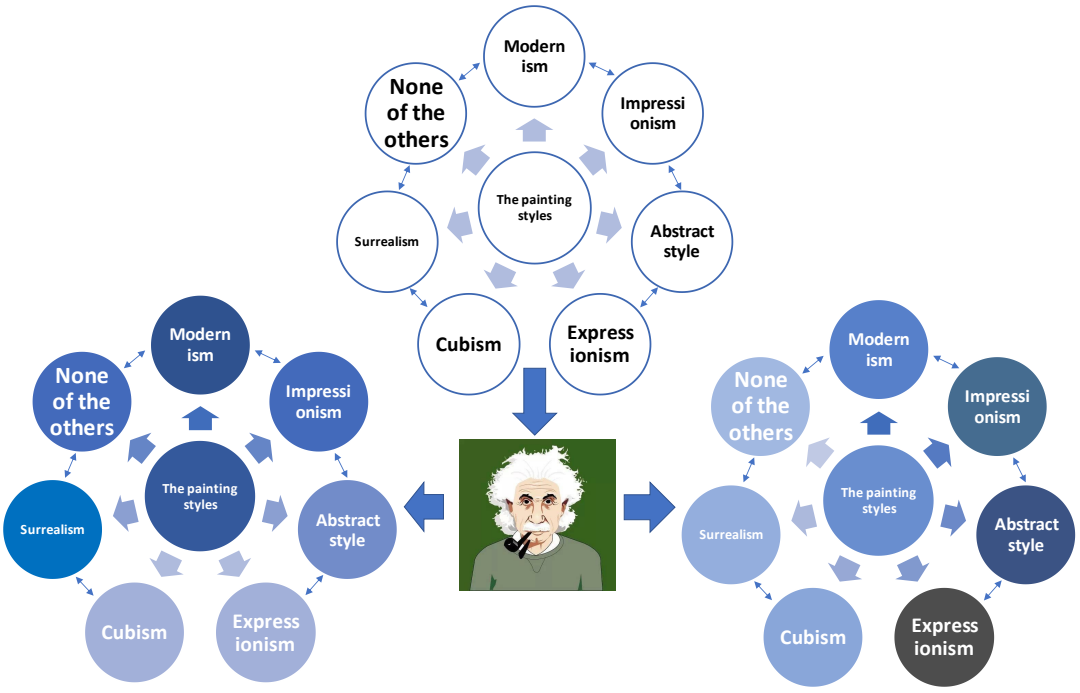

FIGURE 3.9: An ontology-based highly inter-connected semantic image labelling set.

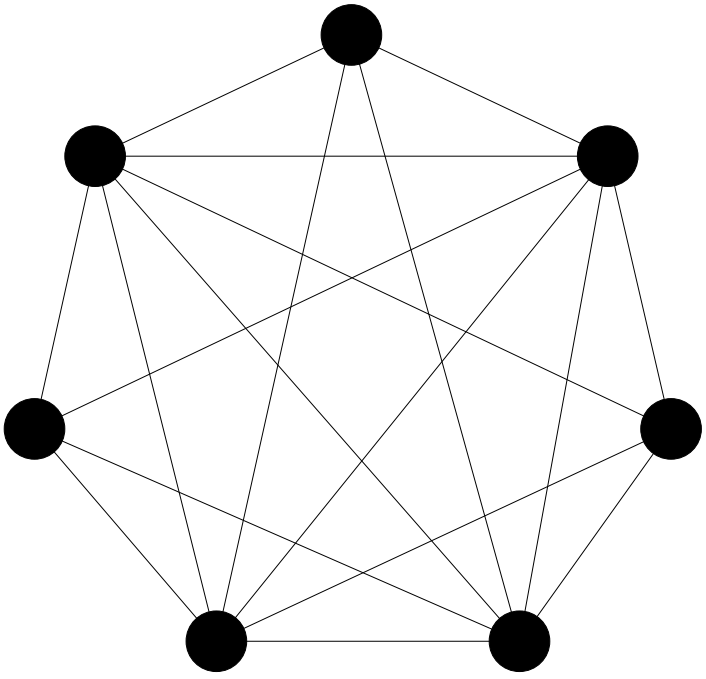

FIGURE 3.10: A structural representation of the connectivity of the ontology-based semantic image labelling set $\mathcal{L}_1$ in running example 2.

a disagreement with any other substantive label. It is worth noting that the structural variation happens on both $\mathbf{x}$ and $\mathbf{y}$ sides. Given the high level of inter-dependences in $\mathcal{L}$, the indicator set $\mathbf{Y}$ forms a fully-connected undirected graph with 7 nodes as depicted in Fig 3.10. While the segmentation of the image determines the structure of $\mathbf{x}$, which is non-isomorphic to the structure of $\mathbf{y}$, unless in exceptional rare cases. Moreover, numerical value of Y_i is used to measure the confidence level of the corresponding label l_i. Thus, the desirable output of this prediction model for $\mathcal{L}_1$ is essentially a numerical distribution of confidence values across highly inter-depended label set $\mathcal{L}_1$.

2. $\mathcal{L}_2$: (The Plutchik's wheel for emotions): Affection Anger Angst Anguish Annoyance Anticipation Anxiety Apathy Arousal Awe Boredom Confidence Contempt Contentment Courage Curiosity Depression Desire Despair Disappointment Disgust Distrust Ecstasy Embarrassment Empathy Envy Euphoria Fear Frustration Gratitude Grief Guilt Happiness Hatred Hope Horror Hostility Humiliation Interest Jealousy Joy Loneliness Love Lust Outrage Panic Passion Pity Pleasure Pride Rage Regret Remorse Resentment Sadness Saudade Schadenfreude Self-confidence Shame Shock Shyness Sorrow Suffering Surprise Trust Wonder Worry. Note that $\mathcal{L}_2$ is the same with $\mathcal{L}_3$ in running example 1 and Fig 3.7.

The emotion prediction problem has a similar setting to the painting style prediction. Nevertheless, there are still differences. First of all, the emotion label set $\mathcal{L}_2$ does not need a dummy label "None of the Others", because it is generally assumed that the ontology for emotions is complete and that one can always find out one or more most suitable emotion(s) to describe the target people. Secondly, as discussed in running example 1, $\mathcal{L}_2$ has a more rigid ontology-style structure, where domain knowledge has a stronger presentation.

Challenge 3. General Modelling Requirements for a Special Type of Image Tagging Problem:

We consider a special type of image tagging problem with a label set $\mathcal{L}$ which can be described as follows:

1. Members of $\mathcal{L}$ forms a highly inter-depended undirected graph, constructed either from a semantic structure or an ontology. In the case of non-fully-connected graph, there exists a hierarchy in $\mathcal{L}$, the root of the spanning tree in each label branch represents the most abstract level of concept and the level of abstraction decreases with travelling down along the edges. The dummy variables as parents to the corresponding root indicate the absence of the whole branch.

2. The indicator set $\mathbf{Y}$ for $\mathcal{L}$ take values in $[0, +\infty)$, thus the model output $\mathbf{y}$ is a numerical distribution of confidence values across highly inter-depended label set $\mathcal{L}$.

3. The structure of $\mathbf{x}$ determined by the segmentation of the observed image is non-isomorphic to the structure of $\mathbf{y}$.

General Modelling Requirement:

Given the previously described label set $\mathcal{L}$, input image $\mathbf{x}$ and the output $\mathbf{y}$, the image tagging problem requires methodologies to completely and accurately model the high-order inter-dependencies among $\mathcal{L}$ and the structural relations between $\mathbf{x}$ and $\mathbf{y}$ in a way that efficient training and predicting can be achieved.

3.3.4 Summary

In this section, we examine the multi-label prediction problem and discuss two classic prediction examples. By applying the challenges to the traditional settings, we demonstrate that the challenges posed by EHR prediction are applicable to general prediction problems in machine learning. We use them as running examples to lead the discussion and development throughout the rest of this chapter.

3.4 The Ontology-Based Semantic Hierarchy for Structured Status Prediction

As discussed in previous sections, the semantic hierarchy plays an importance role in describing the status in structured prediction. In this section, we discuss and answer the following questions:

- What is a structured status in structured prediction?
- What is a semantic hierarchy for structured status prediction?
- How to build a semantic hierarchy from an ontology?

3.4.1 The Extended Concept of Structured Status

We discuss the different types of structured status in prediction problems before we formalise the concept of generally-indexed structured status. It is worth noting that

the term with a similar name "structured state space" appears in signal processing and system control literature[FC93, SK94], but with a totally different definition compared to the one in machine learning problems.

3.4.1.1 The Static Structured Status in Structured Prediction

The term "status" has a direct correspondent target "stage". A structured status output $\mathbf{y}$ is generally assumed to be the description to the implied stage. However, there are cases where it is difficult to find multiple stages from the observation $\mathbf{x}$. For example, in Running example 2, there is only one stage in the observed image $\mathbf{x}$ simply because $\mathbf{x}$ is static.

Nevertheless, these two concepts bring quite a confusion in the literature when stages and status are defined on different objects. Let's consider the sequential tagging problem [LMP01, Sar06, YLCW09]. Given an observed sequence $\mathbf{x} = x_1, x_2, \ldots, x_n$, we want to predict a structured output $\mathbf{y} = y_1, y_2, \ldots, y_n$, such that each $y_i \in \mathbf{y}$ is the optimal label associated with x_i. In a structured prediction setting, the output $\mathbf{y}$ is a description to the static structured status of the whole observation $\mathbf{x}$ and there is only one stage for $\mathbf{x}$ in this setting, because $\mathbf{x}$ is static. Whereas it is also possible to define stages according to index i, so that there are n stages for the single variable x_i. In this setting, single variable y_i could also be a description to a status of stage i. Nevertheless, it is not a structured prediction problem anymore because the output for one stage has only one variable y_i so it is not a description to a structured status. Thus, the stage index is critical in defining the nature of a prediction problem. To summarise, in the sequence tagging problem, when the prediction model is $p(\mathbf{y}|\mathbf{x})$, there is one static stage and this is a structured prediction problem. Whereas when the prediction model is $p(y_i|x_1, \ldots, x_{i-1}, x_i, y_1, \ldots, y_{i-1}, i > 1, |y_i| = 1)$, the stage index is i and this is not a structured prediction problem.

Given the above, we give definition to static structured status as follows:

Definition 3.1. *Static Structured Status*

A structured output $\mathbf{y}$ *in structured prediction becomes a description to a static structured status when its implied stage index is the only one in the stage index set.*

3.4.1.2 The Time-Indexed Structured Status in EHR Models

Diseases have temporal progression trajectories[SLW+15, ENL+16, EA12]. The disease progression models (DPM) have attracted a wide interest from machine learning communities [NZM+16, ZLNY12, HNN+14, ZLNY13, WSW14, MKR+16]. Given that the majority components of an EHR observation $\mathbf{x}$ has a time stamp, $\mathbf{x}$ is roughly along the temporal trajectory and thus, stage indices are time-based ones in the corresponding structured prediction settings. A related but greatly simplified time-indexed model is the time series, which has been studied extensively and applied in many domains. For an overview of its related algorithms, see [Ham94, EA12].

3.4.1.3 The Generally-Indexed Structured Status Setting

The status of a stage is snapshot to the underlying system in progression. The stage index indicates the order of these snapshots. Although generally stage indices are not limited to time-based ones, time is a critical variable in most of progression models. We give a general definition to structured status here.

Definition 3.2. *Structured Status:*

A structured status S is a snapshot indexed by $t \in T$ to a underlying progression system, where T is a totally ordered set and S can only be fully described by a valid semantic structure.

Note that we will discuss and define the validity of a semantic structure later.

3.4.2 The Ontology

Structured status description is a direct motivation for constructing and setting up semantic structures. However, it requires a complete yet well-defined semantic label set $\mathcal{L}$ in the first place. Ontology is a formal naming and definition of the types, properties, and interrelationships of the entities in a domain, which has attracted great interest from the research community [WPBPM13, MV01, WMD08, Gru93, RLM+06, WD09, MS02, ABB+00, SCMD13, HYBV05]. Without further discussion on its mechanism on domain knowledge abstraction, we address its potentials in constructing semantic structure here.

A major component of an ontology can be essentially described as a gathering of concepts from human knowledge in a domain with a rigid taxonomy. Thus, it provides

a hierarchy of semantic labels with an excellent coverage and structural presentation. Before moving forward to introducing ontology-based semantic hierarchy, we examine an example in the medical domain first.

3.4.2.1 The ICD Coding System

The International Classification of Diseases (ICD), short for the International Statistical Classification of Diseases and Related Health Problems, is a code system released and maintained by the World Health Organization (WHO) focusing on standardising diagnostic codes for classifying diseases[2]. The ICD system has a complete coverage of human diseases and the hierarchy defined upon individual diseases provides a comprehensive and systematic description to the inter-relations. Another standardised coding system worth noting is the SNOMED Clinical Terms or SNOMED CT [3] [Don06], which also has a significant appearance as a complete label set for medical concepts in recent researches[CLSB11, MMMS+13, SRFL+14, SLL+16, DLN+09]. Considering the equivalence between the two coding systems for machine learning in terms of completeness and structural characteristics, we adopt the ICD coding system here in this thesis.

The need for standardising the disease taxonomy and description was greatly stimulated by the development of medical expert systems in the 1980s. For example, a diagnosis training system for medical students mimics the strategy of a medical expert to provide assistance to learning[WHB+88], which requires all the coding system for related diseases should be precise and suitable for machine-based inference. In addition to representing diseases themselves, the coding system should also be applicable to rule-based diagnostic reasoning, where directed graphs could be built for depicting and modelling the reasoning process[JP90, FS08]. With the technological developments for text categorisation and processing, diseases labels are used to categorise the content of discharge summaries[LC95, LSF00, PPN+14, PE15], where the inter-relationship modelling becomes critical to the overall complexity.

The utilities of ICD codes in most of the existing work on developing auto-diagnosis systems are essentially as a labelling vocabulary, whereas the labelling target is often text-based clinical notes[FS08] or discharge summaries[dLLRN98, LC95]. Without fully adoption of structured prediction related techniques, a reduced set of ICD codes or combinations among them are used as labels. The models in these work are basically with multi-label problem settings with very limited efforts in exploiting the

[2] http://www.who.int/classifications/icd
[3] http://www.snomed.org/

inter-dependencies inside the plain label set. As a classic problem in machine learning, the existing multi-label related techniques need to be further extended to handle the ICD codes based EHR prediction problems.

The ICD coding system also plays part in disease recognition and clustering, particularly in data pre-processing for disease-specific research problems. For example, in the work on heart failure prediction in [WRS10], the heart failure diagnosis criteria adopted is based on the ICD codes appeared in the problem list or clinical assessment notes and the plausibility between the prescribed medications and the ICD codes according to the ICD hierarchy. This methodology prompts potentials to the diagnosis validation problem.

Sole ICD-based prediction is a special type of work. This means the prediction model is built solely upon the momentum and flowing pattern of the appeared ICD codes, without considering any physiological EHR input data[CBS$^+$16a]. Thus, both the input and output diagnosis and medication manifestation take form of ICD codes.

Despite that the ICD coding system provides an information-rich label set with complex inter-relations, which have unfortunately yet to be fully modelled and utilised, it cannot fit into the traditional multi-label classification setting, given its cardinality and complexity. Thus, it poses challenges to the existing label-based structured prediction techniques.

3.4.3 The Ontology-Based Semantic Hierarchy

Before jumping into the formalisation of the ontology-based semantic hierarchy, we discuss its relation to the general semantic structures. Recall the label set $\mathcal{L}_1$ in running example 1 and the $\mathcal{L}_1$ in running example 2, there is no hierarchy-style structure inside the two label sets. Although the pairwise relations can be modelled by edges from a densely-connected undirected graph, there is no "root" concept or abstraction level difference existing in $\mathcal{L}$. So there is no neat way of modelling the case of "none of the others". Moreover, non-hierarchical semantic structure normally means the semantic labels are either logically diverse or not directly comparable, which means the descriptive power is very likely limited due to the granularity. Thus, this directly results in the difficulties in embedding domain knowledge into the semantic label set $\mathcal{L}$. In contrary, the hierarchical representation for taxonomy has great additional advantages in organising and making use of the domain knowledge. Hence, our discussions in this thesis focus on semantic hierarchy, particularly the one based on ontologies.

Given the discussion on structured status prediction and the relationship between the ontology and the semantic hierarchy. We ask this question, what is a valid semantic hierarchy for the demanding structured status prediction task?

Using a reduced set of semantic labels has a directly benefit in simplifying the output structure, where only semantic labels of interest comprise the semantic vocabulary $\mathcal{L}'$. However, the semantic label set adopted should be able to have a complete coverage over the labels needed for describing the structured status of $\mathbf{x}$, instead of the ones in $\mathcal{L}'$.

Definition 3.3. *The Validity of an Ontology-Based Semantic Hierarchy:*

A valid ontology-based semantic hierarchy is the one with a complete coverage over the labels needed for describing the structured status of observation $\mathbf{x}$ *in the ontology, together with related hierarchical relations.*

Then we formalise the definition of Ontology-Based Semantic Hierarchy as:

Definition 3.4. *Ontology-Based Semantic Hierarchy:*

Given a domain-specific ontology $\mathbf{O}$*, an ontology* $\mathbf{O}$*-based semantic hierarchy* $\mathbf{H}$ *is a tree or a forest, which covers all the nodes in* $\mathbf{O}$ *while keeping all the hierarchical information. More concretely, the root of each tree represents the most abstract level of concept and the level of abstraction decreases with travelling down along the edges to leaf nodes.*

3.4.4 Summary

In this section, we formally define the concept of structured status for prediction and the semantic hierarchy for describing the target structured status. The semantic hierarchy construction utilises the concepts taxonomy of the ontology to help with building descriptive structures, however, the potential of ontology has not been fully developed. We will propose ontology-based methodologies for domain knowledge abstraction and embedded in the next section.

3.5 The Ontology-Based Domain Knowledge Abstraction and Embedding

The prospective role of domain knowledge to the structured prediction problem is two-folded:

- Abstraction: The domain knowledge is transformed and abstracted into a suitable form without much information loss or distortion.
- Utilisation: The domain knowledge embedded in the abstraction is utilisable by the model, such that it can actively bring domain knowledge-based influence in a complete and precise way in the process of model training and prediction.

In this section, we first discuss the existing forms of domain knowledge in the literature and then propose our methodologies for knowledge abstraction and embedding based on ontologies. We then formalise the representation of the ontology-based domain knowledge and compare it to the ontology-based semantic hierarchy before conclusion.

3.5.1 Forms of Domain Knowledge in Prediction Models

Knowledge representation is generally based on a conceptualization of entities in the concept space[Gru93]. Concretely, each conceptualized entity can be represented by a label l and thus an enumerable label set $\mathcal{L}$ can be constructed. As the vocabulary of knowledge, the label set $\mathcal{L}$ provides a way of abstracting the relations between entities by mathematically modelling the dependencies between labels, e.g., a probability space $(\Omega, \mathcal{F}, P)$, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra and a probability measurement P can be used to depict the relation among elements in $\forall \mathcal{L}_{\text{relation}} \in \mathcal{F}$, $\mathcal{L}_{\text{relation}} \subseteq \mathcal{L}$.

However, the methodology for abstracting the inter-relations for $\mathcal{L}_{\text{relation}}$ and $\mathcal{F}$ is directly influenced by the potential way of being incorporated into the model. That is, the abstract representation has to be in a form that the model could bring into the probabilistic model. For example, the rule-based abstraction is a prevailing form for Bayesian style models[LIT92, WH03, Gru93, JP90], where the domain knowledge takes form of pairwise associative probabilities between concepts or labels as local "causal rules". Despite its popularity in reasoning, it is often difficult to construct directed graph with quantitative description. An ontology is one step further in the organisation of the concept space, where it adopts a systematic categorization of concepts[JGN$^+$16]. For the

approaches in constructing ontologies from text-based sources for further information extraction, see [WD09, Hwa99, MV01].

The form of domain knowledge in medical-related predictions also mostly falls into the rule-based probabilistic relations. Concretely, directed acyclic graphs(DAG) are often used for modelling plain structures or tree-style hierarchies[LRPV16, CPCC$^+$16]. Conditional probabilities are the quantities for describing the pairwise relationship, however, it is often difficult to infer such values directly from training. Thus, prior knowledge is needed for further modelling the overall probabilities for the graphical model.

Among the various medical domain knowledge abstractions, the simplest one is the prior for similarities. Similarity matrices for both diseases and the observed clinical features are constructed in [NKY$^+$15] to form the regularization term in the loss function. Similarly, the domain knowledge could also take form of rules in decision tree based algorithms or simply as conditional probabilities in Bayesian type methods [BZ08].

3.5.2 Domain Knowledge Abstraction and Embedding Based on Ontology

Given the above discussion, there is a huge diversity in the forms of domain knowledge in prediction models, however, there is no existing methodology in the literature being capable of accurately and comprehensively abstracting the domain knowledge. Thus, the current capability of prediction models in utilising domain knowledge is very limited. We develop and formalise an ontology-based domain knowledge model here.

Given an ontology $\mathbf{O}$, we can directly have a complete semantic label set $\mathcal{L}$ and a descriptive semantic hierarchy $\mathbf{H}$. The target capabilities of a domain knowledge $\mathbf{K}$ can be summarised as follows:

- $\mathbf{K}$ should be able to tell whether labels from a subset $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$ are related.
- $\mathbf{K}$ should be able to tell how labels from a subset $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$ are related in terms of semantic types of relations.
- Given a confidence value configuration $\mathbf{y}_{\text{sub}}$ to labels from a subset $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$, $\mathbf{K}$ should be able to give a numeric score $s \in \mathbb{R}$ indicating the level of support based on the domain knowledge.

In light of these summarised target capabilities, we develop our methodology for modelling $\mathbf{K}$.

- The complete semantic label set $\mathcal{L}$ can be treated as nodes in an undirected graph $G =< V, E >$, where the node set $V = \mathcal{L}$. An edge e_{ij} from G's edge set E provides a simple categorical information regarding the relation between nodes l_i and l_j. Concretely, nodes l_i and l_j are related if and only if e_{ij} exists. Thus, given $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$, the domain knowledge $\mathbf{K}$ can tell whether l_i and l_j ($\forall l_i, l_j \in \mathcal{L}_{\text{sub}}, i \neq j$) are related by checking the edge set E of G.

- To describe how labels from $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$ are related we first need to model the representation of relations for $\forall l_i, l_j \in \mathcal{L}_{\text{sub}}, i \neq j$. The types of relation between entities in human knowledge could be in various forms. So a categorical indicator for related or not between labels does not suffice for $\mathbf{K}$. It is worth noting that different types of relations between l_i and l_j could co-exist in many cases, as long as there is no semantic conflict. We denote $\mathbf{r}_{ij} = \{r_{ij}^1, r_{ij}^2, \ldots, r_{ij}^{m_{ij}}\}$ as the relation set indexed by l_i and l_j, where r_{ij}^k stands for the kth relation type between l_i and l_j with a total number m_{ij}. The existence of $\mathbf{r}_{ij}$ has a direct relation to the edge e_{ij}. If $e_{ij} \in E$ exists, e_{ij} has an associated relation set $\mathbf{r}_{ij}$ with $|\mathbf{r}_{ij}| \geq 1$, otherwise, $\mathbf{r}_{ij}$ does not exist. Note that because $\mathbf{r}_{ij}$ is uniquely indexed by l_i and l_j, the components and sizes vary accordingly.

  It is also important to note that the ability of describing the relation for multiple labels is critical to the model. We examine this in the context of an undirected graph $G =< V, E >$. Note that any non-fully-connected subgraph of G can be decomposed into a set of cliques. A clique-based relation model would suffice to describe any type of relation over the semantic label set $\mathcal{L}$. Consequently, we denote $\mathbf{r}_{G_{\text{sub}}} = \{\mathbf{r}_{C_1}, \mathbf{r}_{C_2}, \ldots, \mathbf{r}_{C_{m_C}}\}$ as the relation set indexed by a sub-graph $G_{\text{sub}} \subseteq G$, where $\mathbf{r}_{C_k}$ stands for the kth clique in G_{sub} with a total $m_{G_{\text{sub}}}$ cliques. To summarize, with this clique-based relation modelling, $\mathbf{K}$ could tell how labels from a given subset $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$ are related by checking ALL the $\mathbf{r}_{C_i}$ indexed by clique $\forall C_i \subseteq \mathcal{L}_{\text{sub}}$.

- In addition to modelling relations between labels, another important capability of domain knowledge is evaluating a given hypothesis. In the context of ontology-based knowledge representation, a hypothesis takes form of a confidence value configuration $\mathbf{y}_{\text{sub}}$ to labels from a subset $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$. The evaluation outcome is a numeric score $s = \mathbf{K}(\mathbf{y}_{\text{sub}}), s \in \mathbb{R}$ indicating the level of support based on the domain knowledge. The hypothesis $\mathbf{y}_{\text{sub}}$ actually is based on the semantic structure

H as vocabulary. Following the clique-based relation notation, the embedded domain knowledge for an individual G_{sub}-indexed relation $\mathbf{r}_{G_{\text{sub}}}$ is modelled by the output numeric score $s_{G_{\text{sub}}} \in \mathbb{R}$. Similarly, $s_{G_{\text{sub}}}$ indicates the level of support to the hold of relation $\mathbf{r}_{G_{\text{sub}}}$ based on the domain knowledge, given the hypothesis $\mathbf{y}_{\text{sub}}$. More concretely, $s_{G_{\text{sub}}} = \mathbf{r}_{G_{\text{sub}}}(\mathbf{y}_{\text{sub}})$. Consequently, the relation set $\mathbf{r}_{G_{\text{sub}}}$ will be associated with a score set $\mathbf{s}_{G_{\text{sub}}} = \{s_{C_1}, s_{C_2}, \ldots, s_{C_{m_C}}\}$. Then $\mathbf{s}_{G_{\text{sub}}}$ can be further aggregated to a single-valued $s_{G_{\text{sub}}} \in \mathbb{R}$ for $\mathbf{r}_{G_{\text{sub}}}$, which can be simply written as $s_{G_{\text{sub}}} = \mathbf{r}_{G_{\text{sub}}}(\mathbf{y}_{\text{sub}})$. Again, **K** could give an overall numeric score $s \in \mathbb{R}$ by checking s_{C_i} from ALL the $\mathbf{r}_{C_i}$, $\forall C_i \subseteq \mathcal{L}_{\text{sub}}$.

3.5.3 Definition

We summarize the above ontology-based domain knowledge modelling by first formalizing the key concepts here.

Definition 3.5. *Single Type Pairwise Relation between Semantic Labels in the Ontology-Based Domain Knowledge Abstraction:*

In the ontology-based domain knowledge abstraction, given an ontology **O***, a semantic label set* $\mathcal{L}$*, a descriptive semantic hierarchy* **H** *and an undirected graph* $G\langle V, E\rangle$ *with a node set* $V = \mathcal{L}$*, a single type pairwise relation* r_{ij} *associated with edge* $e_{ij} \in E$ *between semantic labels* $l_i, l_j \in \mathcal{L}$ *is a function of a confidence value* $\mathbf{y}_i$ *for* l_i *and another confidence value* $\mathbf{y}_j$ *for* l_j *in the form of* $s_{ij} = r_{ij}(\mathbf{y}_i, \mathbf{y}_j), s_{ij} \in \mathbb{R}$*, where* s_{ij} *is an output numeric score indicating the level of support to the hold of this single type pairwise relation* r_{ij} *based on the domain knowledge, given* $\mathbf{y}_i, \mathbf{y}_j$*.*

Definition 3.6. *Mixed Type Pairwise Relation between Semantic Labels in the Ontology-Based Domain Knowledge Abstraction:*

In the ontology-based domain knowledge abstraction, given an ontology **O***, a semantic label set* $\mathcal{L}$*, a descriptive semantic hierarchy* **H** *and an undirected graph* $G\langle V, E\rangle$ *with a node set* $V = \mathcal{L}$*, a mixed type pairwise relation* $\mathbf{r}_{ij} = \{r_{ij}^1, r_{ij}^2, \ldots, r_{ij}^{m_{ij}}\}$ *associated with edge* $e_{ij} \in E$ *between semantic labels* $l_i, l_j \in \mathcal{L}$ *is a set of* m_{ij} *single type pairwise relations.* $\mathbf{r}_{ij}$ *is uniquely indexed by* l_i *and* l_j*. When given a confidence value* $\mathbf{y}_i$ *for* l_i *and another confidence value* $\mathbf{y}_j$ *for* l_j*,* $\mathbf{r}_{ij}$ *is associated with a support score set* $\mathbf{s}_{ij} = \{s_{ij}^1, s_{ij}^2, \ldots, s_{ij}^{m_{ij}}\}$*, where each* $s_{ij}^k \in \mathbb{R}$ *is the support score of the single type pairwise relation* $r_{ij}^k(\mathbf{y}_i, \mathbf{y}_j)$*.* $\mathbf{r}_{ij}$ *can also be viewed as a function of* $\mathbf{y}_i$ *and* $\mathbf{y}_j$ *with an output score aggregated from* $\mathbf{s}_{ij}$*, which indicates the level of support to the hold of this mixed type pairwise relation* $\mathbf{r}_{ij}$ *based on the domain knowledge. The term pairwise relation by default refers to mixed type pairwise relation in this thesis.*

Definition 3.7. *Hypothesis in the Ontology-Based Domain Knowledge Abstraction:*

In the ontology-based domain knowledge abstraction, given an ontology $\mathbf{O}$*, a semantic label set* $\mathcal{L}$*, a descriptive semantic hierarchy* $\mathbf{H}$ *and an undirected graph* $G\langle V, E\rangle$ *with a node set* $V = \mathcal{L}$*, a hypothesis takes form of a confidence value configuration* $\mathbf{y}_{\text{sub}}$ *to labels from a subset* $\mathcal{L}_{\text{sub}} \subseteq \mathcal{L}$*.*

Definition 3.8. *Clique-Indexed Relation in the Ontology-Based Domain Knowledge Abstraction:*

In the ontology-based domain knowledge abstraction, given an ontology $\mathbf{O}$*, a semantic label set* $\mathcal{L}$*, a descriptive semantic hierarchy* $\mathbf{H}$ *and an undirected graph* $G\langle V, E\rangle$ *with a node set* $V = \mathcal{L}$*, a relation* $\mathbf{r}_C = \{\mathbf{r}_{C_1}, \mathbf{r}_{C_2}, \ldots, \mathbf{r}_{C_{m_C}}\}$ *indexed by a clique* $C \subseteq G$ *is a mixed type pairwise relation when* C *is an edge, otherwise* $\mathbf{r}_C$ *is a set composed of all the relations indexed by the cliques in* C*, excluding itself, if applicable. The support score value for a hypothesis* $\mathbf{y}_{\text{sub}}$ *on the relation indexed by clique* C *is an aggregation of score values of its member relations.*

Definition 3.9. *Graph-Indexed Relation in the Ontology-Based Domain Knowledge Abstraction:*

In the ontology-based domain knowledge abstraction, given an ontology $\mathbf{O}$*, a semantic label set* $\mathcal{L}$*, a descriptive semantic hierarchy* $\mathbf{H}$ *and an undirected graph* $G\langle V, E\rangle$ *with a node set* $V = \mathcal{L}$*, a relation* $\mathbf{r}_{G'} = \{\mathbf{r}_{C_1}, \mathbf{r}_{C_2}, \ldots, \mathbf{r}_{C_{m_C}}\}$ *indexed by a graph* $G' \subseteq G$ *is a mixed type pairwise relation when* G' *is an edge, otherwise* $\mathbf{r}_{G'}$ *is a set composed of all the relations indexed by the cliques in* G'*, excluding itself, if applicable. The support score value for a hypothesis* $\mathbf{y}_{\text{sub}}$ *on the relation indexed by graph* G' *is an aggregation of score values of its member relations.*

It is worth noting that, although the definitions for graph-indexed and clique-indexed relations are similar, the graph-indexed one relies on the recursive definition of clique-indexed relations. Thus, they cannot be combined.

Definition 3.10. *The Ontology-Based Domain Knowledge:*

Given an ontology $\mathbf{O}$*, a semantic label set* $\mathcal{L}$ *and a descriptive semantic hierarchy* $\mathbf{H}$*, the ontology-based domain knowledge* $\mathbf{K}$ *is a tuple* $\mathbf{K} = \langle G\langle V, E\rangle, \{\mathbf{r}_C\}_{C\subseteq G}\rangle$*, where* $G\langle V, E\rangle$ *is an undirected graph with a node set* $V = \mathcal{L}$ *and* $\{\mathbf{r}_C\}_{C\subseteq G}$ *is a set of relations indexed by every clique in* G*. For every hypothesis* $\mathbf{y}_{\text{sub}}$*, the ontology-based domain knowledge* $\mathbf{K}$ *is able to give a numeric score* $s = \mathbf{K}(\mathbf{y}_{\text{sub}}), s \in \mathbb{R}$ *indicating the level of support based on the domain knowledge* $\mathbf{K}$*.*

3.5.4 Summary

In this section, we discuss and develop a systematic way for ontology-based domain knowledge abstraction and embedding. The definitions proposed in the end presents the methodology and building units for formalizing ontology-based domain knowledge. This section is the second half for utilizing ontology in structured prediction. In contrary to the ontology-based semantic hierarchy, the ontology-based domain knowledge has a way more complex structure and the included graph is undirected. Whereas the ontology-based semantic hierarchy only provides a complete label set and the hierarchical structure is based on tree or forest, which is directed. Despite both being ontology-based, the functionality differs.

We will demonstrate later, the methodologies for domain knowledge abstraction and embedding proposed here are the theoretical basis for the probabilistic graphical model settings. They have deep connections not only to the features construction for the output structure but also for the observed heterogeneous input data.

3.6 The Ontology-Assisted Structured Status Prediction Problem

Given the modelling and formalization on the structured prediction task, ontology-based semantic hierarchy and domain knowledge, we summarize the ontology-assisted structured status prediction problem by formalizing the problem setting. We discuss the generality of this problem and example prediction scenarios in the real world before concluding the problem modelling part of this thesis.

3.6.1 The Prediction Problem Setting

The ontology-assisted structured status prediction problem can be formally described by the following:

Given an ontology $\mathbf{O}$, the semantic label set $\mathcal{L}$ provided by $\mathbf{O}$, an ontology $\mathbf{O}$-based semantic hierarchy $\mathbf{H}$, an ontology $\mathbf{O}$-based domain knowledge $\mathbf{K}$ and a heterogeneous observation $\mathbf{x}$, the model predicts the hypothesis $\mathbf{y}$ on $\mathbf{H}$ with the largest $p(\mathbf{y}|\mathbf{x}, \mathbf{O}, \mathbf{H}, \mathbf{K})$ for a stage indexed by a given t, where the hypothesis $\mathbf{y}$ takes form of the distribution of confidence values over $\mathcal{L}$.

3.6.2 Real-World Scenarios

Although being motivated by the EHR prediction problem, the ontology-assisted structured status prediction is actually a very general yet fundamental problem in machine learning. We demonstrate this by examining three real-world scenarios in the context of parallel examples.

- Text-based interest flow prediction:

  Suppose a man decides to spend 1 hour in reading news online. News websites would like to know more about the temporal progression of the reader's interest in reading within that time. Consider a semantic label set similar to $\mathcal{L}_2$ in running example 1 to be used in the prediction model, $\mathcal{L}_2$ provides a way to describe the reader's interest with these hierarchical topics. The article topic taxonomy from linguistics comprises the ontology $\mathbf{O}$, which provides a complete semantic topic hierarchy $\mathbf{H}$. The relations between semantic labels or even news features could also be defined by professionals, thus they can be abstracted to an $\mathbf{O}$-based domain knowledge $\mathbf{K}$. Let's assume the reader likes both pictures and texts on the news website. The observation $\mathbf{x}$ is heterogeneous because it consists of multiple data formats. The website would like to predict the reader's interest in terms of the distribution over the semantic topic hierarchy after 30 minutes, given all the reading history $\mathbf{x}$ in the first 30 minutes.

- Image-based emotion prediction:

  Recall the discussion for running example 2, the structural status for one image is static. However, when an art student is browsing an online art gallery, there should be a huge number of images. Consider the emotion semantic label set $\mathcal{L}_2$ proposed in running example 2, each image actually corresponds to a stage and they can be easily by a main stage index throughout the browsing process. Clearly, the whole emotion progression trajectory provides a good context of structural status prediction.

  The emotion taxonomy from psychologist comprises the ontology $\mathbf{O}$ here, which provides an $\mathbf{O}$-based semantic hierarchy $\mathbf{H}$. It is also possible for psychologist to embed their knowledge regarding the connections between colour, shape and human emotions, etc., into the ontology $\mathbf{O}$. Thus it can be assumed that the $\mathbf{O}$-based domain knowledge $\mathbf{K}$ is obtainable. Images or paintings have intrinsic complexities in colour, texture, shape, etc. Thus the browsed image as observed data $\mathbf{x}$ is heterogeneous. The online gallery wants to predict the art student's

emotion in terms of the distribution over the semantic emotion hierarchy **H** when the student is looking at the 100th painting, given all the previously browsed 99 ones.

- Video-based emotion prediction: Given the same ontology **O** for emotions, the semantic label set $\mathcal{L}$ and the semantic emotion hierarchy **H**, we can easily transfer the image-based emotion prediction problem to a video-based one. Suppose a college student is watching movie. As the types and intensity of visual stimulates changes, together with the story progresses, the student's emotion fluctuates accordingly. Suppose we can bring in the video features into the ontology for emotions, again it turns out to be an ontology-assisted structured status prediction problem.

3.7 Conclusion

In this chapter, we identify and propose solutions to the ontology-assisted structured status prediction problem. The techniques for abstracting and implementing domain knowledge are fully formalised. Essentially, the problem modelling part is addressed and completed. Despite being motivated in the EHR prediction problems, we demonstrate its generality by examining real-world prediction scenarios for progression systems. We will go further into the probabilistic computing models in the next chapters to fulfil the computing requirements discussed and proposed here.

4 The Transitional Random Field (TRF): Modelling

4.1 Introduction

Following up the contributions to identify and model the ontology-assisted structured status prediction problem, in this chapter, we address the probabilistic computing models, particularly for graphical models. As we will demonstrate in this chapter, theoretical improvement is needed to address all the requirements discussed in the ontology-assisted structured prediction problem.

In this chapter, we first discuss the general probabilistic learning and prediction framework as background. Then we fully examine the equivalence relation between a set of feature functions and the underlying structure. Also, the feature function setting is important to comprehensively and precisely represent the domain knowledge. Thus, we propose a novel model for heterogeneous input data and identify and formalise the locality preserving property, which is an important formalisation for further theoretical discussion. In light of these, we propose the transitional random field to conclude this chapter.

The theoretical contribution to the general structured prediction problem in machine learning is as follows:

- Proposed the structural equivalence condition for feature functions and the underlying graph structure
- Proposed the equivalence condition for domain knowledge abstraction and representation

- Proposed the heterogeneous input model with novel connection settings between input and output
- Formalise the locality preserving property, a critical underlying assumption widely existing in the literature, which limits the CRF's capability.
- Mixed type of dependencies analysis and construction for probabilistic graphical models
- Proposed a novel structured prediction model: the transitional random field, to address the identified problems
- Identify the information transition process from heterogeneous input data to a rigid output ontology based structure

4.2 Probabilistic Models for Structured Prediction

The general goal of supervised learning is to model the relation between the input $\mathbf{x}$ and output $\mathbf{y}$ from a given training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where N is the number of training examples. In the context of statistical learning and machine learning, the relation between $\mathbf{x}$ and $\mathbf{y}$ often takes form of the distribution of joint probability $p(\mathbf{x}, \mathbf{y})$ or conditional probability $p(\mathbf{y}|\mathbf{x})$. Probabilistic models are extensively studied in the machine learning literature for efficient computing the conditional probability distribution (CPD) $p(\mathbf{y}|\mathbf{x})$, which is vital to training and prediction. Concretely, the prediction for any new input $\mathbf{x}_j \notin \mathcal{D}$ is achieved by inferring a corresponding $\mathbf{y}_j$ from the model $\mathbf{P}$ learnt from $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$.

The structural characteristics of both $\mathbf{x}$ and $\mathbf{y}$ are important to modelling, training and prediction. In this section, we address the methodologies for computing probability distributions by fully examining related models with various structural settings. We start with single-output and non-structured multi-output settings before fully examining probabilistic graphical models for structured prediction. The structural characteristics of these models, in order of increasing complexity, directly determine the types of dependencies a graphical structure can model, and hence the applicability of each model class to different real-world problems.

4.2.1 Single-Output Prediction Models

We start with very simple cases where $|\mathbf{y}| = 1$, that is, $\mathbf{y}$ contains one single output variable y. The goal is to learn $p(y|\mathbf{x})$, given a training set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, where N is the number of training examples.

The range of y has a direct influence to the way we develop probabilistic models. When the range of y is a finite set $\{c_0, c_1, \ldots, c_k\}$, with each y_i being a categorical or nominal variable, it becomes a classification problem. In this case, the value of y_i is either a discrete number or a categorical value, both representing some class.

On the other hand, when each y_i is real-valued, e.g. $y \in \mathbb{R}$, we are having a regression problem. For both cases, there are many models that have been studied and applied extensively (text books on machine learning normally provide a good coverage on these, e.g. [Mur12, Bis07]). Here we examine two classic models with continuous and discrete output y, respectively.

4.2.1.1 Linear Regression

We consider the situation where the output $\mathbf{y}$ is a single continuous response variable, e.g., $y \in \mathbb{R}$. Linear regression asserts that y is a linear function of the inputs $\mathbf{x}$, which takes the form:

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \epsilon = \sum_{j=1}^{D} w_j x_j + \epsilon$$

where $\mathbf{w}$ is the weight vector and $\mathbf{w}^T\mathbf{x}$ is an inner or scalar product between the input vector $\mathbf{x}$ and $\mathbf{w}$, and ϵ represents the residual error of prediction outcomes compared to true y_i.

Then we come to a second assumption that, the distribution of ϵ is Gaussian with mean μ and variance σ^2, which can be denoted by $\epsilon \sim \mathcal{N}(\mu, \sigma^2)$.

Based on these two assumptions above, $p(y|\mathbf{x}, \boldsymbol{\theta})$ is also Gaussian, given $\mathbf{x}$ and parameter $\boldsymbol{\theta}$:

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mu(\mathbf{x}), \sigma^2(\mathbf{x})) \tag{4.1}$$

The term $\mathbf{w}^T\mathbf{x}$ can be further modified to model non-linear relationships by applying *basis function expansion*, which replaces $\mathbf{x}$ with some non-linear function of the inputs,

$\phi(\mathbf{x})$. It doesn't change the number of assumptions have to be made regarding the numerical relation between $\mathbf{x}$ and $\mathbf{y}$ and the distribution of ϵ, though.

If the training examples in $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ are independent and identically distributed (**iid**), which is almost always assumed for $\mathcal{D}$. We can write the log-likelihood as:

$$\ell(\boldsymbol{\theta}) \triangleq \log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$$

The negative log likelihood (NLL) can be used to please the often minima-targeted numerical optimisers. The NLL is defined by simply adding a negative sign to $\ell(\boldsymbol{\theta})$, as:

$$\text{NLL}(\boldsymbol{\theta}) \triangleq -\sum_{i=1}^{N} \log p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$$

The analytical form of $p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$ can be inserted into $\ell(\boldsymbol{\theta})$ with the Gaussian representation. Then we can rewrite $\ell(\boldsymbol{\theta})$ as:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} \log \left[\left(\frac{1}{2\pi\sigma^2} \right)^{\frac{1}{2}} \exp \left(-\frac{1}{2\sigma^2} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 \right) \right]$$

If we define the *residual sum of squares* (RSS) function (also called *sum of squared errors* (SSE)) as:

$$\text{RSS}(\mathbf{w}) \triangleq \sum_{i=1}^{N} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 = \|\epsilon\|_2^2 = \sum_{i=1}^{N} \epsilon_i^2$$

where $\epsilon_i = (y_i - \mathbf{w}^T \mathbf{x}_i)$.

Then we can get a simpler form of $\ell(\boldsymbol{\theta})$ as:

$$\ell(\boldsymbol{\theta}) = \frac{-1}{2\sigma^2} \text{RSS}(\mathbf{w}) - \frac{N}{2} \log(2\pi\sigma^2)$$

Due to the simplicity of the target optimisation function, $\hat{\mathbf{w}}$ is the *maximum likelihood estimate* (MLE) of $\mathbf{w}$ if and only if it minimises the RSS.

Structural characteristics:

In linear regression, each input $\mathbf{x}_i$ is assumed as a plain vector and there is a continuous corresponding single variable y_i for it. The mapping $\mathbf{x}_i \mapsto y_i$ in this model thus

considers no structural information inside $\mathbf{x}_i$ or y_i. Actually, y_i can be seen as a numerical measurement for the status of input $\mathbf{x}_i$ by aggregating it into one single numerical value.

4.2.1.2 Logistic Regression

We consider the case where the output $\mathbf{y}$ is a single categorical response variable, e.g., $y \in \{c_0, c_1, \ldots, c_k\}, k \geq 1$. Similar to the case of linear regression, we also need to make assumption on $p(y|\mathbf{x})$.

When $k = 1$, the *multiclass classification* problem reduces to a *binary classification*. We can generalise the linear regression model to a binary classification setting simply by replacing the Gaussian assumption for y with a Bernoulli distribution to suit the output $y \in \{0, 1\}$ as:

$$p(y|\mathbf{x}, \mathbf{w}) = \text{Ber}(y|\mu(\mathbf{x}))$$

where $\mu(\mathbf{x}) = \mathbb{E}[y|\mathbf{x}] = p(y = 1|\mathbf{x})$.

Given that $\mu(\mathbf{x}) = p(y = 1|\mathbf{x})$ is actually a function from the input $\mathbf{x}$ to $[0, 1]$, this mapping can be further represented by a *sigmoid* function (also called *logistic* or *logit* function) of the weighted sum of $\mathbf{x}$. The range can be guaranteed by the definition of sigmoid function as follows:

$$\text{sigm}(\eta) \triangleq \frac{1}{1 + \exp(-\eta)} = \frac{e^{\eta}}{e^{\eta} + 1}$$

Clearly, any $\mu(\mathbf{x})$ can be replaced by $\text{sigm}(\mathbf{w}^T\mathbf{x})$ with the help from $\mathbf{w}^T$. Then we can rewrite $p(y|\mathbf{x}, \mathbf{w})$ for the logistic regression model, as:

$$p(y|\mathbf{x}, \mathbf{w}) = \text{Ber}(y|\,\text{sigm}(\mathbf{w}^T\mathbf{x}))$$

In spite of its name, logistic regression is actually a form of classification because of the range $y \in \{0, 1\}$. The appearance of $\mathbf{w}^T\mathbf{x}$ in formalising $p(y|\mathbf{x})$ brings some similarity to linear regression and hence the name. The assumption of Bernoulli distribution for $p(y|\mathbf{x})$ in logistic regression distinguishes itself from regression problems.

Given the probability distribution above, the negative log-likelihood (NLL) for logistic regression with $y \in \{0, 1\}$ can be rewritten as follows:

$$\begin{aligned} \text{NLL}(\mathbf{w}) &= -\sum_{i=1}^{N} \log[\mu_i^{\mathbb{I}(y_i=1)} \times (1-\mu_i)^{\mathbb{I}(y_i=0)}] \\ &= -\sum_{i=1}^{N} [y_i \log \mu_i + (1-y_i) \log(1-\mu_i)] \end{aligned}$$

Clearly, the MLE of $\mathbf{w}$ is not in closed form and an optimisation algorithm is needed to compute it.

Structural characteristics:

In logistic regression, a categorical value is associated with each $\mathbf{x}_i$. That is, the status of $\mathbf{x}_i$ is described by a vocabulary of class indices. The mapping $\mathbf{x}_i \mapsto y_i$ in logistic regression also assumes plain vector structure in $\mathbf{x}_i$ and no structural consideration for y_i. The information from $\mathbf{x}_i$ is aggregated by the model into one single categorical value y_i, indicating some specific class rather than a numerical value, as the only difference compared to the output of linear regression.

4.2.2 Vector-based Multi-Output Prediction

Although having been applied widely, the single output setting is limited in its capability of representing the underlying system, either as a descriptive random variable or a latent generative factor. In linear regression and logistic regression, neither a continuous real-valued y_i nor an output with categorical/discrete value representing the belonging class of $\mathbf{x}_i$ has the capability of precisely describing overall status for complex systems. In order to equip predictive models with capabilities of giving a comprehensive and precise overlook of the underlying system, multiple output random variable setting is needed.

When $|\mathbf{y}| > 1$, $\mathbf{y}$ becomes a multi-dimensional output variable. The goal is to learn the joint distribution $p(\mathbf{y}|\mathbf{x})$, given a training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, where N is the number of training examples. We discuss prediction models where $\mathbf{x}$ and $\mathbf{y}$ can be represented by plain vectors, as a direct extension to the single-output prediction models. Because the learning target $p(\mathbf{y}|\mathbf{x})$ is in $\mathbb{R}$, the example discussed here is based on an extension to the classic single-output regression problem.

4.2.2.1 Multi-Output Linear Regression

Recall Eq. 4.1, given $\mathbf{x}$ and parameter $\boldsymbol{\theta}$, $p(y|\mathbf{x}, \boldsymbol{\theta})$ is Gaussian in linear regression. A most straightforward extension to the single-output linear regression to accommodate vector $\mathbf{y}$ as output is the independence assumption between elements $\{y_i \in \mathbf{y}\}_{0 \leqslant i < |\mathbf{y}|}$. Without further modelling any interdependence, $\{y_i \in \mathbf{y}\}_{0 \leqslant i < |\mathbf{y}|}$ are independent random variables. Thus, the joint distribution $p(\mathbf{y}|\mathbf{x}, \mathbf{W})$ is Gaussian, which can be factorized across dimensions as:

$$p(\mathbf{y}|\mathbf{x}, \mathbf{W}) = \prod_{j=1}^{M} \mathcal{N}(y_j|\mathbf{w}_j^T \mathbf{x}_i, \sigma_j^2)$$

where $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_M]$ is the overall weight vector and M is the number of dimensions of $\mathbf{y}$.

Similar to the likelihood factorization, the MLE for $\mathbf{W}$ also factorize across dimensions in the form below:

$$\hat{\mathbf{W}} = [\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_M]$$

where $\hat{\mathbf{w}}_i$ is the MLE for $\mathbf{w}_i$.

Structural characteristics: The independence assumption across dimensions of $\mathbf{y}$ gives a neat analytical form of $p(\mathbf{y}|\mathbf{x}, \mathbf{W})$ and the MLE for $\mathbf{W}$. However, such assumption ignores the inter-relationship between $\{y_i \in \mathbf{y}\}_{0 \leqslant i < |\mathbf{y}|}$. These random variables from each dimension of the plain-vector-based output structure contribute equally and individually to the overall distribution. Actually, the MLE for $\mathbf{W}$ can be obtained by firstly applying the corresponding single-output model separately on each dimension and then combining the results. Thus, in order to build up a more powerful descriptive output structure, we need to further model the possible interdependences to lose the independence assumption by this model.

4.2.3 General Probabilistic Graphical Models

Given the structural limitation identified in single-output and vector-based multi-output prediction models, there is an apparent need for a stronger descriptive power from the output. Recall that the primary goal of structured prediction problem is to model $p(\mathbf{y}|\mathbf{x}$ using the factorized inter-dependencies defined by the graph. Clearly, the way a graph G defines the relations among output variables determines the methodologies for modelling the target distribution. We first brief general methodologies for

modelling structured dependencies in the probabilistic graphical models before fully examining classic ones, e.g., the Markov random field (MRF) and the conditional random field (CRF), etc.

4.2.3.1 Background

Prediction models generally target at computing $p(\mathbf{y}|\mathbf{x})$ with two different possible approaches. One way is to create a joint model of the form $p(\mathbf{y}, \mathbf{x})$ based on the generative relations assumed between $\mathbf{y}$ and $\mathbf{x}$, and then derive $p(\mathbf{y}|\mathbf{x})$ by modelling $p(\mathbf{x})$, which is called the generative approach. An alternative way, namely, the discriminative approach, is to model the marginals needed, in this case $p(\mathbf{y}|\mathbf{x})$, without calculating $p(\mathbf{y}, \mathbf{x})$ or $p(\mathbf{x})$. In this thesis, we adopt term definitions as follows: "graphical model" denotes a family of distributions defined by a graph structure; "random field" or "distribution" denotes a single probability distribution; "network" denotes the graph structure itself.

The probabilistic graph model has been extensively studied in relational learning and many other fields. Under specific assumptions, a graph model is normally able to provide an analytical representation of probability distributions over a set of random variables, including both input and output ones. Given the network composed of the input variable set $\mathbf{X}$ and output variable set $\mathbf{Y}$, the analytical form of the joint or conditional distribution $p(\mathbf{y}, \mathbf{x})$ or $p(\mathbf{y}|\mathbf{x})$, respectively, can often be greatly simplified by making use of local dependencies in the network, and hence the probability computation.

The conditional independence (CI) is the basis for probabilistic graphical models to describe the dependencies. The underlying requirement for the CI property to hold is that edges of a graph representing dependencies are complete. The CI property between two variables only holds when all the depended variables of the current two are visible (text books on machine learning normally provide a good coverage on this topic, e.g. [KF09, Mur12]).

Graphical models are a major group for structured prediction, however, it is worth noting that there are several other models that are also good with basic output structures, e.g. hierarchical/multilevel logistic regression [WM85, Mur12], structured SVM [TGK04, YFRJ07, LJ09]. The term structured regression often appears in recent papers referring models with extensions to classic regression settings [LUW$^+$14, Kim14]. Nevertheless, technically structured regression is a fairly general term, many graphical model based settings with continuous structured output could also be categorised into structured

regression problems, e.g. ordinal-valued label predictions [Kim16, Kim14], continuous CRF [PBX09, BRM14], etc. For a complete introduction to the related models, see [KF09, Mur12].

4.2.3.2 Directed Graphical Model

The directed graphical model is a widely used probabilistic graphical model, which is commonly known as *Bayesian network* [HNC65]. Every node except the root has its parent node because every edge has a direction. The directed graph under the model can be written as $G = \langle V, E \rangle$, where V and E are the node set and edge set, respectively. The directed graphical model essentially provides a direct way for factorization, so that chain rule can be applied according to the directed edges. Thus, we have:

$$p(\mathbf{y}, \mathbf{x}) = \prod_{v \in V} p(v|\pi(v))$$

where $\pi(v)$ are the parents of v in G.

In an *generative model*, nodes representing output variables topologically precede the inputs, that is, no $x \in X$ can be a parent of an output $y \in Y$. Essentially, a generative model is one that directly describes how the outputs probabilistically "generate" the inputs. However, the single-directional pairwise relation normally requires extra modelling in describing the conditional probability. In addition, such relation can potentially over-simplify the relation between a pair of nodes, particularly when there are possibilities for multiple co-existing relations. Thus, in the case of domain knowledge representation, the undirected graphical model is a more suitable choice.

4.2.3.3 Undirected Graphical Model

An *undirected graphical model* represents structural dependencies where there is no topological ordering associated, so the chain rule in the directed graphical model does not suit here to represent $p(\mathbf{y})$ [SM12]. As a consequence, the methodologies for formalizing distributions based on undirected graphical model usually less straightforward. The general strategy for representing the distribution is utilising the conditional independence (CI) depicted by the graph to simplify the number of related clique-indexed potential functions. We examine the related techniques in detail in corresponding sections.

4.2.4 Markov Random Field (MRF)

A Markov random field (MRF), as a class of parametric models, was motivated by the need for methodology improvement in statistical analysis of spatial data [Bes74, Cli90]. It is often also named as Markov network, which is widely used in the domain of probability theory and physics. An MRF is a set of random variables described by an undirected graph complying the Markov property, with its nodes and edges representing the random variables and their inter-dependencies, respectively. Nevertheless, in order to have a factorized representation of the probability distribution, an MRF has additional requirements, which we will examine in detail. MRF is widely studied in applications for low to mid-level tasks in image processing and computer vision [NL11, BKR11, Li12, Liu15]. For a time line and history of the development of mathematical theories for MRF , see [Cli90].

4.2.4.1 The Markov Property

We examine the Markov Property here in detail, given its importance in constructing the numerical probability measure for a random field. The Markov property is actually the cornerstone for bridging the overall probability of a given configuration to a random field and the CI defined by the graphical network. We follow the notations adopted in [Cli90] here.

Let $G = \langle V, E\rangle$ be an undirected graph so that a set of random variables $\mathbf{X} = (X_v)_{v\in V}$ can be indexed by the members of the node set V. E is the edge set and two nodes which form an edge $e \in E$ are said to be *neighbours* of each other. For any subset $Y \subseteq V$ we define the boundary of Y, denoted as ∂Y, by

$$\partial Y = \{x : (x, y) \in E, x \notin Y, y \in Y\}$$

A *clique* is thus defined as a node set that always includes its boundary. Formally, a node set Y in a graph $G = (V, E), Y \subseteq V$ is said to be a clique if and only if

$$Y \subseteq y + \partial y, \forall y \in Y$$

In other words, a clique $Y, Y \subseteq V$ takes the form of either a singleton or a node set where every member of Y is a neighbour of every other member of Y.

There are three forms of Markov property describing dependencies in the undirected graph G:

- Pairwise Markov property: Any two non-adjacent variables are conditionally independent given all other variables: $X_u \perp X_v | X_{V \setminus \{u,v\}}$ if $\{u, v\} \notin E$
- Local Markov property: A variable is conditionally independent of all other variables given its neighbours: $X_v \perp X_{V \setminus \mathrm{cl}(v)} | X_{\mathrm{ne}(v)}$, where $\mathrm{ne}(v)$ is the set of neighbours of v, and $\mathrm{cl}(v) = v \cup \mathrm{ne}(v)$ is the closed neighbourhood of v.
- Global Markov property: Any two subsets of variables are conditionally independent given a separating subset: $X_A \perp X_B | X_S$, where every path from a node in A to a node in B passes through S.

4.2.4.2 Definition of MRF

Recall the previous definition for a *graphical model*, it represents a family of probability distributions defined by the graph structure. In addition to the Markov property, we examine the general conditions for the probability measurement here.

Consider a probability mass function P for the variable set $\mathbf{X}$ indexed by the node set V in an undirected graph G.

- The regularization condition: A direct requirement is $\sum_{\mathbf{x} \in \mathcal{X}} P(\mathbf{x}) = 1$, which is a summation over the possible configuration set $\mathcal{X}$. Thus, the probability of a partial configuration to a node set $A \subseteq V$, denoted as $\mathbf{x}_A$, can be represented as

$$P(\mathbf{x}_A) = \sum_{\mathbf{x}' \in \mathcal{X}} P(\mathbf{x}')$$

  where $\mathbf{x}'$ are all the global configurations in $\mathcal{X}$ that yield a partial configuration $\mathbf{x}_A$.

- The positivity condition: $\forall \mathbf{x} \in \mathcal{X}, P(\mathbf{x}) > 0$. This allows us define logarithmic likelihood

$$Q(\mathbf{x}) = \log P(\mathbf{x})$$

  and the conditional probabilities, such as $\forall A, B \subseteq V$,

$$P(\mathbf{x}_A | \mathbf{x}_B) = \frac{P(\mathbf{x}_A, \mathbf{x}_B)}{P(\mathbf{x}_B)}$$

- The Markov condition: $\forall A \subseteq V, \forall \mathbf{x} \in \mathcal{X}$

$$\frac{P(\mathbf{x})}{P(\mathbf{x}_{V-A})} = \frac{P(\mathbf{x}_{A+\partial A})}{P(\mathbf{x}_{\partial A})}$$

  If A is a singleton set, it turns to the form of local Markov property.

As pointed out in [Cli90], the global and local Markov properties are equivalent to each other. Thus, we directly give the definition of MRF as follows:

Definition 4.1. *Markov Random Field*

An undirected graphical model G is called a Markov Random Field (MRF) if any two nodes are conditionally independent whenever they are separated by evidence nodes and if the associated probability mass function P obeys the above conditions. In other words, for any node X_i in the graph, the following conditional property holds:

$$P(X_i|X_{G\setminus i}) = P(X_i|X_{N_i})$$

where $X_{G\setminus i}$ denotes all the nodes except X_i, and X_{N_i} denotes the neighbourhood of X_i - all the nodes that are directly connected to X_i.

A MRF or a Markov network is often compared with Bayesian network due to their similarity in the representation of dependencies. However, a Bayesian network is directed and **acyclic** while a MRF is undirected and may be **cyclic**. Thus, an MRF can represent certain dependencies that a Bayesian network cannot (e.g. cyclic dependencies), particularly when the relation between nodes are complex such that it is not possible to obtain conditional probability distribution as priors. Besides, the underlying graph of a Markov random field may be **finite or infinite**.

Moreover, there are stronger mathematical results on the relation between being Markovian and the analytical forms of P. We will examine the details in the context of factorisation in corresponding sections.

4.2.4.3 The Log-Linear Form of MRF

Without going further to the details of structural binding feature functions and a number of possible probability mass function Ps, we examine a typical form of an MRF where the overall probability can be factorized in the log-linear form. Given that an MRF should have a strictly positive density, the full-joint distribution is a perfect match for a log-linear model of feature functions $\{f_k\}$ as

$$P(\mathbf{X}=\mathbf{x}) = \frac{1}{Z}\exp\left(\sum_k w_k^T f_k(\mathbf{x}_{\{k\}})\right)$$

where Z is the partition function and $w_k^T f_k(\mathbf{x}_{\{k\}})$ is simply a dot product over field configurations as follows:

$$Z = \sum_{\mathbf{x}\in\mathcal{X}} \exp\left(\sum_k w_k^T f_k(\mathbf{x}_{\{k\}})\right)$$

Clearly, calculating the partition function Z requires an enumeration over $\mathcal{X}$, the set of all possible assignments of values to all the network's random variables. Besides, the local score $w_k^T f_k(\mathbf{x}_{\{k\}})$ can represent multiple relations, with separate weight $w_{k,i}$ for individual feature $f_{k,i}$.

$$w_k^T f_k(\mathbf{x}_{\{k\}}) = \sum_{i=1}^{N_k} w_{k,i} \cdot f_{k,i}(\mathbf{x}_{\{k\}})$$

4.2.5 The Conditional Random Field (CRF)

The model of CRF is a direct extension to the MRF model by conditioning the probability distribution of the output random variables on the whole observations. We adopt the notations in the original paper here [LMP01].

Definition 4.2. *The Conditional Random Field:*

Let $G = (V, E)$ be a graph such that $\mathbf{Y} = (\mathbf{Y}_v)_{v\in V}$, so that $\mathbf{Y}$ is indexed by the vertices of G. Then $(\mathbf{X}, \mathbf{Y})$ is a conditional random field in case, when conditioned on $\mathbf{X}$, the random variables $\mathbf{Y}_v$ obey the Markov property with respect to the graph: $p(\mathbf{Y}_v|\mathbf{X}, \mathbf{Y}_w, w \neq v) = p(\mathbf{Y}_v|\mathbf{X}, \mathbf{Y}_w, w \sim v)$, where $w \sim v$ means that w and v are neighbours in G.

Concretely, the probability distribution of a linear-chain CRF can be defined as follows:

Definition 4.3. *The Linear-Chain CRF:*

Let Y, X be random vectors, $\Lambda = \{\lambda_k\} \in \Re^K$ be a parameter vector, and $\{f_k(y, y', \mathbf{x}_t)\}_{k=1}^K$ be a set of real-valued feature functions. Then a linear-chain conditional random field *is a distribution $p(\mathbf{y}|\mathbf{x})$ that takes the form*

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})}\exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t)\right\} \tag{4.2}$$

where $Z(\mathbf{x})$ is an instance-specific normalization function

$$Z(\mathbf{x}) = \sum_{\mathbf{y}} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, \mathbf{x}_t)\right\} \tag{4.3}$$

The linear-chain CRF is widely used for part-of-speech (POS) tagging problems due to the simplicity of both $\mathbf{X}$ and $\mathbf{Y}$ structures. Actually, the linear-chain CRF has a deep root in the hidden Markov model (HMM) and its feature functions have direct interpretations similar to the ones in the HMM setting. Given that the linear-chain CRF style feature setting takes an important role in almost all the tagging-targeted applications, we first illustrate the transformation from an HMM to a linear-chain CRF and then fully examine the model in the context of generally structured CRF.

4.2.5.1 A Transformation From HMM to Linear-Chain CRF

Recall the original form of an HMM distribution (for more details of HMM, see [Mur12, Bis07]):

It can be factorized as:

$$p(\mathbf{y}, \mathbf{x}) = \prod_{t=1}^{T} p(y_t|y_{t-1})p(x_t|y_t) \tag{4.4}$$

The joint model in (4.4) can be easily rewritten as:

$$p(\mathbf{y}, \mathbf{x}) = \exp\left\{\sum_{t}\sum_{i,j\in S} \lambda_{ij}\mathbf{1}_{\{y_t=i\}}\mathbf{1}_{\{y_{t-1}=j\}} + \sum_{t}\sum_{i\in S}\sum_{o\in O} \mu_{oi}\mathbf{1}_{\{y_t=i\}}\mathbf{1}_{\{x_t=o\}}\right\} \tag{4.5}$$

where the total parameter set of the distribution is $\theta = \{\lambda_{ij}, \mu_{oi}\}$, which is composed of two parts, $\lambda_{ij} = \log p(y' = i|y = j)$ and $\mu_{oi} = \log p(x = o|y = i)$, both can take any real value.

We can write (4.5) more compactly by bringing the concept of *feature functions*. Each feature function has the form $f_k(y_t, y_{t-1}, x_t)$. In order to duplicate (4.5), there needs to be one feature $f_{ij}(y, y', x) = \mathbf{1}_{\{y=i\}}\mathbf{1}_{\{y'=j\}}$ for each transition (i, j) and one feature $f_{io}(y, y', x) = \mathbf{1}_{\{y=i\}}\mathbf{1}_{\{x=o\}}$ for each state-observation pair (i, o). Then we can write an HMM as:

$$p(\mathbf{y}, \mathbf{x}) = \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\right\} \tag{4.6}$$

Then it is straightforward to get the conditional distribution $p(\mathbf{y}|\mathbf{x})$ that results from (4.6) as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{y},\mathbf{x})}{\sum_{\mathbf{y}'} p(\mathbf{y}',\mathbf{x})} = \frac{\exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}, x_t)\right\}}{\sum_{\mathbf{y}'} \exp\left\{\sum_{k=1}^{K} \lambda_k f_k(y'_t, y'_{t-1}, x_t)\right\}} \tag{4.7}$$

Clearly, the conditional probability is already in the form of linear-chain CRF. Several key ideas presented in this transforming process are important to other advanced CRF variants, e.g., skip-chain CRF and dynamic CRF [SM07, SMR07, SRM04, TDF16], etc. For more details on this topic, see [SM12, LMP01].

4.2.5.2 Generally Structured CRF

Without examining the theoretical criteria for factorization in detail, we give the probability distribution in the form of Gibbs measure here [SM12].

Following the notations in the CRF definition,

$$p(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c\in\mathcal{C}} \psi_c(\mathbf{y}_c|\mathbf{x},\boldsymbol{\theta}_c) \tag{4.8}$$

where $\mathcal{C}$ is the set of all the (maximal) cliques of G, $\psi_c(\mathbf{y}_c|\mathbf{x},\boldsymbol{\theta}_c$ is a set of real-valued feature functions indexed by clique c and $Z(\boldsymbol{\theta})$ is the *partition function* given by

$$Z(\boldsymbol{\theta}) \triangleq \sum_{\mathbf{y}} \prod_{c\in\mathcal{C}} \psi_c(\mathbf{y}_c|\mathbf{x},\boldsymbol{\theta}_c) \tag{4.9}$$

It is worth noting that $Z(\boldsymbol{\theta})$ is an instance-specific normalisation function. Without causing confusion, we can write it as $Z(\mathbf{x},\boldsymbol{\theta})$.

4.2.6 Probabilistic Models for Structured Status Prediction along Progression Trajectories

Status prediction is the core of structured prediction problem. The sequential prediction has been widely studied and applied. However, its single-variable per stage setting is limited in status description. Meanwhile, the multi-label model demonstrates superior capabilities in information aggregation and inter-label dependency modelling. Only until very recently, the unified view of these two models attracted interest from the research community[RMH17].

Nevertheless, many real-world prediction problems require even stronger power in status description and modelling. Often, the status of the underlying progression system is extremely difficult to describe without domain knowledge-assisted vocabulary and semantic structures. We were motivated to bring the ontology-based semantic hierarchy to the probabilistic prediction model for structured status description with any given global index. This requires us to fully examine the dependency source along the progression trajectory and inside the semantic hierarchy.

4.2.6.1 The Progression System

It is ubiquitously true that almost every physical system progresses with time. Progression-based observation thus becomes a major source of human knowledge. In machine learning, it is also desirable to enable computing models to capture the characteristics during the course of progression. We now study a special case of supervised learning, where the $(\mathbf{x}_i, \mathbf{y}_i)$ pair is from partial or the whole trajectory of a progression system. It is worth noting that the i.i.d. assumption still hold for $\mathcal{D}$ in this setting.

The importance of the progression system in the context of supervised learning is two-folded:

- The consistency and inter-dependence along the progression trajectory, which provides the basis of model training and prediction.
- The wide existence of such systems in everyday life, e.g., the emotion changes when reading diaries or novels, etc.

4.2.6.2 Probabilistic Models for Sequence-based Progression Trajectories

In sequence-based models, e.g., hidden Markov model (HMM)(Fig 4.1), maximum-entropy Markov model (MEMM)(Fig 4.2) and linear-chain conditional random field (linear-chain CRF)(Fig 4.3), etc., each $y_i \in \mathbf{y}$ is associated with its corresponding state i as status description[Mur12, RMH17]. Although the probability distribution $p(y_i|\mathbf{x})$ or $p(y_i, \mathbf{x})$ varies according to the model, a single-value y_i suffices for the status description in these models. For simplicity, we call it simple status.

Fig 4.4 is an example of a complex progression trajectory, where any stage indexed by some time t can only be represented by a semantic structure. It can be regarded as a direct abstraction to Fig 3.4.

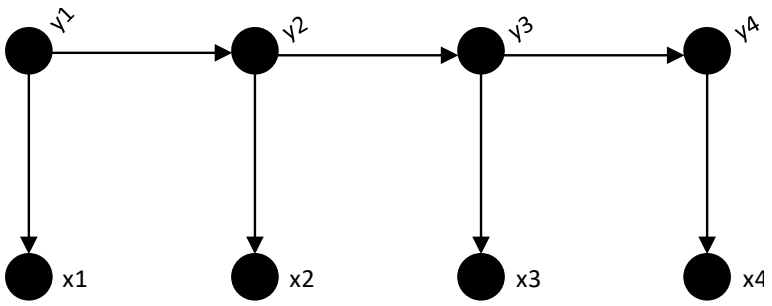

FIGURE 4.1: The hidden Markov model (HMM).

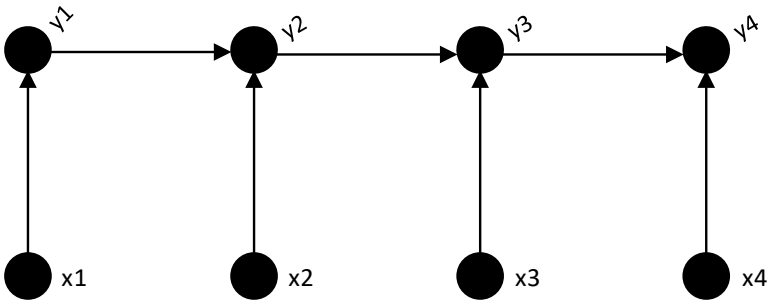

FIGURE 4.2: The maximum-entropy Markov model (MEMM).

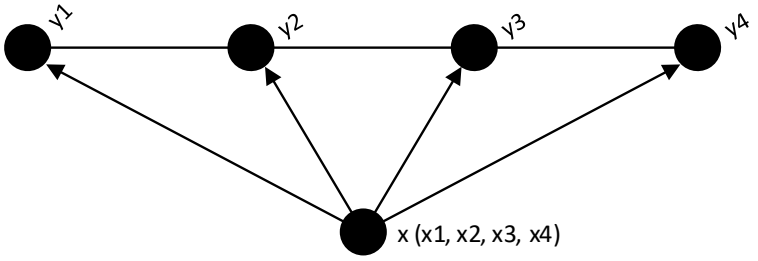

FIGURE 4.3: The linear-chain conditional random field (linear-chain CRF).

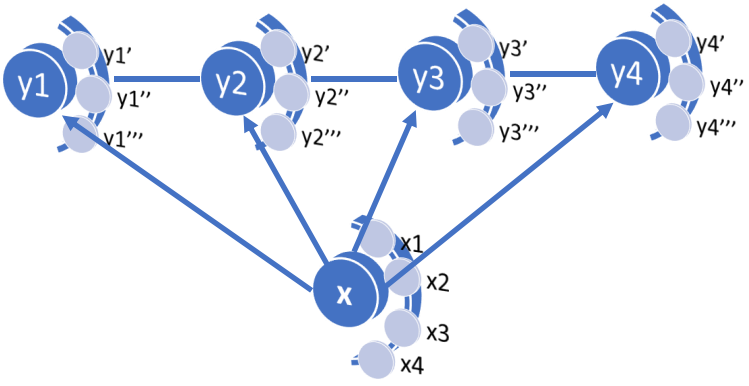

FIGURE 4.4: The CRF with a semantic output structure for structured status prediction along a progression trajectory.

4.2.7 Summary

In this section, we introduce models with single-output and non-structured multi-output settings before fully examining the probabilistic graphical models for structured prediction. The MRF and CRF discussed here provide general frameworks for modelling distributions based on structured dependencies described by undirected graphs. Different sequence-based probabilistic models for trajectory modelling are also discussed. We will go further to address the structural settings in more detail in the next section.

4.3 Feature Functions for Structural Binding and Knowledge Embedding

Graph factorisation is the basis methodology for handling the structural complexity of an undirected graphical model. Although the criteria for judging whether an MRF/CRF is clique-decomposable is clear, there is a surprisingly huge confusion in the literature on a valid setting of feature functions that could truthfully reflect the structural decomposition and the domain knowledge to be embedded.

In this section, we discuss and solve the following two questions:

- What is a valid set of feature functions that can completely and precisely reflect the structural information defined by the graph?
- What is a valid set of feature functions that can completely and precisely embed the domain knowledge into the model of structured prediction?

4.3.1 The Structural Characteristics of the Observable and Latent Variables

In probabilistic graph model, we consider two sets of random variables, a set $\mathbf{X}$ which can be observed before model training and another set $\mathbf{Y}$ which cannot be observed and thus becomes the prediction target. $\mathbf{X}$ is also called the set of *input variables* and $\mathbf{Y}$ is the set of *output variables*.

An MRF constructs the network of both $\mathbf{X}$ and $\mathbf{Y}$ together, which means the dependencies among $\mathbf{X} \cup \mathbf{Y}$ have been fully modelled. For any configuration $(\mathbf{y}, \mathbf{x})$ to the

overall graph, $p(\mathbf{y}, \mathbf{x})$ can be represented by the CI defined by the network. Given the training data, the $p(\mathbf{y}|\mathbf{x})$ for calculating overall log-likelihood requires computing all the possible $p(\mathbf{y}', \mathbf{x})$ and then do a summation. In that sense, although $p(\mathbf{x})$ cannot be computed directly due to the absence of related $\mathbf{y}$, $p(\mathbf{x})$ has been fully modelled and it has a clear analytical form of a summation over all the possible configurations of its boundary variables. In this case, the dependencies are fully modelled by the MRF itself and there is no clear division between $\mathbf{X}$ and $\mathbf{Y}$ in terms of the position inside the graph network.

The CRF, on the other hand, has a clear division between $\mathbf{X}$ and $\mathbf{Y}$. Moreover, the network of a CRF does not model the dependencies between $\mathbf{X}$ and $\mathbf{Y}$ or the intra-dependencies of $\mathbf{X}$ explicitly. Instead, a CRF depicts any $\mathbf{Y}$ distribution by conditioning over the whole observed configuration $\mathbf{x}$ and put more emphasis on the inter-dependencies on the output variable set $\mathbf{Y}$ side. This setting makes it more suitable for building discriminative classifiers.

Some initial work demonstrated some initial discussion on the relationship between MRF and the Gibbs Random Field. [Bes74] provides an alternative proof for the result.

4.3.2 The Feature Function Setting for Structural Binding

Recall the definition of CRF, the output random variables $\mathbf{Y}$ form an MRF and the factorization follows the same form of clique-based decomposition. Thus, the discussions on the MRF factorisation also applies to CRF. We first discuss different forms of the Hammersley-Clifford theorem and then examine the Gibbs measure. Then we conclude the structural requirement on feature functions for factorization by defining the structural binding feature function set, which answers the first question proposed at the beginning of this section.

4.3.2.1 The Hammersley-Clifford Theorem

The relation between a Markovian probability mass function and the form of Gibbs measure has been discussed in some initial work [Spi71, She73]. The equivalence between them was originally proved in an unpublished paper [HC71] and the main result was named after the authors as the Hammersley-Clifford Theorem. An alternative proof was given in an influential work [Bes74]. For an interesting review of the development

of Hammersley-Clifford theorem, see [Cli90]. For more MRF-related mathematical results, see [Spi71, She73, Mou74, Cli90].

It is worth noting that there are several forms of the Hammersley-Clifford Theorem. Reference in the literature often points to the original work in [HC71]. However, the different forms of description actually have minor differences. For example, the original form is in the context of graph colouring problem as follows:

Theorem 4.1. *The Original Form of Hammersley-Clifford Theorem:*

A probability mass function P is Markovian if and only if it can be written in the form

$$P(\mathcal{X})/P(\mathcal{X}_Z) = \exp\left(\sum_{Y \in L_{\mathcal{X}}} Q(\mathcal{X}^Y)\right)$$

where P is a probability mass function, $\mathcal{X}$ is a configuration of light colouring to the overall node set Z, $P(\mathcal{X})/P(\mathcal{X}_Z)$ is the probability of a colouring $\mathcal{X}$ when no additional information is given, $L_{\mathcal{X}}$ is the set of all cliques from the nodes appeared in $\mathcal{X}$, $\mathcal{X}^Y$ is an partial light colour configuration on clique Y and Q is an arbitrary real-valued function of light colourings on cliques.

We can have another form which often appears in recent text books [KF09, Mur12].

Theorem 4.2. *The Hammersley-Clifford Theorem:*

A positive distribution $p(\mathbf{y}) > 0$ satisfies the conditional independence (CI) properties of an undirected graph G iff p can be represented as a product of factors, one per (maximal) clique, i.e.,

$$p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\boldsymbol{\theta}_c) \tag{4.10}$$

where $\mathcal{C}$ is the set of all the (maximal) cliques of G, and $Z(\boldsymbol{\theta})$ is the partition function *given by*

$$Z(\boldsymbol{\theta}) \triangleq \sum_{\mathbf{y}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\boldsymbol{\theta}_c) \tag{4.11}$$

Note that the partition function is what ensures the overall distribution sums to 1.

Note that this form does not really distinguish the factorization according to all the cliques or all the maximal cliques.

Thus, we look back into the proofs, we find that what the Hammersley-Clifford theorem really tells is the equivalence between an Markovian probability mass function P and the Gibbs measure [HC71, Cli90]:

Theorem 4.3. *The Gibbs Measure Form of The Hammersley-Clifford Theorem:*

The Hammersley-Clifford theorem basically states that a distribution that has a positive probability mass or density satisfies one of the Markov properties with respect to an undirected graph G if and only if it is a Gibbs random field, that is, its density can be factorized over the cliques of the graph.

More concretely, the property of being Markovian is equivalent to the ability of being Gibbs measure applicable.

Thus, we fully examine the Gibbs measure next.

4.3.2.2 The Gibbs Measure

The Gibbs measure roots from thermodynamics and thus has a direct entropy style interpretation [Jay65, Mou74, Mur12]. The Gibbs measure for finite systems is actually the vocabulary for describing the probability distribution for all the MRF and its derivatives. Given its importance in all the MRF-based structured prediction models and the confusion often presented in the literature, we feel it is necessary to review the definition of related concepts in measure theory [TJ02, Wei00, Bes74].

We assume the target system is finite, which can be represented by a finite set of nodes S. We also assume that for each node $i \in S$, there is a measure space $(\mathcal{X}_i, \mathcal{F}_i)$ and all the $\mathcal{X}_i$ are finite.

let

$$(\Omega, \mathcal{F}) \triangleq \left(\prod_{i \in S} \mathcal{X}_i, \prod_{i \in S} \mathcal{F}_i \right)$$

equal the product measure space.

On the measure space $(\Omega, \mathcal{F})$, we define an independent reference measure $\lambda = \prod_{i \in S} \lambda_i$ where each λ_i is the uniform measure on $(\mathcal{X}_i, \mathcal{F}_i)$.

Let

$$\mathbb{S} \triangleq \{\Lambda \subset S, \Lambda \neq \emptyset\}$$

be the set of all non-empty subsets of S. For $A \in \mathbb{S}$, let Ω_A and $\mathcal{F}_A$ equal the restriction of Ω and $\mathcal{F}$ to A, respectively. Similarly, ω_A represents the projection of $\omega \in \Omega$ to the set Ω_A.

Now we have the definition for the potential associated with a Gibbs measure.

Definition 4.4. *A potential is a family* $\Phi = \{\Phi_A\}_{A \in S}$ *of functions* $\Phi_A : \Omega \to \mathbb{R}$ *such that*

(1) For each $A \in \mathbb{S}$ *we have* Φ_A *is* $\mathcal{F}_A$*-measurable.*

(2) For all $\Lambda \in \mathbb{S}$ *and* $\omega \in \Omega$ *the energy* $H_\Lambda^\Phi(\omega)$ *exists.*

$$H_\Lambda^\Phi(\omega) \triangleq \sum_{A \in \mathbb{S}, A \cap \Lambda \neq \emptyset} \Phi_A(\omega)$$

Now we can define Gibbs measure on $(\Omega, \mathcal{F})$.

Definition 4.5. *The finite Gibbs measure is defined as*

$$\mu^\Phi(\omega) \triangleq \frac{e^{-H_S^\Phi(\omega)}\lambda(\omega)}{Z_S^\Phi}$$

where

$$Z_S^\Phi = \sum_{\omega \in \Omega} e^{-H_S^\Phi(\omega)}\lambda(\omega)$$

where Z_S^Φ *is called the partition function.*

We review the definition for Gibbs distribution from [Che08], which states clearly about the requirement for clique-based decomposition.

Definition 4.6. *A probability distribution* $P(X)$ *on an undirected graphical model* G *is called a Gibbs distribution if it can be factorized into positive functions defined on cliques* that cover all the nodes and edges of G. *That is,*

$$P(X) = \frac{1}{Z} \prod_{c \in C_G} \phi_c(X_c) \tag{4.12}$$

where C_G *is a set of all (maximal) cliques in* G *and* $Z = \sum_{c \in C_G} \phi_c(X_c)$ *is the normalization constant.*

Essentially, a Gibbs distribution is the result from applying Gibbs measure and it provides a direct criteria for a valid set of feature functions that can equivalently reflect

the structural information in an MRF/CRF. Nevertheless, it also provides possibilities for many different but equivalent ways of factorisation.

It is important to note that the clique set $\mathcal{C}$ should cover all the nodes and edges of G.

4.3.2.3 Summary

We summarize the structural requirement on feature functions for factorization by giving the definition of the structural binding feature function set as follows:

Definition 4.7. *The Structural Binding Feature Function Set:*

Given an undirected graph G of an MRF/CRF, the feature function set $F = f_{C_i}, C_i \subseteq G$ is a structural binding feature function set of G if and only if:

1. *The index cliques $\{C_i\}, C_i \subseteq G$ together provide a complete cover over all the nodes and edges of G*
2. *The feature function f_{C_i} indexed by clique C_i is locally defined on the same clique, and there exists a non-zero clique configuration for f_{C_i}. Concretely, $\exists\ \mathbf{y}_{C_i} \in \mathcal{Y}, s.t.\ f_{C_i}(\mathbf{y}_{C_i}) \neq 0$.*

This definition answers the first question proposed at the beginning of this section. Note that although there are usually multiple clique decompositions to achieve the same coverage, the clique size and the number of overall cliques have direct impacts to time complexity and the effectiveness of the model as well.

4.3.3 The Feature Function Setting for Knowledge Embedding

The feature function setting has a direct relation to the knowledge abstraction. We first examine the forms of feature functions and then formally define the knowledge embedding feature function set.

4.3.3.1 The Forms of Features Functions

Feature functions are normally defined as indicators of the cliques' configuration. for example, if some local configuration $\mathbf{x}_{C_k}^i$ corresponds to the i-th possible configuration of the k-th clique C_k, a feature function $f_{k,i}$ identified by the clique index k and local

configuration index i can be defined as $f_{k,i}(\mathbf{x}) = 1$ when the current input $\mathbf{x}$ matches a local configuration $\mathbf{x}_{C_k}$ and $f_{k,i}(\mathbf{x}) = 0$ otherwise.

There is a direct interpretation to the clique factorisation model given by 4.6. Actually, for a clique C_k and the input $\mathbf{x}_{\mathbf{C_k}}$ indexed by C_k, the weighted sum over all the feature functions $\sum_i w_{k,i} \cdot f_{k,i}(\mathbf{x}_{C_k})$ corresponds to the logarithm of the clique factor $\log \phi_C(\mathbf{x}_C)$.

The types of feature functions or simply features can be generally summarised as follows:

- *A configuration indicator/binary features*: Such feature simply stands for a particular configuration to a subgraph or a clique that indicates a probabilistic relation to some specific category of output. Because such feature is an indicator, the domain of the feature function is simply $\{0, 1\}$.
- *A numerical value*: Such feature is better formalised in contrast to the previous one. No matter its domain is a continuous space or simply discrete numbers, the numerical value stands for different levels of relation to the overall probability. Such feature has already embedded a measure of the degree of dependency. Thus together with its corresponding weight, this type of feature function provides a more fine-grained way to emphasis some particular pattern presented in the observation.

The feature function is allowed to be more general than indicator functions [SM12]. However, label-observation features can result in a large number of parameters in feature engineering, e.g. 3.8 million binary features in the best model of [SM12, SP03]

4.3.3.2 The Source and Semantic Meaning of Features

Recall the definition of the structural binding features functions and the relation-based domain knowledge representation, feature functions have a deep connection to the structure presented in a graph. It is interesting to bridge the inter-dependencies defined by an MRF/CRF and the undirected-graph-based domain knowledge abstraction. It is straightforward to note the connection between cliques in both. Actually, the CI and dependencies are direct abstractions to the relations defined in the knowledge abstraction. Moreover, a feature behind a feature function directly corresponds to the logical relation on the same clique.

Definition 4.8. *The Knowledge Embedding Feature Function Set:*

Given an undirected graph G of an MRF/CRF and an ontology-based domain knowledge $\mathbf{K} = \langle G, \{\mathbf{r}_{\mathbf{C}C \subseteq G}\}\rangle$, a structural binding feature function set $F = f_{C_i}, C_i \subseteq G$ is a knowledge embedding feature function set of G and $\mathbf{K}$ iff

1. *For every clique C_i where these exists a $\mathbf{r}_{C_i}$ from $\{\mathbf{r}_{\mathbf{C}C \subseteq G}\} \in \mathbf{K}$, there exists a C_i-indexed feature function $f_{C_i} \in F$, and vice versa.*
2. *Given any confidence value configuration $\mathbf{y}_{C_i}$ to an index clique C_i, each C_i-indexed feature function $f_{C_i} \in F$ takes value of the numeric score $s \in \mathbb{R}$ given by $\mathbf{K}$, indicating the level of support based on the domain knowledge.*

4.3.4 Summary

In this section, we examine the relation between the feature function setting and the structural and domain knowledge information embedded in a graph. The two questions asked in the beginning of this section were addressed by defining the structural binding and knowledge embedding feature set, which truthfully reflects equivalent information from the graph structure and domain knowledge, respectively.

4.4 The Heterogeneous Input Data and The Locality Preserving Property

In this section, we discuss the challenges posed to the structured prediction model from the heterogeneous input data. We develop the model for the heterogeneous input and then identify a widely existing underlying assumption in almost all the CRF applications - the locality preserving property. We discuss the potential simplification brought by this assumption to the CRF-related model and then propose a general case where the locality preserving property cannot hold. This section addresses the input side as the first part of the development of a novel model.

4.4.1 Background

Feature generating and selecting are important to capturing signals from the observed input data, particularly when the data is heterogeneous. In the EHR prediction literature, the text-based information consists a major part of the input feature set, e.g., decompositions from free-text hospital notes as latent topic-derived features [GNDV$^+$14]. Without the independence assumption, the feature source could be a fairly large range, e.g., the category of the diagnosis [YIN$^+$13], top features with positive coefficients [NKY$^+$15], etc. Nevertheless, we need to further work on modelling the heterogeneous data for structured prediction, particularly for discriminative models.

4.4.2 The Heterogeneous Input Data Modelling

The input space $\mathcal{X}$ of heterogeneous input data $\mathcal{D}$ is infinite because the vocabulary cannot be fixed. Thus, a single observation $\mathbf{x}$ could have various forms. As a result, the structure of $\mathcal{X}$ usually is not able to be captured or abstracted to analytical representations. This unstructured or semi-structured input space $\mathcal{X}$ often makes direct learning $f(\mathbf{x}) : \mathcal{X} \to \mathcal{Y}$ a difficult task. The infinite $\mathcal{X}$ also makes enumeration generally computationally intractable, which brings great difficulties in calculating MRF-style probability distributions when modelling these variables directly.

A natural abstraction is to use a group of random variables $\mathbf{x}'(\mathbf{x})$ as descripting surrogates. A surrogate variable $x'_i(\mathbf{x})$ takes its value from an aspect of $\mathbf{x}$, though it is not necessarily a feature in the sense of knowledge abstraction and representation.

Comparison between a feature and a surrogate variable:

1. A feature in the context of domain knowledge abstraction $\mathbf{K}$ corresponds to a relation defined over a sub-structure, e.g., a clique, however, it is usually difficult to capture the CI relations between $x'_i(\mathbf{x})$ such that a covering graph could be built.

2. A feature in a $\mathbf{K}$ stands for a relation which could give a confidence value to indicate the support to the given configuration based on $\mathbf{K}$, however, the influence from $x'_i(\mathbf{x})$ to the likelihood of given $\mathbf{x}$ cannot usually analytically represented or even captured by a probability mass function p. Concretely, $p(\mathbf{x}|x'_i(\mathbf{x})) \simeq p(\mathbf{x})$, for a given heterogeneous input $\mathbf{x}$.

Clearly, the surrogate $\mathbf{x}'(\mathbf{x})$ set is the vocabulary for describing $\mathbf{x}$. As will demonstrate later, $\mathbf{x}'(\mathbf{x})$ is on a suitable level of abstraction for heterogeneous data, particularly for further feature construction. To address its potential connection to a real feature from a subgraph of an ontology, we call $x'_i(\mathbf{x})$ a *feature fragment*. We adopt the representation of heterogeneous input $\mathbf{x}$ as feature fragments $\mathbf{x}'$. Unless specified individually, we simply use $\mathbf{x}$ to stand for the feature fragment representation of the input throughout this thesis.

The key points of feature fragments $\mathbf{x}$ can be summarised by this definition:

Definition 4.9. *Feature fragment for heterogeneous input model*

A feature fragment is a surrogate variable in $\mathbf{x}'(\mathbf{x})$ *for heterogeneous input* $\mathbf{x}$*, where no structural representation is possible. Its properties can be described as follows:*

- *The CI relation among* $\mathbf{x}$ *is generally limited or unknown. That is, the dependencies inside* $\mathbf{x}$ *cannot be sufficiently modelled and the maximum size of known maximal clique could be very small or even 1.*
- *A feature fragment* $\mathbf{x}_i$ *must be associated to a relation* $\mathbf{r}_j$ *in a domain knowledge abstraction* $\mathbf{K}$ *of an ontology-based output structure to be an influencing factor* $f(\mathbf{x}_i, \mathbf{r}_j)$ *to* $p(\mathbf{y}|\mathbf{x})$*, where* $\mathbf{y}$ *is the structured output in discriminative models, e.g., a CRF.*
- *Each feature fragment* $\mathbf{x}_i$ *or a group of feature fragments* $\mathbf{x}_C$ *indexed by a known clique* C *must be associated with any relation* $\forall\, \mathbf{r}_j \in \mathbf{K}$ *from an ontology-based output structure, because the relation between different parts of* $\mathbf{x}$ *and* $\mathbf{y}$ *is unknown to the model.*

4.4.3 The Locality Preserving Property

Recall the probability distribution of a CRF, every clique-indexed feature function f_C is based on a local configuration $\mathbf{y}_C$ and the global observation $\mathbf{x}$. However, we observe a strong presence of a locality-preserving property of the feature function setting for a local $\mathbf{y}_C$ and the corresponding part of $\mathbf{x}$ in the literature. We examine classic CRF application scenarios in the following, e.g., part of speech (POS) tagging for human languages [SRM04, SC04], image segmentation and tagging [NGL10, ZC12], etc.

- Consider the POS tagging problem, each observation $\mathbf{x}$ is a sentence composed of words and the corresponding sequential order. Note that the possible word set

is generally assumed to be finite and it is possible to enumerate over the word space. Thus, in this scenario the input space $\mathcal{X}$ is not heterogeneous and $\mathbf{x}$ is the original input rather than surrogate feature fragments. Clearly, there is a rigid structure for $\mathbf{x}$. The CI relation can be represented in a form of sequence. On the other side, the output $\mathbf{y}$ is a tagging for $\mathbf{x}$, where each y_i corresponds to a word x_i. Thus, the structure of $\mathbf{y}$ is inherited from $\mathbf{x}$ and there is a direct structural mapping relation between them. Concretely, a x_i has a direct corresponding tag y_i and for $\forall y_i, y_j \in \mathbf{y}, y_i \neq y_j$, y_i and y_j could not map to the same $x \in \mathbf{x}$. That said, two different locations in $\mathbf{x}$ can have the same tag, while two different tags cannot be assigned to the same location in $\mathbf{x}$.

- Consider the image segmentation and tagging problem. The observed image $\mathbf{x}$ is composed of multiple super-pixels $\{x_i\}$. For simplicity, each $y_i \in \mathbf{y}$ takes value from a finite label set indicating the segment type for x_i. Clearly, the structure of $\mathbf{y}$ is a direct inheritance from $\mathbf{x}$, both having been fully modelled. Similarly, one super-pixel x_i cannot have two different segmentation tags $y_i, y_j, y_i \neq y_j$ assigned.

Given its wide appearance in CRF applications, it is worth pointing out that such underlying assumption greatly limits the range of CRF applications. Although it is still a special type of CRF, it counterfeits the claim of global condition on observation in the original CRF definition. We illustrate this in the following.

Given a CRF model $p(\mathbf{y}|\mathbf{x})$, where Y forms an MRF, $p(\mathbf{y}|\mathbf{x})$ is represented by the Y-based factorisation. That is, every feature function f_C for $\mathbf{y}_C$ is indexed by and defined on a clique $C \in Y$, while being conditioned on the global observation $\mathbf{x}$. Thus, this clique C-indexed feature function could be written as $f_C(\mathbf{y}_C|\mathbf{x})$ or simply, $f_C(\mathbf{y}_C, \mathbf{x})$. Clearly, in the original CRF setting, a clique C-indexed local configuration $\mathbf{y}_C$ is not bounded by the association from any local $\mathbf{x}_{\text{sub}}$. Concretely, any local configuration $\mathbf{y}_C$ is an indicator connected with the overall $\mathbf{x}$, instead of a local one.

Recall the features of heterogeneous input data, . This structural mapping cannot hold, because. The locality-preserving property cannot be maintained even for tagging-type problems.

Definition 4.10. *The Locality Preserving Property:*

Given a conditional random field G with input $\mathbf{x}$ and a clique-indexed feature function set $F = \{f_C(\mathbf{y}_C, \mathbf{x})\}$, for any two non-overlapping maximal cliques $\forall C_1, C_2 \subseteq G, C_1 \cap C_2 = \emptyset$, let $\mathbf{x}_1 = \{x_i \mid \exists f_{C_{sub1}}(\mathbf{y}_{C_{sub1}}, x_i \in \mathbf{x}) \in F, C_{sub1} \subseteq C_1\}$ and $\mathbf{x}_2 = \{x_j \mid$

$\exists f_{C_{sub2}}(\mathbf{y}_{C_{sub2}}, x_j \in \mathbf{x}) \in F, C_{sub2} \subseteq C_2\}$, *if* $\mathbf{x}_1 \cap \mathbf{x}_2 = \emptyset$, *the locality preserving property of* G *holds.*

Corollary 4.4. *Given a condition random field* G *whose maximal cliques are all edges, if the input and output set* $\mathbf{x}$ *and* $\mathbf{y}$ *has one-to-one* $(x_e, y_e$ *mapping and the feature function takes the form* $f(y_{e-1}, y_e, x_e)$ *and/or* $f(y_e, x_e)$*, the locality preserving property of* G *holds.*

Proof. Consider any two non-overlapping maximal cliques in the previously described CRF G, (y_{i-1}, y_i) and y_{j-1}, y_j, where $i \neq j, |i-j| \geq 2$, the only x associated with the feature function defined on (y_{i-1}, y_i) and (y_{j-1}, y_j) is x_i and x_j, respectively. Because $i \neq j$, we have $\mathbf{x}_1 = \{x_i\}, \mathbf{x}_2 = \{x_j\}, \mathbf{x}_1 \cap \mathbf{x}_2 = \emptyset$. □

Corollary 4.5. *Given a conditional random field* G *with heterogeneous input data* $\mathbf{x}$ *as feature fragment set, the locality preserving property of* G *does not hold.*

Proof. According to the feature of heterogeneous input data of a CRF G, each feature fragment $\mathbf{x}_i \in \mathbf{x}$ or a group of feature fragments $\mathbf{x}_C \in \mathbf{x}$ indexed by a known clique C must be associated with any relation $\forall \mathbf{r}_j \in \mathbf{K}$ from an ontology-based knowledge abstraction $\mathbf{K}$. Given the one-to-one mapping relation between relation and feature function in $\mathbf{K}$ and G, for any two non-overlapping maximal cliques C_1 and C_2 in G , every $\mathbf{x}_i \in \mathbf{x}$ or $\mathbf{x}_C \in \mathbf{x}$ has an associated $f_{C_1}(\mathbf{y}_{C_1}, \mathbf{x}_i)$ and $f_{C_2}(\mathbf{y}_{C_2}, \mathbf{x}_i)$, respectively. Because $\mathbf{x}_1 \cap \mathbf{x}_2 \neq \emptyset$, the locality preserving property of G does not hold. □

Clearly, the locality is preserved by the projection from $\mathcal{Y} \rightarrow \mathcal{X}$, which implies a strong mapping between the structures on the input and output sides. This property guarantees that the input and output structures could be divided into separated and little-or-non-overlapping parts by single or groups of input and output random variables.

Adopting the same set of index for both $\mathbf{X}$ and $\mathbf{Y}$ is the source of the locality preserving property as discussed in the tagging-based problems, because the index v for $\mathbf{X}_v$ and v' for $\mathbf{X}_v$'s corresponding $\mathbf{Y}_{v'}$ are always local to each other in the same index space if the feature functions are defined accordingly.

4.5 Dependency Source Analysis for Graphical Models

The conditional independence (CI) relation is the basis of probabilistic graphical models. The network or the structure of a graphical model is used to represent different

kind of dependencies. In this section, we discuss the potential sources of dependencies in a graph.

We examine three types of dependencies as the sources of the adjacency in the index space, including temporal-based, position-based indices and a combination of these two. The first two types have strong appearances in the literature, however, the combination type has yet to be sufficiently discussed and addressed.

Being motivated by the EHR prediction problem, where we need to model the connectivity in an ontology and consider the temporal trajectory at the same time, we address the mix model of dependencies here.

4.5.1 Temporal-based Dependencies

The temporal dependency describes the relationship between random variables that can be indexed and sorted according to the temporal order. It is a natural way of modelling the adjacency in the position defined by the index.

Following the previous notations, given a set of input random variables $\mathbf{X}$ and output random variable $\mathbf{Y}$, the dependencies in the graphical network describe the relationships between all the possible pairs $(\mathbf{O}_i, \mathbf{O}_j), \forall i, j\ i \neq j, \mathbf{O}_i \in \mathbf{X} \cup \mathbf{Y}, \mathbf{O}_j \in \mathbf{X} \cup \mathbf{Y}$. For time series, however, the set of input random variables $\mathbf{X}$ is actually generated by $\boldsymbol{x} \in \mathbb{R}^{D_x}$, whose value changes over time $t = \{0, \ldots, T\}$. In this case, $\forall\ \mathbf{X}_i \in \mathbf{X}$ corresponds to the random variable representing the value of $\boldsymbol{x}$ at some time t in some dimension $d, 1 \leq d \leq D_x$. Thus we have $\mathbf{X}_i = f(\boldsymbol{x}, t, d), \forall\ \mathbf{X}_i \in \mathbf{X}, 0 \leq t \leq T, 1 \leq d \leq D_x$. A configuration $\mathbf{y} \in \mathbb{R}^{D_y}$ to the output random variable set $\mathbf{Y}$ can be written as $\mathbf{y} = \{\mathbf{y}_0, \ldots, \mathbf{y}_T\}$, with each $\mathbf{y}_t \in \mathbf{y}$ often associated with the corresponding $\boldsymbol{x}_t$ at some time t rather than some $\mathbf{X}_i \in \mathbf{X}$. However, another commonly seen notation $\mathbf{x}_t$ differs from $\boldsymbol{x}_t \in \mathbb{R}^{D_x}$, a random variable representing the value of $\boldsymbol{x}$ at time t. For some configuration $\mathbf{x}$ to $\mathbf{X}$, $\mathbf{x}_t$ is defined as the components of the global observations $\mathbf{x}$ that are needed by the feature functions at time t. Thus, unlike $\{\mathbf{y}_0, \ldots, \mathbf{y}_T\}$, which is always a division of $\mathbf{y}$, $\mathbf{x}_0 \cup \mathbf{x}_1 \ldots \cup \mathbf{x}_T$ could be an incomplete cover of $\mathbf{x}$ in some cases and there could be some overlapping between some $(\mathbf{x}_t, \mathbf{x}_{t'})$ pair. We note the importance of clarifying this issue, because there has been some confusion in the literature where the relation of $\mathbf{X}$ and $\mathbf{Y}$ has not been made sufficiently clear when applying graphical models to temporal sequences.

It is important though to note the existence of the concept of a step or stage indexed by the current time t in $\boldsymbol{x}$, which makes $\{\boldsymbol{x}_t\}$ a totally-ordered set. Despite the actual

structure of the input random variable set $\mathbf{X}$, the elementary substructure here is $(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t)$ and its corresponding output is $(\mathbf{y}_{t-1}, \mathbf{y}_t)$. Every $\mathbf{y}_t$ is a descriptive variable to depict the status of $\boldsymbol{x}_t$, and the current $\mathbf{y}_t$ has possible dependencies with the current input and all the previous input and output $\{\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \ldots, \boldsymbol{x}_0, \mathbf{y}_{t-1}, \mathbf{y}_{t-2}, \ldots, \mathbf{y}_0\}$.

4.5.2 Position-based Dependencies

The index variable $t = \{0, \ldots, T\}$ in the one-dimensional temporal-based dependencies can be extended to a position-based variable v for a structure, where the adjacencies in the index space $\{v\}$ could represent much more complex dependencies. The position-based index variable v could take value either from $\mathbb{R}^2$ for a dense two-dimensional structure or the node set $V = \{V_0, V_1, \ldots, V_N\}$ for a graph $G = (V, E)$, where E is the edge set describing the connectivity between any node pair $\langle V_i, V_j \rangle, 0 \leq i, j \leq N$. Thus the connectivity of any node pair $< V_i, V_j >$ in G maps to the dependency between random variable pair $(\mathbf{O}_{V_i}, \mathbf{O}_{V_j}), V_i, V_j \in V, \mathbf{O}_{V_i}, \mathbf{O}_{V_j} \in \mathbf{X} \cup \mathbf{Y}$. Different settings of graph G provide convenient ways to describe not only the dependencies between spatially adjacent objects, e.g. the adjacent image segments, but also similarity/dissimilarity in the concept space, e.g. the logically closely-related entities in an ontology.

4.5.3 A Combination of Different Types of Dependencies

As described in Section 4.5.1, in temporal-based dependencies, the random variable $\boldsymbol{x} \in \mathbb{R}^{D_x}$ generates the input random variable set $\mathbf{X}$ so that the relation between $\forall\, \mathbf{X}_i \in \mathbf{X}$ and $\boldsymbol{x}$ can be described by a function $\mathbf{X}_i = f(\boldsymbol{x}, t, d), \forall\, \mathbf{X}_i \in \mathbf{X}, 0 \leq t \leq T, 1 \leq d \leq D_x$. To further our understanding of the dependencies among the output random variable set $\mathbf{Y}$, a generating random variable $\boldsymbol{y} \in \mathbb{R}^{D_y}$ is considered. Similarly, the relation between $\forall\, \mathbf{Y}_i \in \mathbf{Y}$ and $\boldsymbol{y}$ can be described by the function $\mathbf{Y}_i = f(\boldsymbol{y}, t, d), \forall\ \mathbf{Y}_i \in \mathbf{Y}, 0 \leq t \leq T, 1 \leq d \leq D_y$. Suppose $\mathbf{y}$ is a configuration to $\mathbf{Y}$, $\mathbf{y} = \{\mathbf{y}_0, \ldots, \mathbf{y}_T\}$ has a more clearer mapping relation to $\boldsymbol{y}_t$, a random variable representing the value of $\boldsymbol{y}$ at time t. $\mathbf{y}_t$ is actually a configuration direct to $\boldsymbol{y}_t$. In contrast to $\{\mathbf{x}_0, \ldots, \mathbf{x}_T\}$, which are determined by the $X - Y$ dependencies, where $\mathbf{x}_0 \cup \mathbf{x}_1 \ldots \cup \mathbf{x}_T$ could be an incomplete cover of $\mathbf{x}$ and there could be some overlapping between some $(\mathbf{x}_t, \mathbf{x}_{t'})$ pair, $\{\mathbf{y}_0, \ldots, \mathbf{y}_T\}$ is always a strict division of $\mathbf{y}$ and there should be no overlapping between any $(\mathbf{y}_t, \mathbf{y}_{t'})$ pair.

After modelling the output random variable set $\mathbf{Y}$ with a generating time-variant random variable $\boldsymbol{y}$, the position-based dependencies existing in $\boldsymbol{y}$ is often worth considering

for sake of the completeness of the dependencies among $\mathbf{Y}$. Following the notations in Section 4.5.2, we model the position-based dependencies among $\boldsymbol{y}$ by assuming elements in $\boldsymbol{y}$ can be indexed by a position-based index variable v, which takes value either from a Euclidean space $\mathbb{R}^2$ or from a node set $V = \{V_0, V_1, \ldots, V_N\}$ for a graph $G = (V, E)$ where V is the node set and E is the edge set. Because $\mathbb{R}^2$ can be equivalently modelled as the node set V of a locally-connected graph G' with nodes for every grid position in $\mathbb{R}^2$, we only adopt the node-based index variable $v \in \{V_0, V_1, \ldots, V_N\}$ for simplicity.

The combination of the position-based dependencies among $\boldsymbol{y}_v$ and the temporal-based ones among $\boldsymbol{y}_t$ provides great potentials in describing complex inter-dependent and time-variant output space for modelling information-rich structured prediction problems. We adopt this model in EHR prediction problem.

4.5.4 Summary

In this section , we examine the dependency sources of probabilistic graphical models.

We summarize the dependencies presented in the ontology-based structured prediction problem as follows:

- The dependencies among related labels in an ontology
- The dependencies among related variables from the observed data
- The temporal dependencies for labels in an ontology along the temporal trajectories
- The mixed dependencies between cliques from both of the input and output sides.

4.6 Transitional Random Field (TRF)

We see a clear trend in the development of probabilistic graphical models. An MRF models both the input and output set $\mathbf{X}$ and $\mathbf{Y}$ together in the same random field, with fully specified inter-dependencies. A CRF, on the other side, totally splits $\mathbf{X}$ from $\mathbf{Y}$ and only model the inter-dependencies among $\mathbf{Y}$ as an MRF, leaving the structure of $\mathbf{X}$ unknown. This discriminative model brings great benefits and flexibility in setting up the conditioning of $\mathbf{Y}$ on $\mathbf{X}$, however, it also brings chaos. As identified in the previous sections, the classic settings adopted in the literature are mostly not global conditioning.

In this section, we fill the gap by asking these questions:

1. How to describe and define the global conditioning between $\mathbf{y}$ and $\mathbf{x}$?
2. Given the locality preserving property which is widely existing in the CRF literature, what is the methodology for really achieving global conditioning in CRF models?
3. What is the mechanism for passing the information in $\mathbf{x}$ to the output distribution of a globally-conditioned model?

4.6.1 An Extension to the Heterogeneous Input Model

Recall Definition 4.9, despite that the structure of the heterogeneous input $\mathbf{x}$ cannot be sufficiently modelled, due to the lack of CI information and the complex input space $\mathcal{X}$, the connections between $\mathbf{x}$ and $\mathbf{y}$ have been well enhanced. The strong need for association to cliques in $\mathbf{y}$ from $\mathbf{x}$ actually makes a global conditioning to the input from $\mathbf{y}$.

Such relation could be roughly described by : Each feature fragment $\mathbf{x}_i$ or a group of feature fragments $\mathbf{x}_C$ indexed by a known clique C must be associated with any relation $\forall \mathbf{r}_j \in \mathbf{K}$ from an ontology-based output structure, because the relation between different parts of $\mathbf{x}$ and $\mathbf{y}$ is unknown to the model.

In light of this, we propose a special type of CRF with true global conditioning on $\mathbf{x}$ with formalised connection model between $\mathbf{y}$ and $\mathbf{x}$.

4.6.2 The TRF Definition

Definition 4.11. *The Transitional Random Field:*

Given a conditional random field (CRF) $\mathbf{Y}$ and its associated undirected graph $G_{\mathbf{y}} = (V_{\mathbf{y}}, E_{\mathbf{y}})$, where $\mathbf{Y}$ is indexed by $G_{\mathbf{y}}$'s node set $V_{\mathbf{y}}$ as $\mathbf{Y} = (\mathbf{Y}_v)_{v \in V_{\mathbf{y}}}$ and $E_{\mathbf{y}}$ is $G_{\mathbf{y}}$'s edge set, let $G_{\mathbf{x}} = (V_{\mathbf{x}}, E_{\mathbf{x}})$ be an undirected graph on the input variable $\mathbf{X}$ side, such that $\mathbf{X}$ is indexed by the vertices of $G_{\mathbf{x}}$ as $\mathbf{X} = (\mathbf{X}_{v'})_{v' \in V_{\mathbf{x}}}$. Let $C_{\mathbf{y}}$ be a maximal clique in $G_{\mathbf{y}}$, $C_{\mathbf{x}}$ be a maximal clique in $G_{\mathbf{x}}$, and $F = f_C(\mathbf{y}_C|\mathbf{x}) \mid_{C \subseteq G}$ be the total feature function set, the CRF $\mathbf{Y}$ becomes a transitional random field (TRF) if and only if:

- *F is any structural binding feature function set of $G_{\mathbf{y}}$*

- $\forall C_{\mathbf{y}} \subseteq G_{\mathbf{y}}, C_{\mathbf{x}} \subseteq G_{\mathbf{x}}, \exists f_{C'}(\mathbf{y}_{C'}, x \in \mathbf{x}_{C_{\mathbf{x}}}) \in F \mid_{C' \subseteq C_{\mathbf{y}}}$

Recall Definition 4.7, F could be an arbitrary clique-based decomposition to $G_{\mathbf{y}}$ as long as it satisfies the two conditions.

It is worth noting that a graph $G_{\mathbf{x}} = \langle V, E \rangle, V \neq \emptyset, E = \emptyset$ on the input side can still form a TRF with a structured output structure $G_{\mathbf{y}}$. This could actually be an extreme case for heterogeneous input model, where none of the structural information is available and each maximum clique is a single node in $\mathbf{x}$. Nevertheless, it still could be a global conditioning scenario.

Corollary 4.6. *A conditional random field with the heterogeneous input data model is a TRF.*

Proof. Recall the third feature in the definition of the heterogeneous input model 4.9, each feature fragment $\mathbf{x}_i \in \mathbf{x}$ or a clique C-indexed $\mathbf{x}_{C_{\mathbf{x}}}$ is associated to any relation $\forall \mathbf{r}_j \in \mathbf{K}$, where the domain knowledge $\mathbf{K}$-based relation $\mathbf{r}_j$ is defined on all the cliques. Thus, any clique-indexed $\mathbf{x}_{C_{\mathbf{x}}}$, including the maximal clique indexed ones form feature functions with all the cliques in $G_{\mathbf{y}}$. Therefore, such feature function exists for any pair of cliques from both $\mathbf{x}$ and $\mathbf{y}$. This is actually a stronger connection type. Moreover, given all the formed feature functions indexed by cliques from $\mathbf{x}$ and $\mathbf{y}$, a structural binding feature function set can always be extended and developed. Thus, a conditional random field with the heterogeneous input data model is a TRF. □

Theorem 4.7. *A TRF does not have the locality preserving property.*

Proof. The proof is straightforward: $\forall \mathbf{x}_{C_{\mathbf{x}}} \mid_{C_{\mathbf{x}} \subseteq G_{\mathbf{x}}}$, and any two non-overlapping maximal cliques (if not exists, the locality preserving property does not hold automatically) $\forall C_1, C_2 \subseteq G_{\mathbf{y}}, C_1 \cap C_2 = \emptyset$, according to the definition of TRF, $\exists f_{C_1'}(\mathbf{y}_{C_1'}, x \in \mathbf{x}_{C_{\mathbf{x}}}) \in F \mid_{C_1' \subseteq C_1}$ and, similarly, $\exists f_{C_2'}(\mathbf{y}_{C_2'}, x \in \mathbf{x}_{C_{\mathbf{x}}}) \in F \mid_{C_2' \subseteq C_2}$. Thus, $\mathbf{x}_{C_{\mathbf{x}}} \in \mathbf{x}_1$ and $\mathbf{x}_{C_{\mathbf{x}}} \in \mathbf{x}_2$, $\mathbf{x}_1 \cap \mathbf{x}_2 \neq \emptyset$. Thus, a TRF does not have the locality preserving property. □

4.6.3 Features of TRF

Now we can answer the motivation questions as follows:

1. The global conditioning between $\mathbf{y}$ and $\mathbf{x}$ is described and defined by the connection between any pair of maximal cliques from $\mathbf{y}$ and $\mathbf{x}$.

2. Given the proof the locality preserving property does not hold for TRF, being a TRF is a gold standard for global conditioning.

3. Given the classic definition of a CRF (a TRF is still a special type of CRF), the probability given by the model $p(\mathbf{y}|\mathbf{x})$ is factorized according to $G_{\mathbf{y}}$. Given that the feature functions defined over cliques are having corresponding $\mathbf{x}$ configurations as parameter(s), the probability mass is inevitably influenced by $\mathbf{x}$. This is the mechanism for passing information.

We now can have a brief summary over the features of TRF:

- A loose of clique size limitation: recall that the feature function set F could be any structured binding feature function set of $G_{\mathbf{y}}$, the TRF retain the full capability of the expressing power from an arbitrary clique. Despite that, it is worth noting that it is not compulsory to cover all the large-sized cliques.
- A true global conditioning over $\mathbf{x}$, with formalised methodologies.
- A wider range of applications for structured prediction, where a global conditioning is necessary.

4.7 Conclusion

In this chapter, we address the problem of probabilistic modelling, given the formalised problem modelling from previous chapters. Being the basis of a probabilistic graphical model, the feature functions are the real vocabulary for representing the structural information defined by a graph. Given the surprisingly amount of confusion in the literature, we fully examine the structural equivalence between a set of feature functions and the structure and propose the criterial for structural binding feature function set. Moreover, by bridging the relation in semantic knowledge abstraction and the feature function, we propose the definition of knowledge embedding feature function for semantic equivalent knowledge embedding. After discussions on the features of heterogeneous input and the dependency source on the output side, we propose the concept of TRF to lose some vital limitations to classic CRF models. This inevitably will impact the model training and inference process, which we will examine in next sections.

5

The Transitional Random Field (TRF): Inference and Estimation

5.1 Introduction

The structural setting for TRF proposed in the last chapter stands for a general case of CRF abstracted from a majority real-world prediction problems. To address the efficiency problem of traditional algorithms, we develop related methodologies to make efficient training and inference possible for highly-connected graphical models.

In this chapter, we first discuss the challenges from this class of problems in the context of training and prediction, and then progressively develop novel methodology to tackle these problems.

To capture the pairwise relations implied/fixed by the model, we also develop a novel similarity measure. It turns out that this new measure has multiple benefits to model learning, in addition to measuring the similarity between heterogeneous object pair itself. Thus, a training algorithm can be built based on it. Actually, similarity measure plays a critical role in the era of data explosion, not only because it is a basis of the traditional data querying and clustering techniques, but also because it is a decisive metric over the entire data topological structure, which stores huge information. However, the similarity measure is often not straightforward especially when considering data with heterogeneous structures and features. Moreover, the loss of certainty in time series data with complex structure and conspicuous noise makes defining similarity even more difficult. Thus, probabilistic models with adaptive and expressive structures are more capable of understanding the inner structure of such data set. In

this paper, probabilistic graph model, specifically Conditional Random Field (CRF) is adopted to solve the similarity measure learning problem over the real world Electronic Health Record(EHR) data.

The discriminative conditional random field is an expressive model to connect the topologically indexed output random field to the observed input data. The difficulty in training and inferencing over general structured probabilistic graphs has greatly limited the application of CRFs to real world problems. Although accelerated sampling based inference method is possible, as examined in this paper, the deterministic approaches are still the optimal solution. The feature functions have intrinsic connections with the graph structure. Thus, a further exploitation over the feature functions results a novel similarity measurement as well as a fast deterministic training algorithm.

The traditional methods for inference also do not fit in our TRF model. It either oversimplified the problem by making assumptions on the distribution (e.g. Gaussian) or rely on special sub-structures (e.g. chordal graph). Moreover, there lacks the method for inferencing over continuous output space without the distribution space.

In the ontology-based prediction problem with a semantic hierarchy as the vocabulary, we identify the inference target, a group of predictions. Thus, we develop a straightforward method for inferring the result. Experimental results prove its effectiveness and efficiency, as will be demonstrated in the experiment chapter.

The contributions in this chapter can be summarised as follows:

1. Examining the existing training and inference algorithms for probabilistic graphical models do not fit for TRF, thus a novel training and inference framework is needed.

2. Propose a semantic hierarchy-based novel similarity measurement, such that heterogeneous input data can be put into a rigid yet comprehensive descriptive structure.

3. Discuss the benefits of the proposed similarity measurement and provide theoretical proofs for the effectiveness and its efficiency.

4. Develop a novel training framework to solve the additional training and inference problem for the whole MRF/CRF family.

5. Develop a novel semantic feature-based inference algorithm with the ability to fully utilise the ontology-assisted domain knowledge in the hierarchy.

5.2 Background

Given the formal definition of the TRF, we need to utilise the probability distribution defined by the model to conduct training, inferencing and distance calculating efficiently. To fully exploit the potential of the TRF model, we first examine the linear-chain structure in a CRF as background and then discuss the variations and challenges from TRF. For more details on this topic, see [Mur12].

5.2.1 The General Setting

For a general training problem, we have a training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where N is the number of training examples and the hypothesis h is represented by the model. The two major target estimations are the maximum a posteriori (MAP) and the maximum likelihood estimate (MLE), which can be formally written as the following:

$$\hat{h}^{MAP} = \arg\max_h p(\mathcal{D}|h)p(h)$$

$$\hat{h}^{mle} = \arg\max_h p(\mathcal{D}|h) = \arg\max_h \log p(\mathcal{D}|h)$$

Clearly, the core part of the MAP or MLE is $\log p(\mathcal{D}|h)$, which is the log-likelihood of the data set $\mathcal{D}$, given the hypothesis/model. Thus, a rewritten to a log-linear form $\sum \log p(\mathbf{y}_i)$ is more straightforward because normally $p(\mathbf{y})$ has analytical forms by the model.

5.2.1.1 Linear Chain CRF Training

We first examine the training process for a linear chain CRF based on unary and pairwise elementary decomposition for obtaining the MLE of the parameter $\boldsymbol{\theta}$.

Concretely, the parameter estimation method aims at estimating the parameters $\theta = \{\lambda_k\}$ of a linear-chain CRF from the given independent and identically distributed (iid) training data $\mathcal{D} = \{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^N$, where each $\mathbf{x}^{(i)} = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_T^{(i)}\}$ is a sequence of inputs, and each $\mathbf{y}^{(i)} = \{y_1^{(i)}, y_2^{(i)}, \ldots, y_T^{(i)}\}$ is a sequence of the desired predictions.

Given that a CRF is a discriminative model, the conditional probability is directly modelled. Thus, we can have this *conditional log likelihood* form as follows:

$$\ell(\theta) = \sum_{i=1}^{N} \log p(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}) \tag{5.1}$$

Recall that the linear CRF has a direct decomposition over edges, which are essentially their maximal cliques. Thus, we have a neat form of decomposition based on feature functions defined on edges and/or nodes.

$$\ell(\theta) = \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{k=1}^{K} \lambda_k f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) - \sum_{i=1}^{N} \log Z(\mathbf{x}^{(i)}) \tag{5.2}$$

In general, the function $\ell(\theta)$ cannot be maximized in closed form, so numerical optimization is used. The partial derivatives are:

$$\frac{\partial \ell}{\partial \lambda_k} = \sum_{i=1}^{N} \sum_{t=1}^{T} f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) - \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{\mathbf{y}^{(i)}} f_k(y_t^{(i)}, y_{t-1}^{(i)}, \mathbf{x}_t^{(i)}) p(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}) - \frac{\lambda_k}{\sigma^2} \tag{5.3}$$

The function $\ell(\theta)$ is concave, thanks to the general convexity of functions of the form $g(\mathbf{x}) = \log \sum_i \exp x_i$.

5.2.1.2 Training for Generally Connected CRF/MRF

Clearly, the simplest form of decomposition is based on linear structured feature functions. Concretely, the number of feature functions is linear to the length of the target sequence. However, the definition of TRF allows a feature function set with arbitrary clique size setting. Thus, the number of feature function indexed by cliques is potentially much higher. Actually, this is exact the same problem in the training problem for generally connected CRF/MRF models.

We first examine the log-linear form of an MRF:

$$p(\mathbf{y}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_c \boldsymbol{\theta}_c^T \phi_c(\mathbf{y})\right)$$

where c is the clique index.

the scaled log-likelihood is thus:

$$\ell(\boldsymbol{\theta}) \triangleq \frac{1}{N} \sum_i \log p(\mathbf{y}_i|\boldsymbol{\theta}) = \frac{1}{N} \sum_i \left[\sum_c \boldsymbol{\theta}_c^T \phi_c(\mathbf{y}_i) - \log Z(\boldsymbol{\theta})\right]$$

Because MRFs are in the exponential family, this log-likelihood function is convex in $\boldsymbol{\theta}$. Thus, there exists a unique global maximum so that we can apply gradient-based optimizers.

The derivative for the weights of a particular clique c as input to the optimizer:

$$\frac{\partial \ell}{\partial \boldsymbol{\theta}_c} = \frac{1}{N} \sum_i \left[\phi_c(\mathbf{y}_i) - \frac{\partial}{\partial \boldsymbol{\theta}_c} \log Z(\boldsymbol{\theta}) \right]$$

The CRF model, on the other side, is a discriminative one. We follow the conditional distribution representation as follows:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{\Psi_A \in G} \exp \left\{ \sum_{k=1}^{K(A)} \lambda_{Ak} f_{Ak}(\mathbf{y}_A, \mathbf{x}_A) \right\} \tag{5.4}$$

Note that now the feature functions are indexed by cliques, thus the number of feature functions could be exponential if we enumerate all the possible cliques, while this setting is still a valid decomposition. Given the simplicity of log-linear form, the *conditional log likelihood* can be written as:

$$\ell(\theta) = \prod_{C_p \in \mathcal{C}} \prod_{\Psi_c \in C_p} \sum_{k=1}^{K(p)} \lambda_{pk} f_{pk}(\mathbf{x}_c, \mathbf{y}_c) - \log Z(\mathbf{x}) \tag{5.5}$$

The partial derivatives are:

$$\frac{\partial \ell}{\partial \lambda_{pk}} = \prod_{\Psi_c \in C_p} f_{pk}(\mathbf{x}_c, \mathbf{y}_c) - \prod_{\Psi_c \in C_p} \sum_{\mathbf{y}'_c} f_{pk}(\mathbf{x}_c, \mathbf{y}_c) p(\mathbf{y}'_c|\mathbf{x}) \tag{5.6}$$

Clearly, in the training process, we need to enumerate all the possible configuration $\mathbf{y}'_c$ in calculating the partial derivatives, which is exponential to the size of the clique c. In addition, inferring the value for $Z(\mathbf{x})$ also requires an enumeration over all the possible $\mathbf{y}$ over the whole graph, for even one $\mathbf{x}$. This involves the inference problem, which we examine in the following section.

5.2.1.3 The CRF/MRF Inference

There are two common inference problems for CRF/MRFs. The first inference task happens during training, as discussed, computing the gradient requires marginal distributions for each clique $p(\mathbf{y}'_c|\mathbf{x})$ and computing the likelihood requires inferring the

value of $Z(\mathbf{x})$. Moreover, this calculation is required for every step of optimization, which makes it computationally intractable.

The second type of inference is the labelling process in $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$, e.g., the most likely (Viterbi) labelling. Because the inference is mostly about how to compute for every possible configuration $\mathbf{y}'$. A naive inference method is to enumerate all the possible $\mathbf{y}'$. Thus, in linear-chain CRFs, both inference tasks can be performed efficiently and exactly by variants of the standard dynamic-programming algorithms for HMMs due to the linear structure. For generally connected MRF/CRF, however, this exact inference problem is a $\#P$-complete problem. Concretely, the time complexity of such algorithm is in $\mathbf{\Theta}(|\boldsymbol{Y}|^n)$, $|\boldsymbol{Y}|$ is the size of $\mathbf{Y}$ space and n is the length of possible $\mathbf{y}'$. Clearly, the naive inference is computational intractable. Because a TRF is also a CRF, its inference is also a $\#P$-complete problem. For more details on this topic, see [Mur12].

5.2.2 Approximations

Given the complexity in exact inference and the overall training and prediction framework, sampling-based method attracted wide interests in the research community and the primary target is to approximate the value of partition function.

5.2.2.1 The Partition Function

The partition function appeared in the MRF and CRF definitions roots from statistical physics and plays an important role in probabilistic modelling and machine learning [Mur12].

However, computing the partition function is normally computationally expensive and it is a major issue with all the Gibbs measure based models, e.g. MRF and CRF.

As introduced in Theorem 4.2, the partition function for discrete output random variables in MRF takes the form of

$$Z(\boldsymbol{\theta}) \triangleq \sum_{\mathbf{x}} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\boldsymbol{\theta}_c)$$

We can also have a more general definition for discrete outputs as:

Definition 5.1. *Given a set of discrete random variables* $\mathbf{X}$ *taking on values* $\{\mathbf{x}\}_i$*, and some sort of potential function or Hamiltonian* $H(x_1, x_2, \ldots)$*, the partition function for* X_i *is defined as:*

$$Z(\beta) = \sum_{\mathbf{x}_i} \exp(-\beta H(x_1, x_2, \ldots))$$

The sum over the x_i is understood to be a sum over all possible values that each of the random variables X_i may take.

5.2.2.2 Efficient Inference by Sampling

A distribution $Q(\mathbf{X})$ is calculated as the best approximation to the true probability distribution $P(\mathbf{X}|\mathbf{I}, \boldsymbol{\theta}) = \frac{1}{Z(\mathbf{I},\boldsymbol{\theta})} \exp(-E(\mathbf{X}|\mathbf{I}, \boldsymbol{\theta}))$ of the model. $Q(\mathbf{X})$ is assumed to be a product of independent marginals over each of the variables $Q(\mathbf{X}) = \prod_i Q_i(X_i)$ with the constraint that $\sum_{x_i} Q_i(X_i) = 1, Q_i(X_i) \geq 0$.

For example, To make an estimation for $\sum_{\mathbf{y}'} p(\mathbf{y}'|\mathbf{x}_t) \times f_j(\mathbf{y}', \mathbf{x}_t)$, if we write:

$$p(\mathbf{y}'|\mathbf{x}_t) = \frac{\exp[\text{score}(\mathbf{y}'|\mathbf{x}_t)]}{Z(\mathbf{x}_t)} = \frac{\tilde{p}(\mathbf{y}')}{Z_p}$$

some other distribution $q(\mathbf{y}')$

$$q(\mathbf{y}') = \frac{\tilde{q}(\mathbf{y}')}{Z_q}$$

if we write:

$$\tilde{r}^{(s)} = \frac{\tilde{p}(\mathbf{y}^{(s)})}{\tilde{q}(\mathbf{y}^{(s)})}$$

then:

$$\frac{Z_p}{Z_q} \approx \frac{1}{S} \sum_s \tilde{r}^{(s)}$$

Thus, the sum over all clique configurations $\mathbf{y}'$ can be approximated by:

$$\begin{aligned}
\sum_{\mathbf{y}'} p(\mathbf{y}'|\mathbf{x}_t) \times f_j(\mathbf{y}', \mathbf{x}_t) &\approx \frac{Z_q}{Z_p} \frac{1}{S} \sum_{s=1}^{S} f_j(\mathbf{y}^{(s)}, \mathbf{x}_t) \frac{\tilde{p}(\mathbf{y}^{(s)})}{\tilde{q}(\mathbf{y}^{(s)})}, \mathbf{y}^{(s)} \sim q(\mathbf{y}) \\
&\approx \frac{1}{S} \sum_{s=1}^{S} f_j(\mathbf{y}^{(s)}, \mathbf{x}_t) \frac{\tilde{r}^{(s)}}{\frac{1}{S}\sum_s \tilde{r}^{(s)}} \\
&= \frac{\sum_{s=1}^{S} f_j(\mathbf{y}^{(s)}, \mathbf{x}_t) \cdot \tilde{r}^{(s)}}{\sum_s \tilde{r}^{(s)}}
\end{aligned}$$

Thus, both the two inferences can be calculated efficiently by sampling according to some arbitrary distribution $q(\mathbf{y})$ without enumerating all the possible $\mathbf{y}'$.

5.2.3 Summary: Problems with Sampling-Based Methods

The sampling-based method seems attractable, particularly when the CRF/MRF is fully connected, which makes the gradient computation require performing exact inference over all the possible distributions for very large cliques [KK13].

The approximation based methods are studied extensively by the community [Bis07, SM12], however, as pointed out by [SM12], the numerical value tends to be very sensitive to the divisor $Z(\mathbf{x})$. It is important to note that there can be complex inter-actions between the inference procedure and the parameter estimation. Actually, putting the approximate inference in the subprocess of numeric optimization could the whole training process much more unpredictable. Moreover, we will demonstrate that the traditional training and prediction framework does not fit the TRF due to its special input and output settings.

5.3 The Structural Challenges from TRF

Given the discussions in previous sections, the traditional training and prediction framework is problematic for generally connected CRF/MRFs. This significantly affects the application of these types of models due to the structural limitation posed by training and inference efficiency problems.

In this section, we first examine the variations TRF could bring to traditional CRF/MRF-based methods. The potential challenges to existing learning and prediction methods motivate us to further exploit the possibilities in developing novel methodologies for this new model.

5.3.1 The Heterogeneous Input

We developed the heterogeneous input model in the context of probabilistic dependencies from $\mathbf{x}$ to $\mathbf{y}$. Nevertheless, it also affects traditional MRF/CRF-based algorithms.

Given a heterogeneous input $\mathbf{x}$, a direct result is the input space $\mathcal{X}$ cannot be enumerated. Thus, for an MRF, the value of partition function $Z(\mathbf{x})$ cannot be calculated simply by enumerating over possible configurations.

The change of the input space $\mathcal{X}$ also affects the potential feature functions. In both a CRF and an MRF, a feature function should put the input space $\mathcal{X}$ into its domain in

order to generate feature scores. This is particularly true for indicator-based features, where features are directly based on the value of inputs.

5.3.2 Number of Indexing Cliques

Recall that a TRF only requires the feature function set F to be structural binding set, which means the number of indexing cliques could be large. Consider the feature functions could also be indexed by a pair of cliques, the potential number is even higher. Putting aside the possible affection the training and inference process. A shear increase of indexing cliques only results an increase of the length of weight vectors. Nevertheless, numeric optimizers are often sensitive to the length of weight vector during training.

5.3.3 Size of Indexing Cliques

The size of a clique is important to traditional training and inference methods. Because in an MRF, we need to enumerate over all the possible configurations for the input $\mathbf{x}$ to calculate the value of the feature functions and then sum them over, the number of the possible $\mathbf{x}$ configurations is a major factor in estimating the complexity in calculating the partition function $Z(\mathbf{x})$.

5.3.4 The Partition Function with Continuous Input

Given a set of continuous random variables X_i and some sort of potential function or Hamiltonian $H(x_1, x_2, \ldots)$, the partition function for X_i is thus defined as:

$$Z(\beta) = \int \exp(-\beta H(x_1, x_2, \ldots)) \,\mathrm{d}\, x_1 \,\mathrm{d}\, x_2 \cdots$$

5.3.5 Summary

Given the above discussions, without going further to the details of existing training and inference techniques, we exploit other possibilities for efficient training and inference.

5.4 Similarity and Distance for Heterogeneous Data

Distance and *similarity* represent a quantitative degree of how far apart or how near two objects are, respectively. The choice of distance/similarity measures depends on the

extent to which we can precisely represent and abstract target objects. In this section, we discuss available distance/similarity measures for different levels of abstractions with a special consideration to the applicability to heterogeneous data.

Given the complexity in training the structured prediction models for heterogeneous data, in this section, we consider using a similarity measurement $s(\mathbf{x}_1, \mathbf{x}_2)$ for two heterogeneous observations $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, where $\mathcal{X}$ is an infinite input space, to learn some basic topological information. We first examine how similarity measurements work and then develop a novel similarity measurement for heterogeneous data.

5.4.1 The Projected Space for Similarity Measurement

A similarity measure $s(\mathbf{x}_1, \mathbf{x}_2)$ actually projects the given input to a target space where it can easily define the concept of distance. Clearly, there are numerous possibilities for constructing such space. We examine several major types in the following.

5.4.1.1 Norm/Form-Based Similarity

Norm/form-based similarity measurements try to define similarity based on representations, e.g., the norm of feature vector, matching level of structural features, etc. The implication is that the norm or form of a representation reflects some location information in the projected space. There are a huge amount of work on this. For a review on this topic, see [SWS+00]. However, because this method lacks of semantic interpretation, it generally does not fit for heterogeneous data, due to the infinite input space $\mathcal{X}$.

5.4.1.2 Probabilistic Similarity Measurement

There are many probabilistic similarity measures that have been proposed in the literature. For a recent survey see [Cha07]. In this type of method, the abstraction of $\mathbf{x}_1$ and $\mathbf{x}_2$ to random variables actually simplifies the problem. By making use of the toolset for measuring probability distributions, one can directly measure the similarity in terms of statistical features. However, none of these methods could be directly applied in our approach, so we defined a new similarity measure to fit the particular requirements of our model in Section 5.2.

5.5 The TRF-based Similarity

Given the difficulties in directly measure the topological-related norm of heterogeneous data, we seek an approach that could

1. reflect the similarity with consideration to semantic meanings.
2. utilise numerical values to indicate the similarity in a precise way.

In this section, we first examine an interesting CRF-based similarity. Then we develop a new framework for the heterogeneous data.

5.5.1 CRF based Similarity

In [MBP05], CRF were applied to measure the distance/dissimilarity between strings. The conditional probability of some alignment a given two strings $\mathbf{x}$ and $\mathbf{y}$ can be formally defined in the form of a variation to the traditional CRF as $p(\mathbf{a}|\mathbf{x},\mathbf{y}) = \frac{1}{Z_{\mathbf{x},\mathbf{y}}} \prod_{i=1}^{|\mathbf{a}|} \Phi(a_{i-1}, a_i, \mathbf{x}, \mathbf{y})$. However, despite CRF itself is a probabilistic model, this method falls in the category of exacting matching based similarity measures, thus cannot be applied to the heterogeneous data.

However, this methods provides some interesting insights as described in the following:

1. Given the difficulties in measuring the similarity between $\mathbf{x}$ and $\mathbf{y}$ directly, an alignment $\mathbf{a}$ is utilized as a descriptive object to indicate how match the two inputs are.
2. The structured prediction model adopted actually projects the information between $\mathbf{x}$ and $\mathbf{y}$ to the output structure. For example, the output distribution varies according to different $\mathbf{a}$ value, thus forms a distribution as prediction.

5.5.2 A New Probabilistic Measurement Scheme

Consider a discriminative probabilistic graphical model, e.g., a CRF, the conditional probability distribution $p(\mathbf{y}|\mathbf{x})$ for any $(\mathbf{x},\mathbf{y})$ pair is given by the model, it the well-defined analytical form. Such mechanism actually bridges $\mathbf{x}$ and $\mathbf{y}$ in a compact yet divisible way. The conditional probability connects these two parts of variables, however, they are still separable. Thus, it is interesting to exploit further potentials from

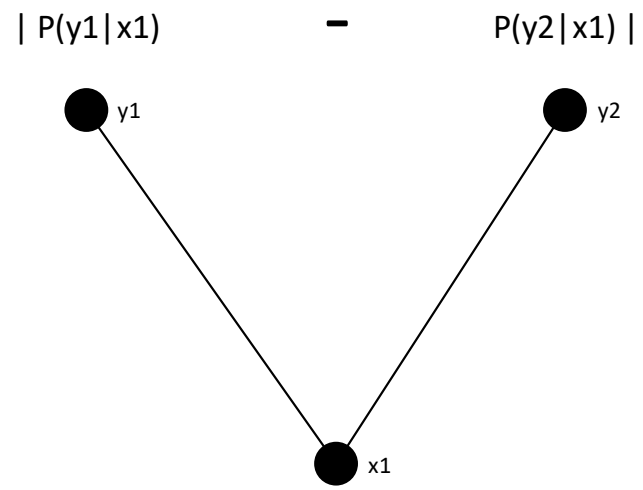

FIGURE 5.1: Using the output configuration as a descriptive structure for a given input in a fixed-structure discriminative model.

this discriminative setting. *Example*: Let us consider a fixed structured CRF model, where the output structure is fixed no matter what the input $\mathbf{x}$ is (a counter example is a linear chain CRF model in the sequence tagging problem, where the model outputs a $\mathbf{y}$ configuration for various-length input sequence with different but similar structures). More concretely, for a group of fixed-length input sequences/sentences, the output $\mathcal{Y}$ space consists all the possible $\mathbf{y}$s with a fixed length as well. Recall the probability mass function definition in CRF, $\forall \mathbf{y} \in \mathcal{Y}$ we have $p(\mathbf{y}|\mathbf{x}) > 0$ for any possible $\mathbf{x}$. In another word, $\forall \mathbf{y} \in \mathcal{Y}$ is possible of becoming the right representation of the input $\mathbf{x}$, as depicted in Fig 5.1. Thus, for two given pairs of input and output configuration $(\mathbf{x}_i, \mathbf{y}_i)$ and $(\mathbf{x}_j, \mathbf{y}_j)$, the configurations to the output structure $\mathbf{y}_i$ and $\mathbf{y}_j$ provide structural interpretations to $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. In this case, the difference in values when using one output configuration, e.g., $\mathbf{y}_i$ or $\mathbf{y}_j$ for another input, e.g., in this case, $\mathbf{x}_j$ or $\mathbf{x}_i$, gives an insight to the difference between $\mathbf{x}_i$ and $\mathbf{x}_j$.

Now let us examine the meanings of these four pairs: $p(\mathbf{y}_i|\mathbf{x}_i)$, $p(\mathbf{y}_j|\mathbf{x}_i)$, $p(\mathbf{y}_i|\mathbf{x}_j)$ and $p(\mathbf{y}_j|\mathbf{x}_j)$. With the assumption that these conditional probabilities are from a fixed-structured CRF, $\mathbf{y}_i$ and $\mathbf{y}_j$ are valid candidate structured output configurations for any $\mathbf{x}$.

Thus, $|p(\mathbf{y}_i|\mathbf{x}_i) - p(\mathbf{y}_i|\mathbf{x}_j)|$ can partially reflect the difference between $\mathbf{x}_i$ and $\mathbf{x}_j$. The same is with $|p(\mathbf{y}_j|\mathbf{x}_i) - p(\mathbf{y}_j|\mathbf{x}_j)|$ for $\mathbf{x}_i$ and $\mathbf{x}_j$ in the sense of $\mathbf{y}_j$. Such difference will have more sense if $(\mathbf{x}_i, \mathbf{y}_i)$ and $(\mathbf{x}_j, \mathbf{y}_j)$ are already the most-likely pairs, where $p(\mathbf{y}_i|\mathbf{x}_i)$ and $p(\mathbf{y}_j|\mathbf{x}_j)$ get their maximals given the model. Thus, without going to much to details here, we only point out that $\forall k, \frac{p(\mathbf{y}_k|\mathbf{x}_i)}{p(\mathbf{y}_i|\mathbf{x}_i)}$ represents the percentage $\mathbf{y}_k$ could

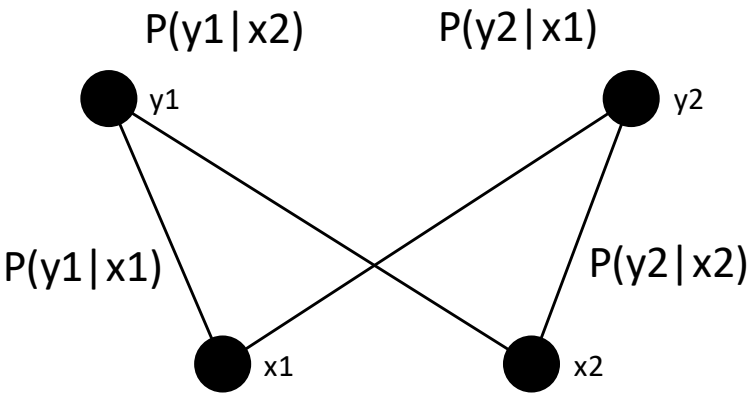

FIGURE 5.2: Using two output configurations as descriptive structures with a cross-representation for two given inputs in a fixed-structure discriminative model.

achieve in terms of representing $\mathbf{x}_i$ compared to $\mathbf{x}_i$'s optimal/most likely representation $\mathbf{y}_i$.

5.5.3 Semantic Hierarchy based Similarity

It is worth nothing the assumption that the criminative model being a fixed-structured one is difficult for real-world problems, particularly for heterogeneous data. For this type of structured prediction problem, a fixed output semantic hierarchy provided by the ontology is important in constructing such descriptive structure. Recall the domain knowledge abstraction and embedding, an ontology-based hierarchy is used as the descriptive output structure for ontology-assisted structured status prediction. In this way, the input $\mathbf{x}$ can be projected into a conditional probability space of $\mathbf{y}$ as long as the semantic hierarchy based discriminative model has the abilities in the following:

- Given a trained model, $\forall \mathbf{y} \in \mathcal{Y}$, the model gives a $p(\mathbf{y}|\mathbf{x})$ indicating the likelihood for $\mathbf{y}$ becoming the best describing structure for $\mathbf{x}$
- Given a trained model and an input $\mathbf{x}$, the model can infer the best $\mathbf{y}$ to describe $\mathbf{x}$, such that $p(\mathbf{y}|\mathbf{x})$ is the largest possible value.

5.5.4 The Similarity Function Definition

Suppose a parameter vector $\Lambda = \{\lambda_k\} \in \mathfrak{R}^K$ is obtained after training with a best overall log-likelihood. A most likely configuration $\mathbf{y}$ can be inferred from the parameter

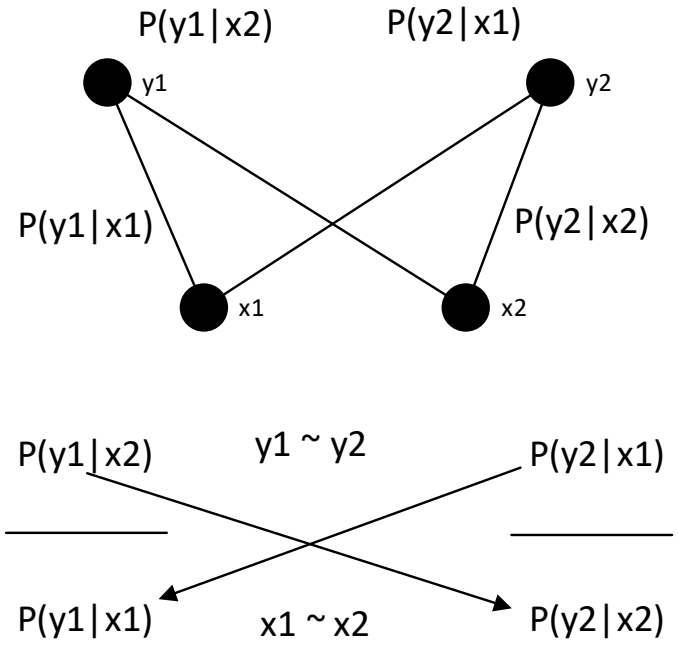

FIGURE 5.3: The cross-representation for describing a pair of input observations by utilising a pair of most-likely output configurations in a TRF.

vector Λ and the feature function set. Thus, for observation $\mathbf{x}_1$ and $\mathbf{x}_2$, we can get two pairs: $(\mathbf{x}_1, \mathbf{y}_1)$ and $(\mathbf{x}_2, \mathbf{y}_2)$.

Given the discuss above, we can have a measurement for this semantic hierarchy based pairwise similarity in a symmetric way.

$$\frac{p(\mathbf{y}_1|\mathbf{x}_2)}{p(\mathbf{y}_1|\mathbf{x}_1)} = s(\mathbf{x}_1, \mathbf{x}_2|\mathbf{y}_1) \tag{5.7}$$

Eq: 5.7 reflects the similarity between $\mathbf{x}_1$ and $\mathbf{x}_2$ in the view of $\mathbf{y}_1$, and similarly, we have:

$$\frac{p(\mathbf{y}_2|\mathbf{x}_1)}{p(\mathbf{y}_2|\mathbf{x_2})} = s(\mathbf{x}_1, \mathbf{x}_2|\mathbf{y}_2) \tag{5.8}$$

Eq: 5.8 reflects the similarity between $\mathbf{x}_1$ and $\mathbf{x}_2$ in the view of $\mathbf{y}_2$.

Because we can have a symmetric representation to measure the aggregated similarity between $\mathbf{x}_1$ and $\mathbf{x}_2$ in both views of $\mathbf{y}_1$ and $\mathbf{y}_2$ at the same time, which can be written as:

$$\frac{p(\mathbf{y}_1|\mathbf{x}_2)}{p(\mathbf{y}_1|\mathbf{x_1})} \times \frac{p(\mathbf{y}_2|\mathbf{x}_1)}{p(\mathbf{y}_2|\mathbf{x}_2)} = s(\mathbf{x}_1, \mathbf{x}_2|\mathbf{y}_1, \mathbf{y}_2) \tag{5.9}$$

It is important to note that, we also have semantic interpretations for the cross terms, as depicted in Fig 5.3: the $\frac{p(\mathbf{y}_1|\mathbf{x}_2)}{p(\mathbf{y}_2|\mathbf{x_2})}$ and $\frac{p(\mathbf{y}_2|\mathbf{x}_1)}{p(\mathbf{y}_1|\mathbf{x}_1)}$ .For example, in the case of $\frac{p(\mathbf{y}_1|\mathbf{x}_2)}{p(\mathbf{y}_2|\mathbf{x_2})}$, it represents the percentage of description power $\mathbf{y}_1$ has as an alternative for $\mathbf{y}_2$ in best describing $\mathbf{x}_2$. Because the $(\mathbf{x}_2, \mathbf{y}_2)$ is known to be most-likely pair, $p(\mathbf{y}_1|\mathbf{x}_2)$ can never exceed $p(\mathbf{y}_2|\mathbf{x_2})$. Thus, we have

$$0 < \frac{p(\mathbf{y}_1|\mathbf{x}_2)}{p(\mathbf{y}_2|\mathbf{x_2})} \leq 1 \tag{5.10}$$

Similarly, we can also have

$$0 < \frac{p(\mathbf{y}_2|\mathbf{x}_1)}{p(\mathbf{y}_1|\mathbf{x_1})} \leq 1 \tag{5.11}$$

Thus, we have a nice property in combining these two terms (Eq: 5.10 and Eq: 5.11) into the cross product-based semantic representation in Eq: 5.9, which we will demonstrate by giving a formal definition for the similarity measurement as follows:

Definition 5.2. *The Similarity Measurement for Heterogeneous Input*

Let $\mathbf{x}_a$ *and* $\mathbf{x}_b$ *be two observations for a TRF,* $\Lambda = \{\lambda_k\} \in \mathfrak{R}^K$ *be a parameter vector learnt from training process,* $\{f_k(\mathbf{y}_t, \mathbf{x}_t)\}_{k=1}^K$ *be a set of real-valued feature functions, and* $(\mathbf{x}_a, \mathbf{y}_a)$ *and* $(\mathbf{x}_b, \mathbf{y}_b)$ *be two most likely configuration pairs. Then* the similarity between $\mathbf{x}_a$ and $\mathbf{x}_b$ *can be defined as:*

$$S(\mathbf{x}_a, \mathbf{x}_b) \stackrel{\text{def}}{=} \sqrt{\frac{p(\mathbf{y}_a|\mathbf{x}_b) \cdot p(\mathbf{y}_b|\mathbf{x}_a)}{p(\mathbf{y}_a|\mathbf{x}_a) \cdot p(\mathbf{y}_b|\mathbf{x}_b)}} \in (0, 1] \tag{5.12}$$

Clearly, the higher value of $S(\mathbf{x}_a, \mathbf{x}_b)$ is, the more similar $\mathbf{x}_a$ and $\mathbf{x}_b$ are to each other.

Theorem 5.1. *The similarity measurement* $S(\mathbf{x}_a, \mathbf{x}_b)$ *defined in a TRF is a Mercer kernel.*

Proof. Given that G is a TRF, any probability mass is non-zero and normalized. $S(\mathbf{x}_a, \mathbf{x}_b)$ is strictly positive definite according to the property of cross-product terms. Thus, the $S(\mathbf{x}_a, \mathbf{x}_b)$ for TRF is a Mercer kernel. □

In summary, this similarity-based approach makes use of the topology fixed by a criminative model defined by a class of general probabilistic models. Although a fully modelling of the overall probability distribution requires calculating the observation $\mathbf{x}$-dependent partition function $Z(\mathbf{x})$, the pairwise relation can be calculated directly from a weight vector $\boldsymbol{\theta}$ for any pair of $(\mathbf{x}_i, \mathbf{x}_j)$. The modelling of this novel similarity measurement relies on the cross relation between different combinations of

$p(\mathbf{y}_a|\mathbf{x}_b), p(\mathbf{y}_b|\mathbf{x}_a), p(\mathbf{y}_a|\mathbf{x}_a), p(\mathbf{y}_b|\mathbf{x}_b)$, as discussed. This results in a neat form and a great computational benefit which we will introduce next.

5.5.5 Efficient Evaluation

According to the definition above, the similarity between two observations is

$$S(\mathbf{x}_a, \mathbf{x}_b) \stackrel{\text{def}}{=} \sqrt{\frac{p(\mathbf{y}_a|\mathbf{x}_b) \cdot p(\mathbf{y}_b|\mathbf{x}_a)}{p(\mathbf{y}_a|\mathbf{x}_a) \cdot p(\mathbf{y}_b|\mathbf{x}_b)}}$$

Recall the form of probability distribution defined by a TRF or a CRF:

$$p(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{c \in \mathcal{C}} \psi_c(\mathbf{y}_c|\mathbf{x}, \boldsymbol{\theta}_c)$$

Each conditional probability can be written in the log-linear form. However, as we mentioned above, computing the instance-specific normalisation function $Z(\boldsymbol{\theta})$ is computational intractable, which is one of the major factors to the difficulty of training and inference for MRF/CRF model family.

If we calculate $p(\mathbf{y}_a|\mathbf{x}_b), p(\mathbf{y}_b|\mathbf{x}_a), p(\mathbf{y}_a|\mathbf{x}_a), p(\mathbf{y}_b|\mathbf{x}_b)$ separately to get the similarity measurement, we'd missed the nice view.

Fortunately, if we expand Eq: 5.12 with the distribution, we can have:

$$S(\mathbf{x}_a, \mathbf{x}_b) = \sqrt{\frac{\frac{\exp[\text{score}(\mathbf{y}_a|\mathbf{x}_b)]}{Z(\mathbf{x}_b)} \cdot \frac{\exp[\text{score}(\mathbf{y}_b|\mathbf{x}_a)]}{Z(\mathbf{x}_a)}}{\frac{\exp[\text{score}(\mathbf{y}_a|\mathbf{x}_a)]}{Z(\mathbf{x}_a)} \cdot \frac{\exp[\text{score}(\mathbf{y}_b|\mathbf{x}_b)]}{Z(\mathbf{x}_b)}}}$$

where the $\text{score}(\mathbf{y}_a|\mathbf{x}_b)$, $\text{score}(\mathbf{y}_b|\mathbf{x}_a)$, $\text{score}(\mathbf{y}_a|\mathbf{x}_a)$ and $\text{score}(\mathbf{y}_b|\mathbf{x}_b)$ are basically linear function of $\boldsymbol{\theta}$ and the feature function set $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$. Then the partition functions are eliminated and the $S(\mathbf{x}_a, \mathbf{x}_b)$ can directly calculated by without any calculation for the partition function $Z(\mathbf{x}_a)$ and $Z(\mathbf{x}_b)$. The neat form we have for $S(\mathbf{x}_a, \mathbf{x}_b)$ now is:

$$S(\mathbf{x}_a, \mathbf{x}_b) = \sqrt{\exp[\text{score}(\mathbf{y}_a|\mathbf{x}_b) + \text{score}(\mathbf{y}_b|\mathbf{x}_a) - \text{score}(\mathbf{y}_a|\mathbf{x}_a) - \text{score}(\mathbf{y}_b|\mathbf{x}_b)]} \quad (5.13)$$

Where the score is the dot product between the trained weight vector and the feature functions. Thus, this similarity function can be efficiently evaluated without the calculation of partition function $Z(\mathbf{x})$.

5.5.5.1 Evaluation Complexity

Theorem 5.2. *The time complexity for evaluating any* $S(\mathbf{x}_a, \mathbf{x}_b)$ *is* $O(1)$.

Proof. According to Eq: 5.13, there are only one factor $\boldsymbol{\theta}$ that actually influences the overall time complexity. Given that $|\boldsymbol{\theta}|$ is constant, which thus makes $|\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})|$ also a constant, the overall time complexity in calculating $S(\mathbf{x}_a, \mathbf{x}_b)$ is in $O(1)$. □

5.5.5.2 Evaluation Algorithm

We propose the algorithm for evaluating $S(\mathbf{x}_a, \mathbf{x}_b)$ as follows:

input : Two pairs of most likely pairs $\mathbf{x}_a$, $\mathbf{y}_a$, $\mathbf{x}_b$, $\mathbf{y}_b$
output : The similarity measurement $S(\mathbf{x}_a, \mathbf{x}_b)$
parameter: Feature function set F, $\boldsymbol{\theta}$

Feature function set F is structural binding and $\boldsymbol{\theta}$ is trained successfully;
for $i \leftarrow 1$ **to** $|F|$ **do**
  $score \leftarrow score + \theta_i \times (f_i(\mathbf{y}_a, \mathbf{x}_b) + f_i(\mathbf{y}_b, \mathbf{x}_a) - f_i(\mathbf{y}_a, \mathbf{x}_a) - f_i(\mathbf{y}_b, \mathbf{x}_b))$;
$score \leftarrow \sqrt{\exp(score)}$;

Algorithm 1: The similarity measure evaluation for heterogeneous input data.

5.6 TRF Training

We demonstrate the methodology for training TRF based on pairwise distance provided by the novel similarity measure function.

5.6.1 Motivation for Similarity Based Training

Consider the classic training setting for a structured prediction problem, given a data set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, the training labels $\mathbf{y}_i \mid_{i\in[1,N]}$ provides a target distribution that the training model is about to fit in. In classic training setting, the information embedded in the $\mathbf{y}_i \mid_{i\in[1,N]}$ is usually processed independently. That is, the $(\mathbf{x}_i, \mathbf{y}_i)$ pairs are assumed to be i.i.d. and $\mathbf{x}_i$ and $\mathbf{y}_i$ are put into the training model directly without further exploiting the embedded information.

In the case of plain label set in the simple multi-label problem setting, although observations $\mathbf{x}_i$ with the same labels can be deemed as similar to each other, it is difficult to

quantify the difference and bring it into the probabilistic model. Meanwhile, it is fairly likely that the person who chose the label did the selection simply because there is no better option. Thus, in this scenario, modelling the similarity between input data is extremely difficult. The possible reasons are as follows:

- Plain label set structure generally means less latent information embedded. For a categorical small label set it usually difficult to measure the similarity solely based on the tagging.
- The main cause for label choosing in the case of plain label set is that, often the chosen label is given simply because the person didn't have other better choice. This falls into the category of MLE. However, such information is not rich enough to model the similarity instead of pure ranking.

In the case of complex structured prediction, however, the training set has much more information embedded, particularly for ontology-based output semantic hierarchies. This is because:

- The ontology-based structures are normally easier to define similarity/distance by its semantic nature. These structures usually have directed hierarchical information embedded inside already. The reason behind is straightforward - due to the increased descriptive power from complex output structures, the ontology-assisted learning has the ability of being describing the heterogeneous observation in a much more fine-grained accuracy.
- The main cause for one configuration $\mathbf{y}_i$ appearing in the training set usually is not the lack of choices. Given that the form of distributions could be various, the person who made the configuration for training usually has other choices. Thus, the $\mathbf{y}_i$ here is a MLE and at the same time they form a topological space where relative similarity and distance among the whole training set could be induced.

In summary, a further exploitation to the latent information in the training set is promising for structured predicting model training, particularly when considering the general computing difficulties from traditional method in training the MRF/CRF-based models. Given the efficiency provided by the proposed similarity measurement for general semantic hierarchy-based output, it is interesting to examine the similarity-based training techniques.

5.6.2 Similarity-based Training for TRF

Given the discussion above, we have an improved description for the training setting of TRF, where a fixed ontology-based semantic hierarchy is used as the descriptive output structure. Given a data set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ and the assumption that these day obeys i.i.d., the training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ implies a topological space where the similarity information is embedded, such that observations which are similar to each other according to the ontology-based domain knowledge would have similar confidence value distributions over the semantic hierarchy.

Theorem 5.3. *The Equivalence between the MLE and the Similarity-based Methods for TRF $\boldsymbol{\theta}$ is the MLE of an exact-inference trained TRF if and only if $\boldsymbol{\theta}$ is also the MLE of the similarity-based method trained TRF.*

Proof. The uniqueness of MLE is guaranteed by the convexity of the log-likelihood function of a TRF. Hence, if a MLE of one method with gradients equal to 0, it is the MLE, then its uniqueness is guaranteed. thus it is also the MLE of the other training method, and vice versa. □

5.6.3 The Training Algorithm for TRF

By revealing the dual form of the training process from the perspective of negative log likelihood and pairwise distances we obtain Algorithm 2 based on the proposed similarity measure. With a discriminatively-trained graphical model, where a unified ontology-based semantic hierarchy is adopted as a fixed output structure, we further demonstrate in Chapter 6 the capability of the proposed similarity measure in simplifying geometric analysis of the projected space.

5.7 Inference for TRF

Given the difficulties in inferencing for traditional MRF/CRF models, we propose the novel methodology for directly inferencing with semantic structural features by giving the proofs directly as follows.

Theorem 5.4. *Given a TRF with continuous output confidence value distribution, a structural binding feature function set F and a fully trained weight vector $\boldsymbol{\theta}$, there is always a range of distributions $\{\mathbf{y}^*\}$ that is $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$.*

input : A training set $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, The similarity measurement $S(\mathbf{x}_a, \mathbf{x}_b)$
output : $\boldsymbol{\theta}$
parameter: Feature function set F

Feature function set F is structural binding and $\boldsymbol{\theta}$ is initialized with 0.5f;
for *step* $\leftarrow 1$ **to** *LBFGS Converge* **do**
  for $i \leftarrow 1$ **to** N **do**
    for $j \leftarrow 1$ **to** $|F|$ **do**
      $score \leftarrow score + \theta_j \times (f_j(\mathbf{y}_i, \mathbf{x}_i))$;
    $score \leftarrow score - 2 \times \log S(\mathbf{x}_i, \mathbf{x}'_i)$;
    for $j \leftarrow 1$ **to** $|F|$ **do**
      $gradient_j \leftarrow gradient_j + 2 \times score \times (f_j(\mathbf{y}_i, \mathbf{x}_i))$;
  LBFGS Optimizing;

Algorithm 2: The similarity-based TRF training.

Proof. The feature function set F could be treated as single function as $F(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})$. Because F is a structural binding feature function set, F is not a full indicator function. Thus, there are always more than one points achieve maximal $p(\mathbf{y}|\mathbf{x})\ |_{\boldsymbol{\theta}}$. □

Theorem 5.5. *Given a TRF with continuous output confidence value distribution, a structural binding feature function set F and a fully trained weight vector $\boldsymbol{\theta}$, one can always induce another $\mathbf{y}^*$ from an origin one $\mathbf{y}$, such that $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$.*

Proof. The same argument as 5.4 above applies here. □

Theorem 5.6. *Given a TRF with continuous output confidence value distribution, an ontology O-based hierarchy H, a structural binding feature function set F with H-edge-indexed structural features and a fully trained weight vector $\boldsymbol{\theta}$, one can induce the $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$ directly.*

Proof. According to 5.4, with a fixed $\boldsymbol{\theta}_H$, one can always induce a group $F_{\boldsymbol{\theta}_H}$, such that the corresponding $\mathbf{y}_{F_{\boldsymbol{\theta}_H}}$ is obtainable. Then according to 5.5, the optimal $\mathbf{y}^*$ is obtainable, s.t., $\mathbf{y}^* = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$. □

5.8 Conclusions

Similarity measures used in supervised machine learning methods typically assume a fixed-width vector representation for data and a scalar class label, but these cannot be used for structured prediction tasks with heterogeneous input data. Following the contributions to modelling the TRF related structured prediction problem, in this

chapter, we developed and proposed novel algorithms to tackle the computing problem over the entire CRF/MRF model family. To better model and understand semantic states underlying heterogeneous data we present a novel similarity measure in the context of discriminatively-trained graphical models, where a unified ontology-based semantic hierarchy is adopted as a fixed output structure. This similarity measure greatly simplifies geometric analysis of the projected space defined by discriminative graphical models. The efficiency of evaluating this similarity measure enables learning discriminatively-trained probabilistic graphical models without many of the standard structural limitations, e.g., linear chains, two dimensional grids, etc. Thus, the proposed novel similarity-based learning and prediction framework can be applied to a much wider range of problems, e.g., recommendation systems, information extraction systems, etc. In the previous chapters including this one, we did semantic problem modelling, theoretical modelling and computing problem modelling. Now we move to the empirical study in the next chapter to provide an insight to the mechanism of the algorithms proposed when applying to solve real-world problems.

6 Implementation and Experiments with TRF over Heterogeneous EHR Data

6.1 Introduction

Based on the theoretical techniques developed above, we conduct empirical studies of the proposed ontology-assisted structured status prediction problem over heterogeneous EHR data. In this chapter, we first introduce the EHR data and then we report our experimental results and observations in detail. We conclude the empirical study in the end.

6.2 The EHR Data Feature

In the empirical study, the major playground for this proposed learning framework is a real-world data set which was generated and collected in several celebrated hospitals in Sydney (see Table 6.1). Different types of health-related data records, including text-based etiology and manifestation descriptions, numerical and categorical test results from different medical departments with various forms together with strings of categorical ICD codes as diagnosis results, etc. were gathered with specific time stamps and patient information. As depicted in Figure 6.1, the heterogeneous EHR data set has 138,737 admission records by 57,941 patients with a huge number of pathological test results of various kinds and in-hospital movement records. According to Figure 6.2, the one-time-admission patients take up to 70% of the cohort. The patient number decreases rapidly as the admission number gets larger.

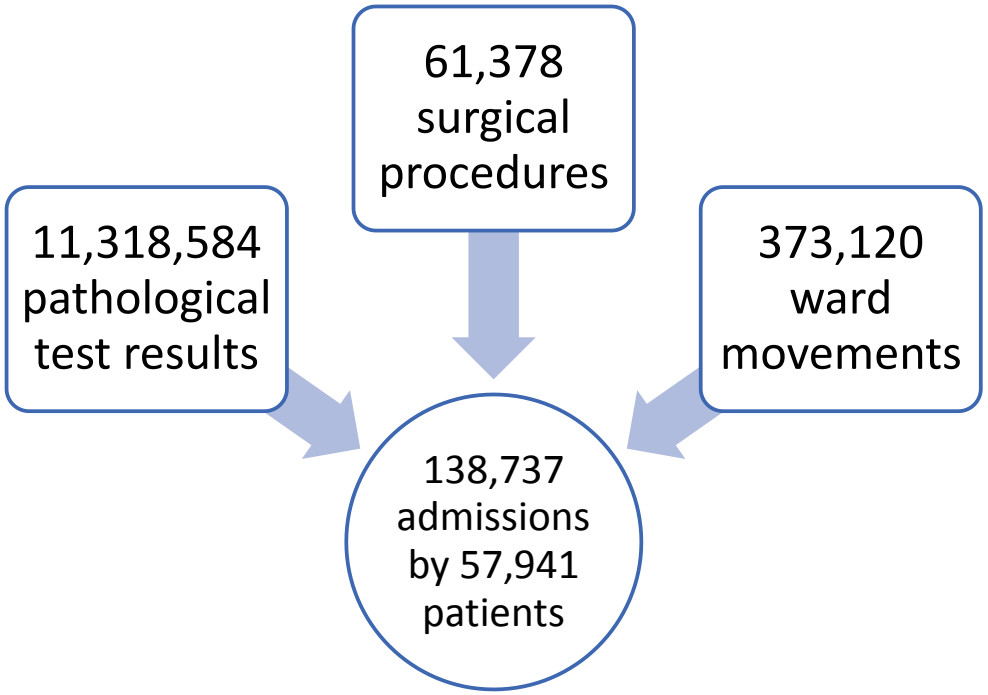

FIGURE 6.1: The EHR data overview

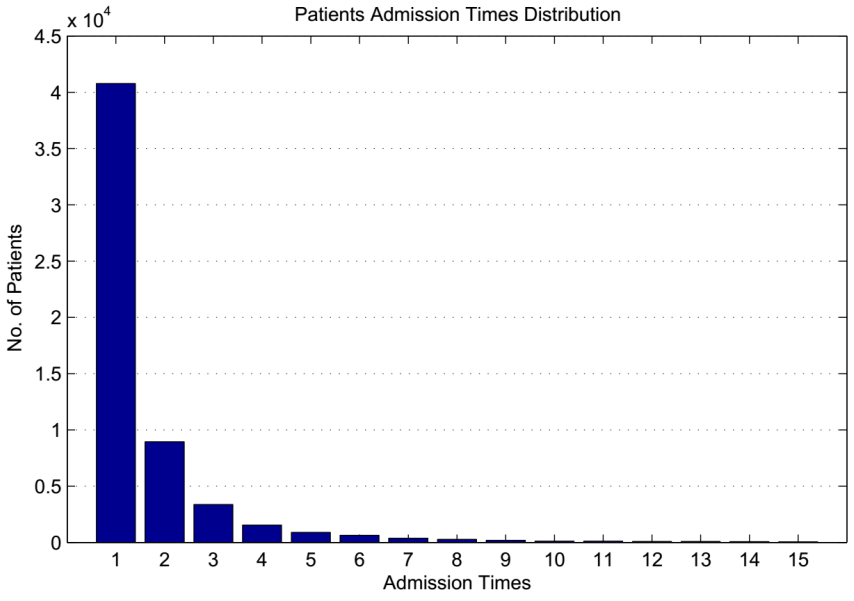

FIGURE 6.2: The Histogram of Patients Admission Times

	Definition
[0]	number of admissions before the current one
[1]	is female
[2]	is male
[3]	cumulative length of stay (LOS) previous year
[4]	time from the beginning of first admission to the current one
[5]	time from the end of last admission to the start of current one
[6]	length of the last admission
[7]	number of pathological tests in the last admission
[8]	number of theatre movements in the last admission
[9]	number of ward movements in the last admission
[10]	current age
[11]	length of the current admission
[12]	number of pathological tests in the current admission period
[13]	number of tests with abnormal results
[14]	percentage of tests with abnormal results
[15]	percentage of tests with normal results
[16-29]	distribution of tests among chosen departments, with a fixed length list : 14 dept_name
[30-43]	distribution of tests with abnormal results among chosen departments, with a fixed length list:14
[44-243]	distribution of tests among different panel name, with a fixed length list : 200 panelname
[244-443]	distribution of tests with abnormal results among different panel names, with a fixed length list : 200 panelname: based on 20000 patients
[444]	max interval of all tests in this admission
[445]	max interval of tests with abnormal results in this admission
[446]	number of adjacent tests(not the same day) with department changes
[447]	number of adjacent test(not the same day) with panel name changes
[448]	number of adjacent test(not the same day) with department changes with abnormal results
[449]	number of adjacent test(not the same day) with panel name changes with abnormal results
[450-599]	mostly concerned test code count list, appeared tag list: 150
[600-749]	mostly concerned test code list, w/ appeared & abnormal tag list: 150
[750]	number of theatre movements in the current admission period
[751]	time from the start of the current admission period to the start of the last theatre movement
[752]	no surgery period time
[753]	accumulative length of surgery for this admission
[754]	max length of surgery in this admission
[755]	min length of surgery in this admission
[756]	average length of surgery in this admission
[757]	max interval of surgery in this admission
[758]	min interval of surgery in this admission
[759]	average interval of surgery in this admission
[760]	number of ward movements in the current admission period
[761-786]	mostly concerned ward tag list: 26
[787]	max length of ward stay in this admission
[788]	min length of ward stay in this admission
[789]	average length of ward stay in this admission
[790]	time to the first ward change from the start of current admission
[791]	time from the last ward change to the end of the current admission

TABLE 6.1: Complete Feature list for **X**

6.2.1 The Heterogeneous EHR Data Features

The EHR data presents several unique features which make traditional prediction model inapplicable to the heterogeneous structures. Potential limitations imposed by these features are discussed as follows.

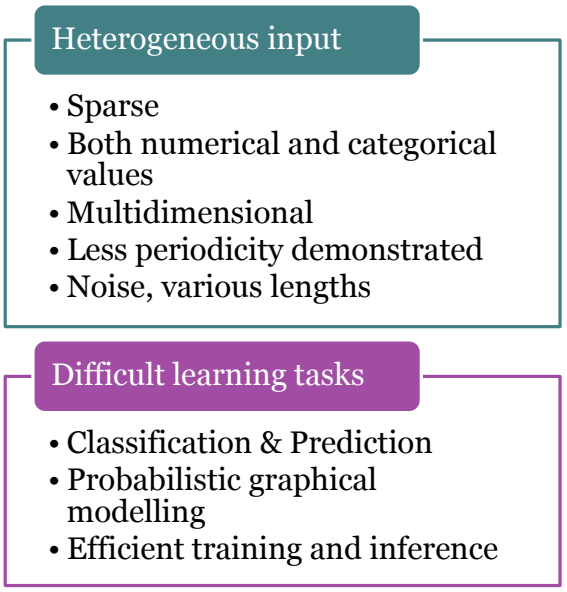

FIGURE 6.3: The heterogeneous EHR data challenge

6.2.2 Different Perspectives of the Input Data

The logical hierarchy for P_k and $\mathbf{A}^k$ described implies different possible ways of event grouping. For every event e_i with a time stamp falls in a corresponding time interval $\mathbf{t}_l^k$ ($\mathbf{t}_l^k \in \mathbf{a}_l^k, \mathbf{a}_l^k \in \mathbf{A}^k, 1 \leqslant l \leqslant |\mathbf{A}^k|$), we have $e_i \in \mathbf{r}_l^k \cup \mathbf{s}_l^k \cup \mathbf{w}_l^k, 1 \leqslant i \leqslant |\mathbf{r}_l^k| + |\mathbf{s}_l^k| + |\mathbf{w}_l^k|$. The whole set $\{e_i\}$ can be grouped according to the nature of the individual event ($e_i \in \mathbf{r}_l^k \cup \mathbf{s}_l^k \cup \mathbf{w}_l^k$), the different levels of logical abstractions (e.g. the level of admissions $\{\mathbf{a}_l^k\}$ or the level of patients $\{P_k\}$) or even to the time stamps associated with events $\{e_i\}$.

These various combinations together with the total order guaranteed by the time stamps of all the events $\{e_i\}$ give different possible logical perspectives of the input EHR data. The choice of the suitable logical view is determined by the desired content and the preferred structure of the prediction target.

6.2.2.1 The Multi-Source Generative Data View

A natural way of alleviating the difficulties brought by the heterogeneity of the EHR data is to divide the overall input into groups with similar structures or patterns. Unfortunately, the name of 'heterogeneous input' itself implies that it is difficult to define or even find any abstract structure or pattern other than the form of representation of the observation data itself. In the health-related scenarios, medical equipments of similar types are normally generating similar forms of records. Thus the type of the data source is almost the most straightforward and most decisive single factor to the form of the observation. Consequently a generative data view is formed by dividing the whole heterogeneous input according to the types of the data-generating sources.

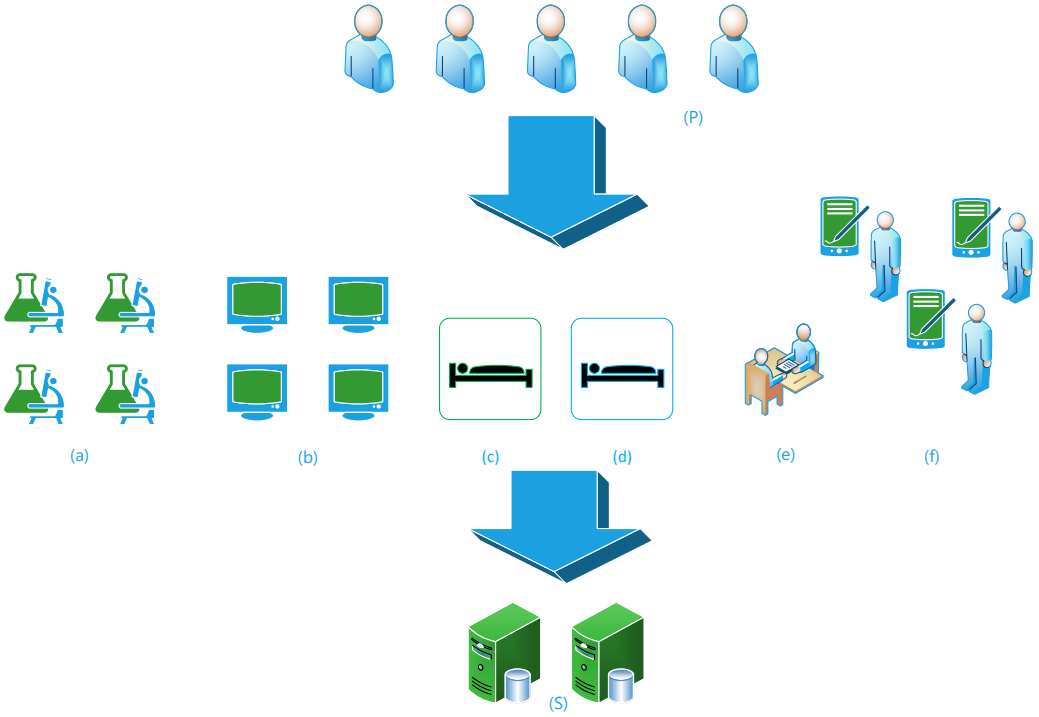

FIGURE 6.4: The EHR data's multi-source view.

The components of the multi-source data view are depicted in Figure 6.4. The EHR data of patients (P) are generated from different sources (e.g. from part (a) to part (f)). Part (a) represents the pathological tests from different departments. The result data could be both numerical and categorical, or even pure descriptive text; Part (b) represents the electronic medical equipment that can check and record the patients' physiological variables automatically; Part (c) represents the ward-related movements;

Part (d) represents the surgical procedures; Part (e) represents the diagnosis information given by doctors in the forms of ICD-10 codes; Part (f) represents the patients' physiological variables checked or recorded by human (e.g. nurses and doctors), where additional noise is inevitably brought in. All the heterogeneous EHR data, no matter its type, volume or the location of source, is collected and stored by central servers (S).

6.2.2.2 The Timestamp-Based Totally Ordered Event Sequence View

The data collected by the central servers in Figure 6.4 can be further merged and sorted according to the associated time stamps to form a sequence of events in the ascending order, as depicted in Figure 6.5.

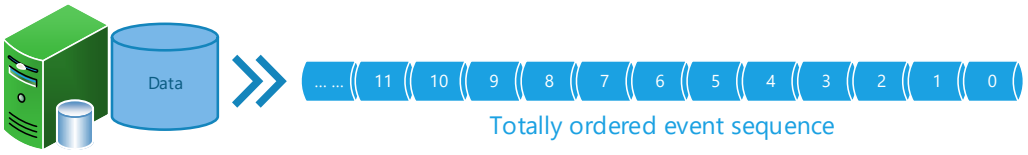

FIGURE 6.5: The EHR data's totally ordered event sequence view.

Such totally ordered sequence $\{e_i\}, i \geqslant 0$ is an event-level representation which simulates the typical type of data input for an online system. Each event e_i contains elementary signals with small pieces of information, however, the content data of e_i could also have heterogeneous substructures as well as various forms.

6.2.2.3 The Patient's Admission Trajectory View

Although the totally ordered event sequence preserves strong inter-event temporal relations, it is difficult to model medical patterns and to associate with the domain knowledge. Thus a patient and admission based logical view with a more natural presentation for patients' health cycles is required to further model the complex dependencies and the underlying medical rules. Recall the data components discussed, each patient $P_k = \{\mathbf{v}^k, \mathbf{A}^k\} \in \mathcal{P}$ can be treated as a combination of an invariable feature vector $\mathbf{v}^k$ and a sequence of admissions $\mathbf{A}^k = \{\mathbf{a}_1^k, \mathbf{a}_2^k, \ldots, \mathbf{a}_n^k\}$. As depicted in Figure 6.6, each admission period $\mathbf{a}_l^k \in \mathbf{A}^k, 1 \leqslant l \leqslant |\mathbf{A}^k|$ of the patient P_k is actually a container for all the heterogeneous information $\mathbf{a}_l^k = \{\text{id}, \mathbf{t}_l^k, \mathbf{r}_l^k, \mathbf{s}_l^k, \mathbf{w}_l^k\}$, which gives an overall description for this admission period. The list of all the previously seen patients is maintained to track the corresponding admission trajectories.

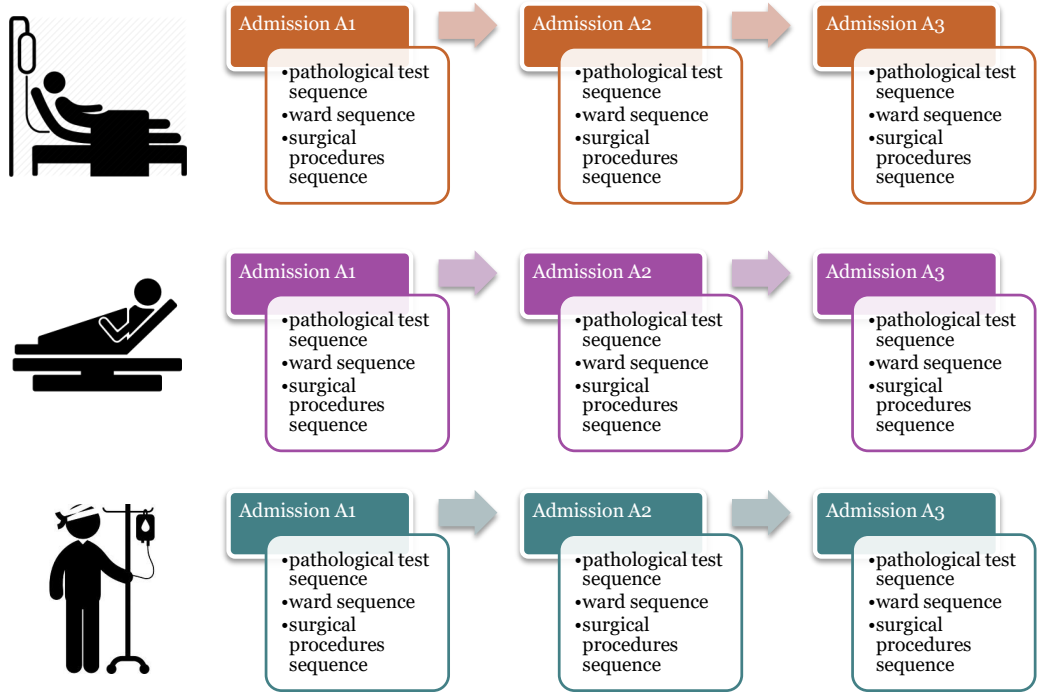

FIGURE 6.6: The EHR data's admission trajectory view

The non-admission periods can be treated as relatively stable time intervals between admissions, hence no signal is assumed during those periods of time. The admission periods, including the short time from the patient's being aware of new symptoms to the point of arriving at the hospital, usually represents relatively rapid physiological change, thus the intense signals embedded and the inter-dependencies between them are information-rich and of great significance in the potential learning and prediction tasks.

6.2.2.4 The Combination of Different Logical Views for Extracting Features of Observations

One direct difficulty in modelling the heterogeneous input data lies in extracting features for the observation. Different logical views contribute separately in extracting features at different levels. The combination of features from different views provides a mechanism for completely extracting features for the heterogeneous input. This results in a large set of highly inter-related features for the observation. Thus, the learning and prediction model is required to have the ability of handling these challenges, as will be demonstrated in Figure 6.7.

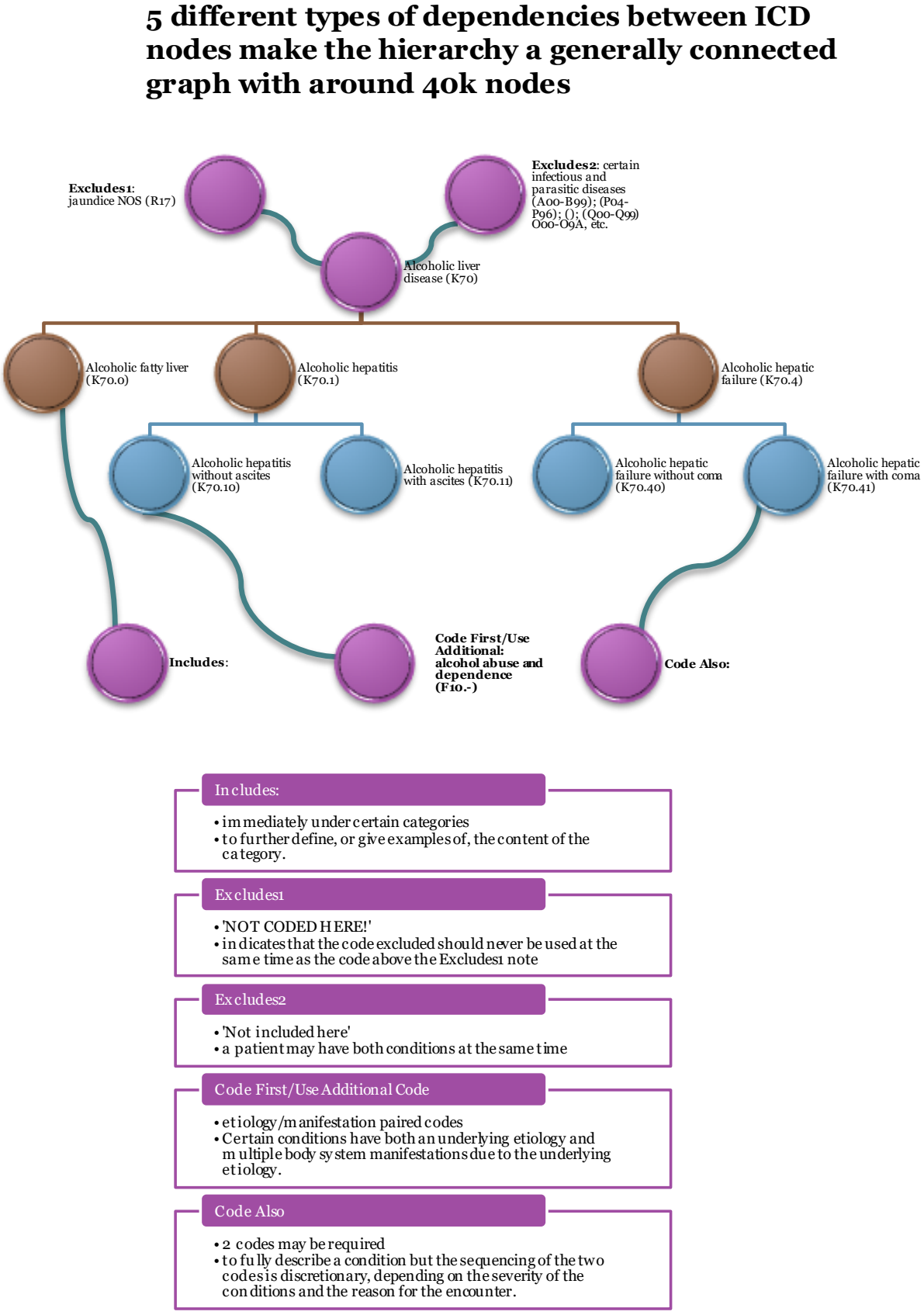

FIGURE 6.7: Dependencies in the ICD-10 hierarchy.

6.3 Experiment Setting

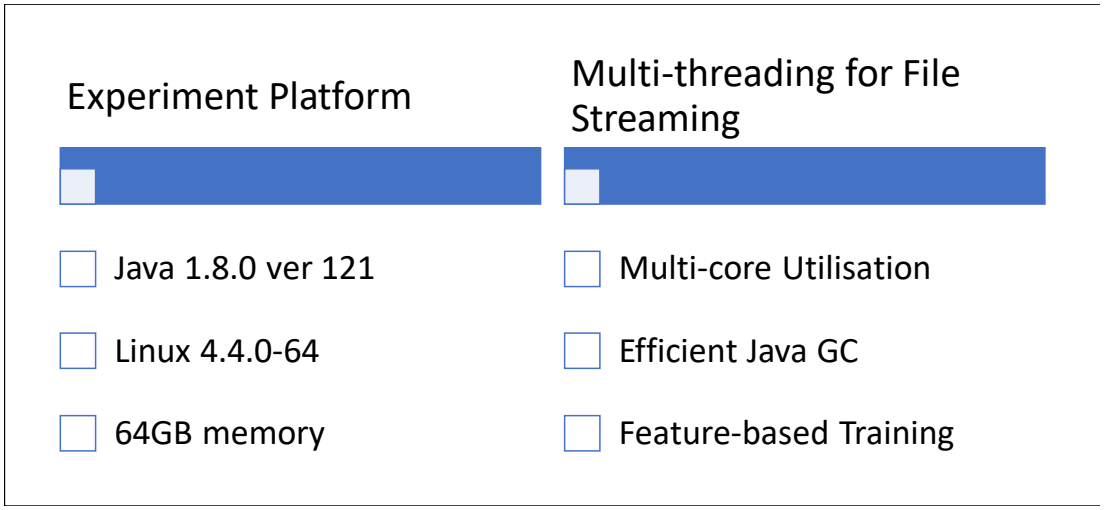

FIGURE 6.8: Experiment setting for empirical study.

6.3.1 The Prediction Framework

The computing resources used were as listed in Fig 6.8 and the major functions of this framework are as follows:

- Build a probabilistic graphic model on the large EHR type data set to represent the knowledge and dependencies among a huge number of factors, so that
- This model can be trained with good efficiency
- Given the heterogenous EHR type data from some patient, a distribution of all the possible health problems (ICD code) can be predicted as an accurate and complete description of the health status of such patient at the corresponding time when the data set was generated.
- Given the heterogenous EHR type data from some patient, a list of admissions from the database with the most similar health trends and status can be generated, without compromising any patient's privacy.

The following are the steps to build the TRF model for EHR type data with confidence distribution for the ICD codes as input and output Y side values.

- Setup the feature functions for the TRF model's X side

- Adapt to a basic Gibbs random field (MRF with a strictly positive probability density function) from the ICD hierarchical structure
- Train the model with different methods for different clique sizes in TRF (different levels of connectivity)
- Inference problem reduces to a numerical optimization problem, given the weight vector and feature functions.
- Find the most similar admissions based on the similarity measures.

6.3.1.1 Similarity Based Training for Different Admission Pair Combinations

Given that the basic methodology for learning the TRF model in this problem is based on pairwise similarities. Thus, an important part of empirical study is to combine admissions from different patients or different positions in the same temporal trajectory. It is worth noting that, once the combinations having been fixed, it is not necessary to rebuild such relation each time.

6.3.2 The ICD-10 Ontology-Based Semantic Hierarchy

The running example below is on the subgroup of the disease of liver with ICD code range K70-K77, particularly K70 - Alcoholic liver disease (page 522 of the ICD-10-CM tabular).

6.3.3 Feature Function Construction

As discussed, the equivalence criteria must be met in order to truthfully reflect the structural information defined in a graph. In TRF, a feature function is generally conditioned on cliques from both the input and output side at the same time. Given the high connectivity between every pair of maximal cliques according to the TRF definition, we first build single-clique-indexed feature functions and then combine them to form the final structural binding set. We examine the details below.

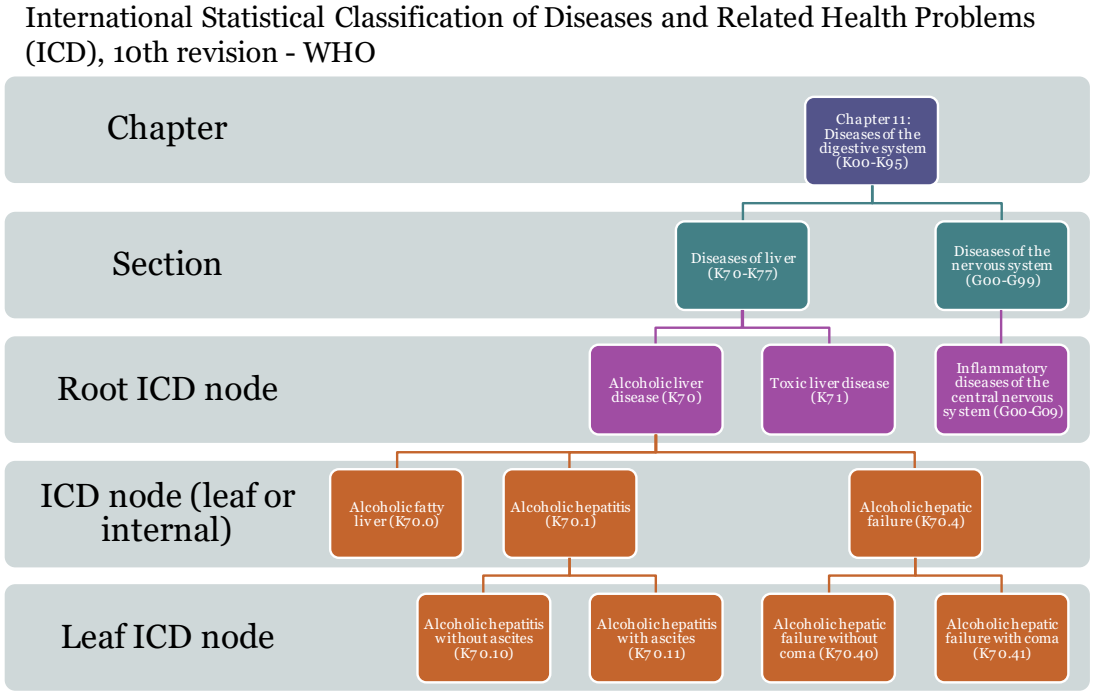

FIGURE 6.9: The ICD-10 hierarchy.

6.3.3.1 Feature functions for the observations

1. A feature vector (length: 792) is constructed for every observation $\mathbf{x}$ to try to capture all the relevant statistical and temporal information of the data.

2. Then all the feature functions can be built up from combining feature functions from both the observation and the output ICD code separately

6.3.3.2 Clique Decomposition: Feature Functions for the Output Structure

For the cliques with max size of 3, it is still possible to use the traditional max-likelihood method to train this model. It also sets up the performance and accuracy baseline for the TRF model. As the clique size becomes larger, the similarity based training method involves to make a more efficient training.

6.3.4 Implementation Techniques

- Data streaming: as discussed in similarity-based training part, the input to the training algorithm is a stream of admission pairs. Given that the number of

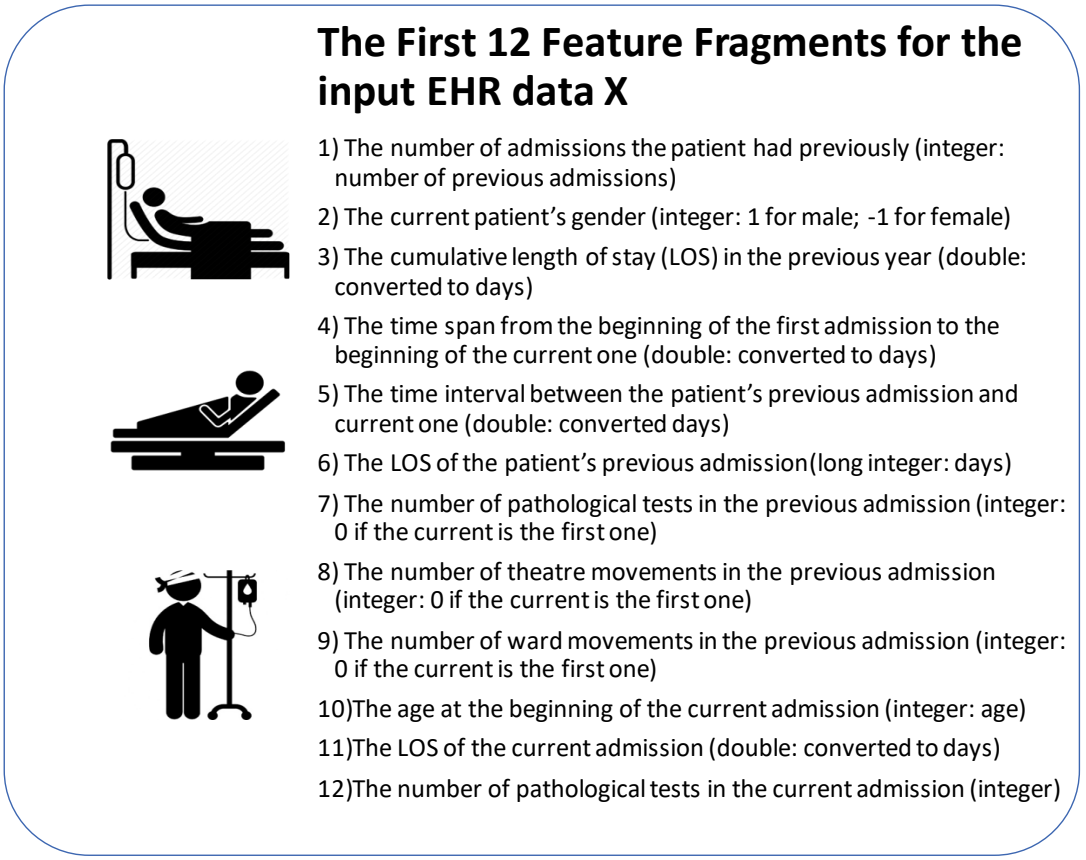

TABLE 6.2: Feature fragment list for the EHR input $\mathbf{X}$.

possible pairwise combinations is n^2, streaming in such high volume is challenging to the implementation techniques.

- Parallel Training: a question right after setting up the data streaming mechanism is on the potential capability of concurrently handling training data. Despite that the training algorithm cannot be fully paralleled due to the central numerical optimizer, handling the individual pair of admissions before entering the LBFGS optimization engine is feasible. It is worth noting that writing back to the weight vector or gradient vector needs synchronisation. Furthermore, because these vectors are generally very long, synchronization with a lock on the vector is very likely to decrease the system throughput, particularly when the concurrency is high.

6.4 Result Evaluation Techniques

In this section, we examine several evaluation techniques as the measurement for evaluating the outcome of the TRF model. These variables actually play an important role in the empirical study.

6.4.1 Distance Measurement for Confidence Value Distribution over Ontology-Based Semantic Hierarchies

Recall the training process of a TRF, we need to measure a numerical distribution over a semantic hierarchy to let the similarity function to simulate the distribution. Thus, we look into the measurement for directly infer the structural difference between two semantic hierarchy-based distributions.

6.4.1.1 The Distance between Two Nodes in a Semantic Hierarchy

The distance between two ICD nodes is formally defined as the average number of edges in the overall hierarchy to the nearest common ancestor node. According to WHO, every ICD node has been assigned a hierarchy-compliant code. Thus instead of traversing inside the ICD hierarchy, the distance between two ICD nodes can be calculated by simply measuring the length of the maximum common prefix and averaging the rest part of ICD code pair.

The maximum common prefix between two ICD codes ICD_1 and ICD_2 can be denoted as $s(ICD_1, ICD_2)$. Then the distance $d(ICD_1, ICD_2)$ between ICD_1 and ICD_2 can be defined as:

Definition 6.1. *the half-length of the shortest distance travelled from one ICD node to another along the ICD hierarchy.*

$$d(ICD_1, ICD_2) := \frac{length(ICD_1) + length(ICD_2)}{2} - length(s(ICD_1, ICD_2))$$

6.4.1.2 The Distance between Two ICD Node Sets

Given two ICD sets, namely the prediction set $\mathbb{P}$ and the true label set $\mathbb{A}$, in order to measure the accuracy of the prediction the corresponding true ICD code should be

found to measure the distance. Formally,

$$\forall p_i \in \mathbb{P} \; \exists \, a_i \in \mathbb{A} \; s.t. \; d(p_i, a_i) = \min_{a_j \in \mathbb{A}} d(p_i, a_j)$$

.

The accuracy of the prediction presented by set $\mathbb{P}$ can be described by the followings three measurements.

1. the best distance among all nearest $< p_i, a_i >$ pairs :

$$d_{best}(\mathbb{P}, \mathbb{A}) = \min_{p_i \in \mathbb{P}} d(p_i, a_i)$$

2. the worst distance among all nearest $< p_i, a_i >$ pairs :

$$d_{worst}(\mathbb{P}, \mathbb{A}) = \max_{p_i \in \mathbb{P}} d(p_i, a_i)$$

3. the mean distance among all nearest $< p_i, a_i >$ pairs :

$$d_{mean}(\mathbb{P}, \mathbb{A}) = \frac{1}{|\mathbb{P}|} \sum_{p_i \in \mathbb{P}} d(p_i, a_i)$$

The extent to which the true ICD nodes in the true label set $\mathbb{A}$ are reflected in the prediction set $\mathbb{P}$ is also needed to measure the completeness of the output prediction, but in the reversed direction. Similarly,

$$\forall a_i \in \mathbb{A} \; \exists \, p_i \in \mathbb{P} \; s.t. \; d(a_i, p_i) = \min_{p_j \in \mathbb{P}} d(a_i, p_j)$$

The ability for the prediction set $\mathbb{P}$ of fully and accurately recalling the true ICD code in $\mathbb{A}$ can be described by the followings three measurements.

1. the best distance among all nearest $< a_i, p_i >$ pairs :

$$d_{best}(\mathbb{A}, \mathbb{P}) = \min_{a_i \in \mathbb{A}} d(a_i, p_i)$$

2. the worst distance among all nearest $< a_i, p_i >$ pairs :

$$d_{worst}(\mathbb{A}, \mathbb{P}) = \max_{a_i \in \mathbb{A}} d(a_i, p_i)$$

3. the mean distance among all nearest $< a_i, p_i >$ pairs :

$$d_{mean}(\mathbb{A}, \mathbb{P}) = \frac{1}{|\mathbb{A}|} \sum_{a_i \in \mathbb{A}} d(a_i, p_i)$$

It is worth noting that the corresponding measurements between $\mathbb{A}$ and $\mathbb{P}$ are normally not symmetric, with the only exception that:

$$d_{best}(\mathbb{P}, \mathbb{A}) \equiv d_{best}(\mathbb{A}, \mathbb{P})$$

However normally,

$$d_{worst}(\mathbb{P}, \mathbb{A}) \neq d_{worst}(\mathbb{A}, \mathbb{P})$$

$$d_{mean}(\mathbb{P}, \mathbb{A}) \neq d_{mean}(\mathbb{A}, \mathbb{P})$$

6.4.2 The Ratio of Valid Predictions

In Figure 6.10 we show the *validity* of model predictions on the patients in the test set. This means, that for each patient admission, the model predicts each ICD-10 code with a *confidence* value. Predictions can be ranked by confidence. For each predicted code, if it appears in the patient test set record, this is counted as a *valid* prediction. In particular, the top-k predictions can be valid for some subset of the patients. As we see from Figure 6.10, for the top 100 ranked predictions, approximately 4000 patients have a valid prediction in the test set.

6.4.3 Precisions and Recalls

From Figure 6.11 the results show that precision and recall exhibit a trade-off. Here, precision is the proportion of predicted codes that actually are recorded in the test set patient admission, and recall is the proportion of recorded codes that are actually predicted. Actually, the best distance-based measurement proposed above has partial connection to these two concepts. Where ICD codes from the prediction set or the true label set can be associated to the other set with different directions. We will see the relations in the figures below.

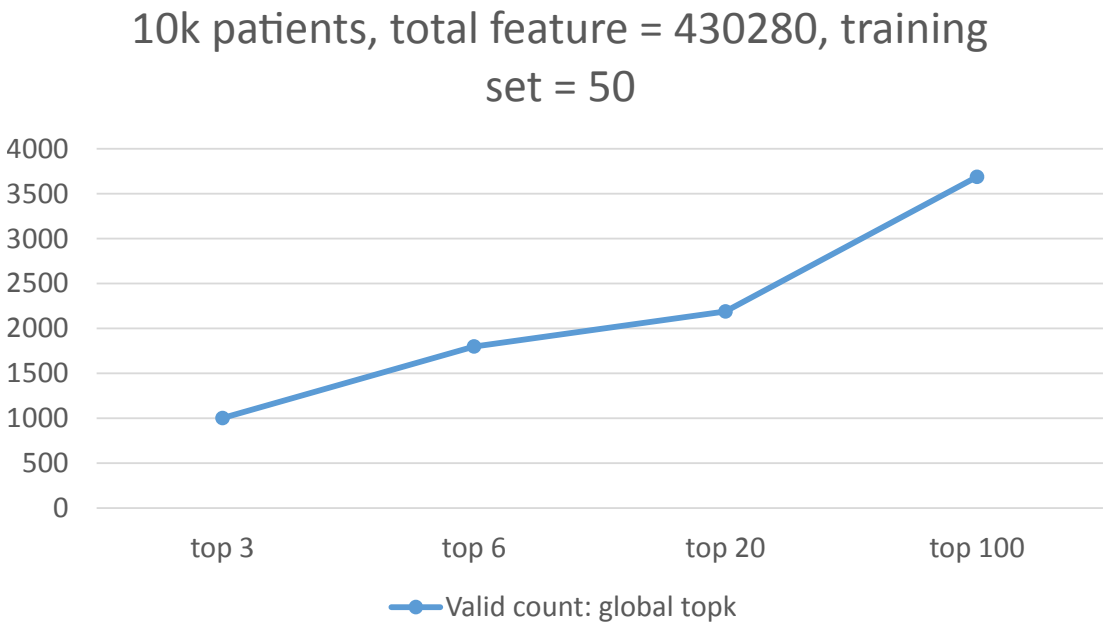

FIGURE 6.10: Number of valid predictions on test set.

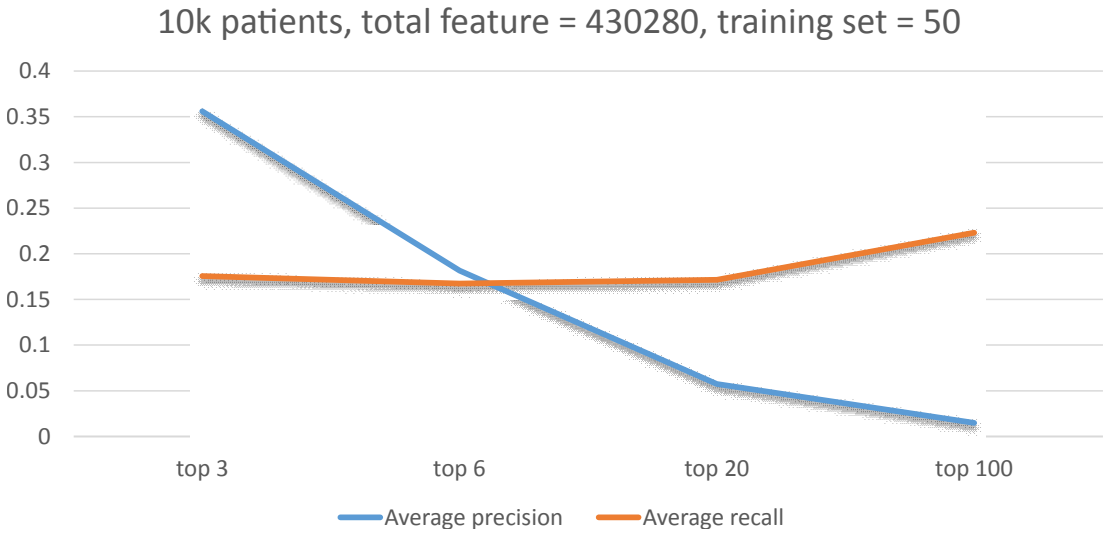

FIGURE 6.11: Precision / recall trade-off on test set predictions.

6.5 Empirical Study

In this section, we report the results and the observations got from a number of experiments, which were carried out on the EHR dataset as outlined above.

Data Set Name		Active Patient Set Size	Active Patients Generation	Training Set Size	Training Set Generation
p01	P01_25	5000	The first 5k	25	Random 25 out of first 5k
	P01_50	5000	The first 5k	50	Random 50 out of first 5k
	P01_100	5000	The first 5k	100	Random 100 out of first 5k
	P01_150	5000	The first 5k	150	Random 150 out of first 5k
p11	P11_25	5000	Random 5k	25	Random out of p11
	P11_50	5000	Random 5k	50	Random out of p11
	P11_100	5000	Random 5k	100	Random out of p11
	P11_150	5000	Random 5k	150	Random out of p11
p21	P21_25	5000	Random 5k	25	Random out of p21
	P21_50	5000	Random 5k	50	Random out of p21
	P21_100	5000	Random 5k	100	Random out of p21
	P21_150	5000	Random 5k	150	Random out of p21
p31	P31_25	5000	Random 5k	25	Random out of p31
	P31_50	5000	Random 5k	50	Random out of p31
	P31_100	5000	Random 5k	100	Random out of p31
	P31_150	5000	Random 5k	150	Random out of p31

TABLE 6.3: The dataset preparation and generation for EHR data.

6.5.1 The Data Set

In this experiment a random sample set from 25 to 150 patients out of 5000 was selected and their EHR data records for each admission was used for training. The resulting model was then tested on a second random sample of patients not used for training. We set up the data set as depicted in Table: 6.3.

6.5.2 The Challenge from the Optimization

The optimization process in machine learning is generally challenging, however, it is particularly true for EHR heterogeneous data where the feature functions are normally set up with various indicative values. See Fig: 6.12 and Fig: 6.13 for the two important values taken by the L-BFGS optimizer.

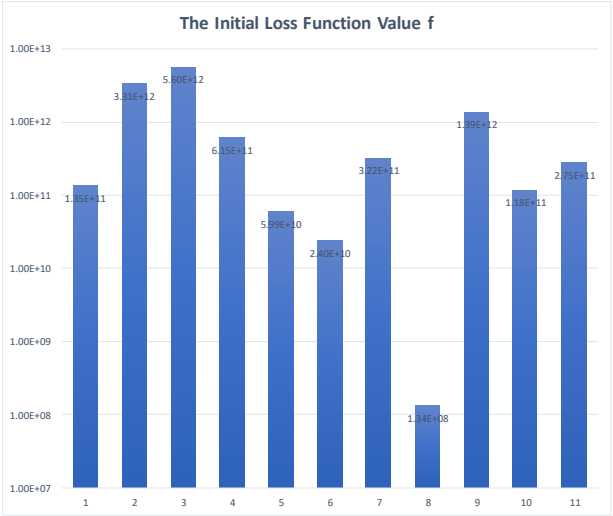

FIGURE 6.12: A distribution of initial loss function values before optimization.

Apparently, the initial values are normally huge, particularly when compared to the ultimate optimization goal - the converged f function and the norm of the gradient vector, as depicted in Fig: 6.14 and Fig: 6.15 below. Recall the sampling based method for computing the partition function, it is pointed out by several works that the complex inter-reactions between the randomized optimizer and the other part of sampling methods. We will demonstrate later that the optimizer is also critical to the TRF training method.

It is also worth noting that, the parameter setting to the optimizer is critical. Often, nowadays numerical optimizers make use of the first-generation Fortran-based

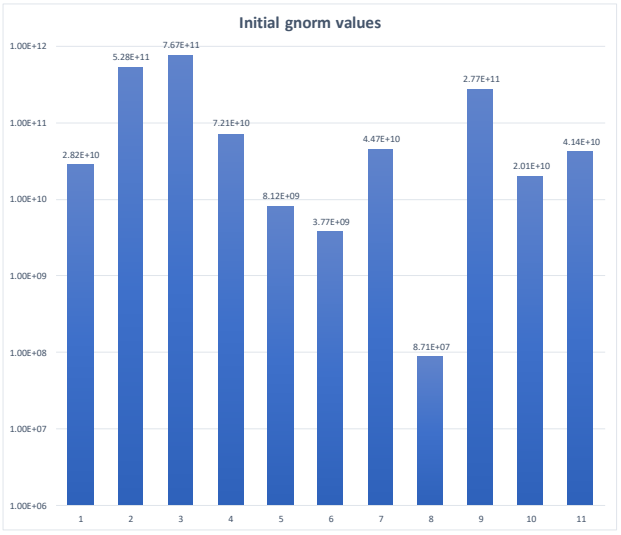

FIGURE 6.13: A distribution of initial gradient norm values before optimization.

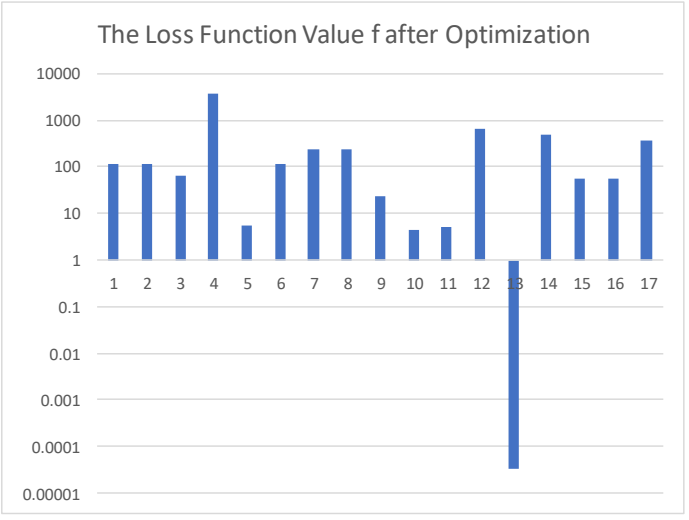

FIGURE 6.14: A distribution of loss function values after optimization.

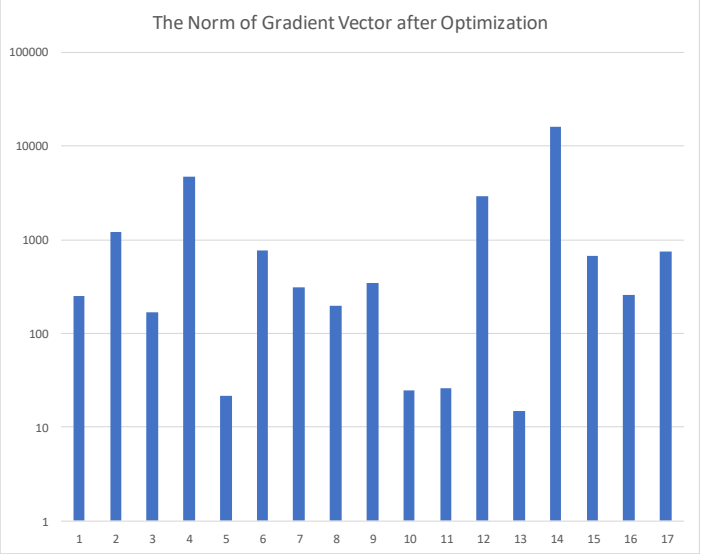

FIGURE 6.15: A distribution of gradient norm values after optimization.

implementations, e.g., the RISO L-BFGS optimizer implemented in Java following the original Fortran code. Normally these optimizers pose fairly strong stopping conditions regarding the relation between the current norm of gradient vector and the current f-value. In real-world training, particularly when the feature space is huge, those default settings might not fit.

We also had difficulties in bridging the traditional single-thread L-BFGS implementation into a multi-threading experiment code framework. We managed to implement a centralized mechanism to control the synchronisation, though it still affects the overall system throughput. Moreover, it is important to consider the power of the current machine and the balance between different workload threads in this sense.

6.5.3 The Time Efficiency Study

As listed in Table: 6.4, the time variance for the training experiments is much larger than expected, even when running on similar sized files for training. This observation leads us to consider the relation between the size of the training set and the resulting training pairs. Nevertheless, we observed that there are many possible influential factors that could take a role in the whole process. We will address these issues later.

Clearly, the time requires by different batch of experiments presents a large variation. Even considering the possible influence from the sampling based training set, there

Training Set Size	Training Time
50	27:42:43
25	6:49:50
25	19:39:43
25	0:04:55
50	40:03:59
50	10:06:53
100	35:56:48
100	12:30:03
25	3:33:26
50	6:47:21
50	2:56:28
100	32:59:11
150	32:20:39
25	0:00:03
100	15:44:29
100	43:36:38
150	47:34:35

TABLE 6.4: The training time needed for L-BFGS-based TRF learning framework to converge.

should be more decisive factors behind. Then we studied the number of iterations needed by the L-BFGS optimizer and its relation to the training size, as depicted in Fig: 6.16 below.

As depicted in Fig 6.16, more than half of the optimization processes were regarded converged simply due to an optimizer setting for the default upper limit number of steps for stop. Though in practice wo found out that this type of termination actually does not bring harm to the numerical performance after training because the optimization process would often already be in the final stages of being about to converge at the given steps.

Nevertheless, we managed to find out a clear relation among the three major factors to the overall performance, as depicted in Fig: 6.17. Given various experiment settings, the number of training pairs follow closely with the increase of training size, while the number of active features remains steady (see the yellow dots).

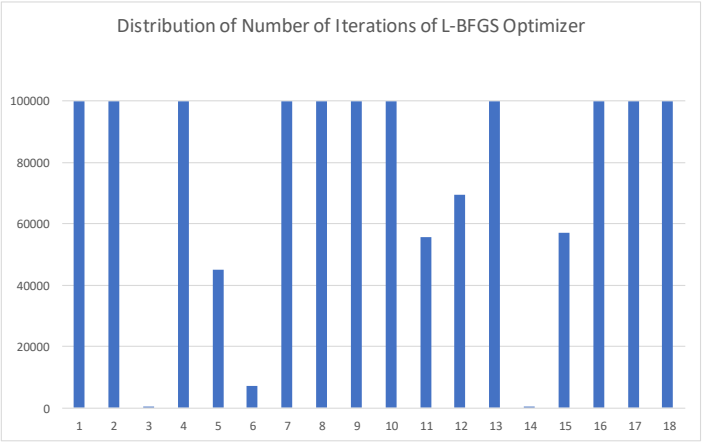

FIGURE 6.16: The number of iterations/steps required by the L-BFGS optimizer to converge.

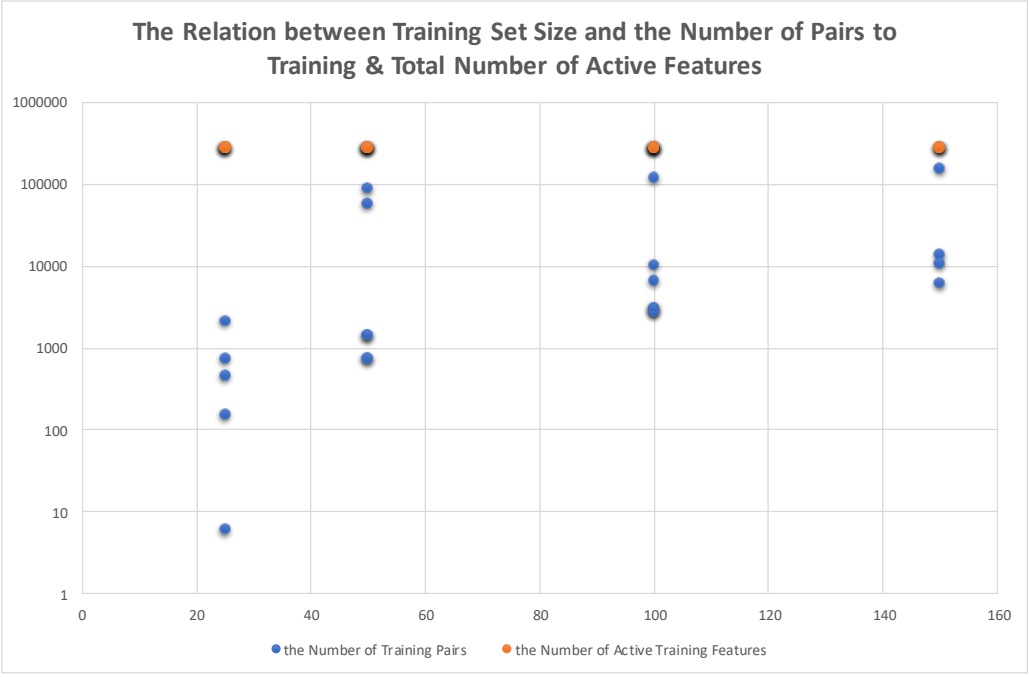

FIGURE 6.17: The direct relation between the size of training set and the resulting number of admission pairs and the total number of active features for the learning process.

However, the high level of diversions between data distributions among different patients from different training and prediction sets discussed above indeed bring extra noise to further studying the relations between major factors for the learning framework. Thus, we consider a controlled scenario where each patient data run a fixed number of steps in the LBFGS optimization to show more about the complexity of the data. The time spent on the same steps of optimizer in a steady state can reasonably eliminate the differences between datasets. We run on different sizes of training sets, which are picked from a randomized 5000 patients out of the total 58k. The results in Fig 6.18 reasonably reflects the expectation, where the time for going through 300 steps of LBFGS optimization increases with the training size.

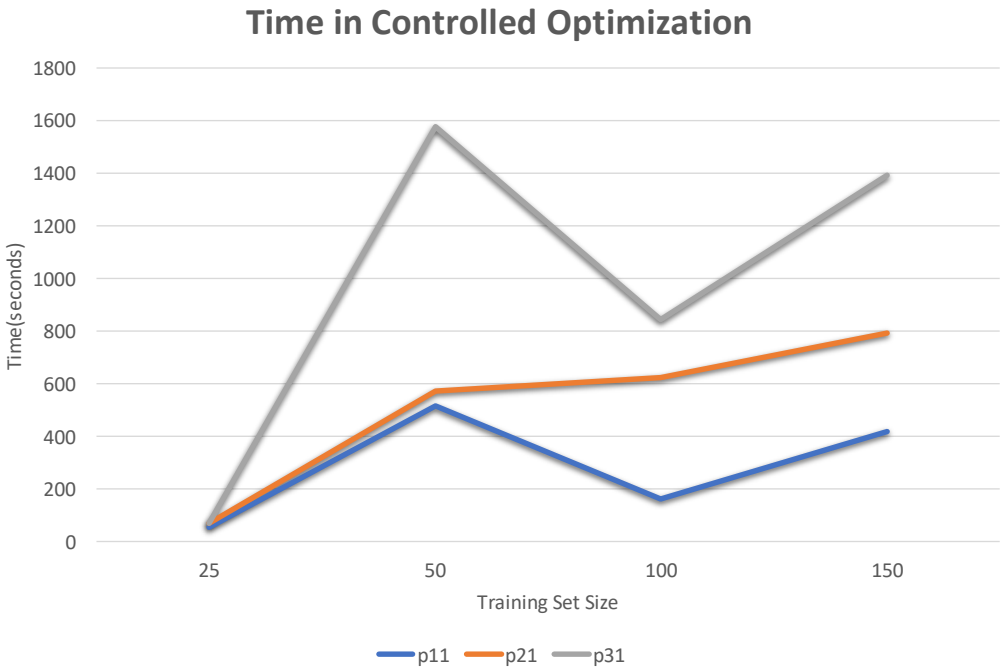

FIGURE 6.18: Time needed for 300 steps of Controlled optimization

Clearly, the learning framework cares more about the number of training pairs rather than the patient training set itself and this is reasonable because the effect of the number of training pairs will be greatly exaggerated during the iterative optimizing process.

On the efficiency study on the prediction side, as we can find in Fig: 6.19, the prediction/inference speed is relevant to the size of the training size, however, the inference process is not sensitive to the change of training size due to our proposed semantic inference techniques discussed above.

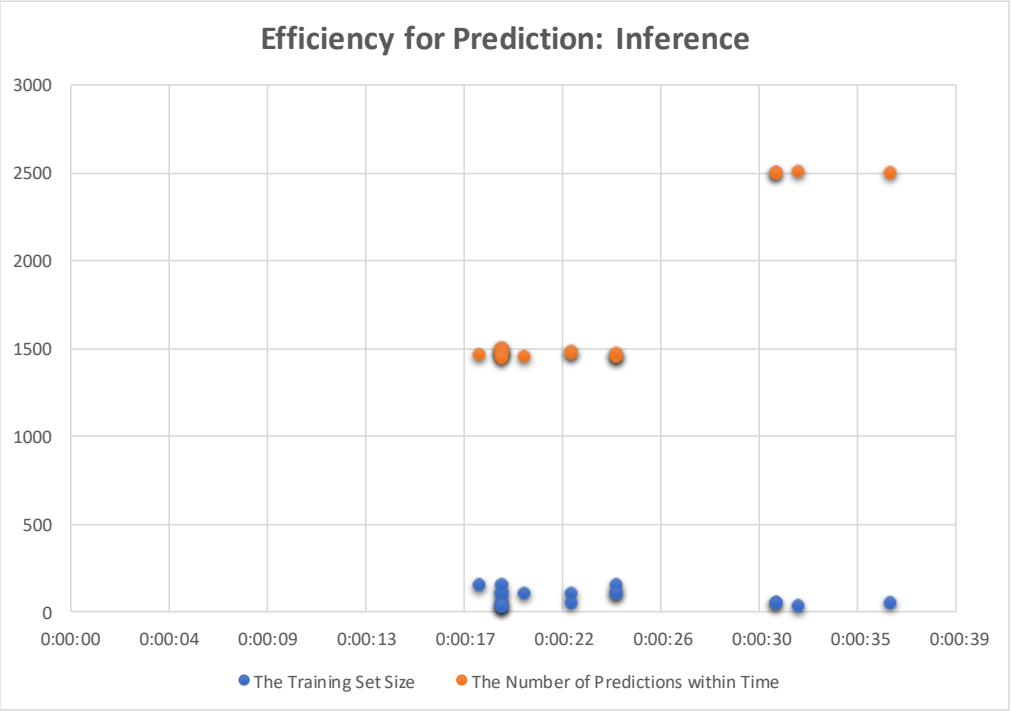

FIGURE 6.19: The efficiency in making structured prediction by inference.

6.5.4 The Effectiveness Study

We also study the effectiveness of the framework by examining the accuracy of prediction results. Recall the hierarchy-based distance measurement proposed above, it measures the distance between two positions within a same hierarchy. We adopt this measurement for evaluating the distance between the predicted ICD code and the corresponding true label.

According to the distance measurement definition, such measure is directional. We report the results utilising the directed measurement from the predicted label set to the true label set in Fig: 6.20. Clearly, the result is promising, particularly when considering the candidate ICD code set has more than 40k potential labels. As shown in the figure, for every prediction, the best predicted code should be very close to one true ICD label, with no more than 1 digit's difference. Considering the average length of a ICD code is roughly 6 digits, this prediction can give a reasonably accurate indication to what is really happening to the patient. The worst single label in each prediction though, is not ideal. The difference is between 4 to 5 digit, which actually a mapping to other branch. It is quite likely though, when there is no single ICD label

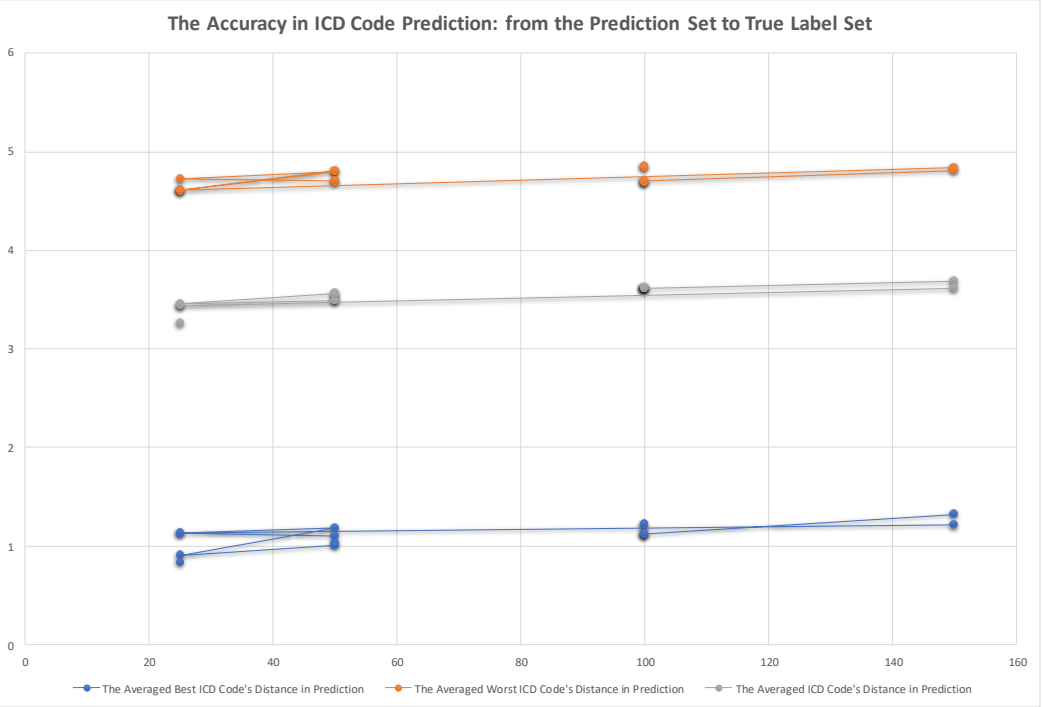

FIGURE 6.20: The accuracy in ICD code prediction when measuring from the predicted code set to the true label set.

from a branch appearing in the final set, while there is still reason to consider some remote branch. Nevertheless, for every prediction, on average, every of the predicted code is not too far away from a corresponding true label. The level of difference usually means the average code in one prediction can only map to a correct branch instead of specific ICD codes. This could largely due to the lack of complete modelling of the observation features, which could lead to the lack of capabilities in capturing signals in the input from several specific types of diseases.

The measurement on bridging from the true label set to the predicted set depicted in Fig: 6.21 shares the same pattern, but with a little improved accuracy. Actually this direction can give a more precise description to the accuracy, because a mapping from a true label to a potentially corresponding one in the predicted set can avoid mismatching between two semantic groups. Nevertheless, on average, the best predicted code in every prediction should be no more than 1 digit away from the true label.

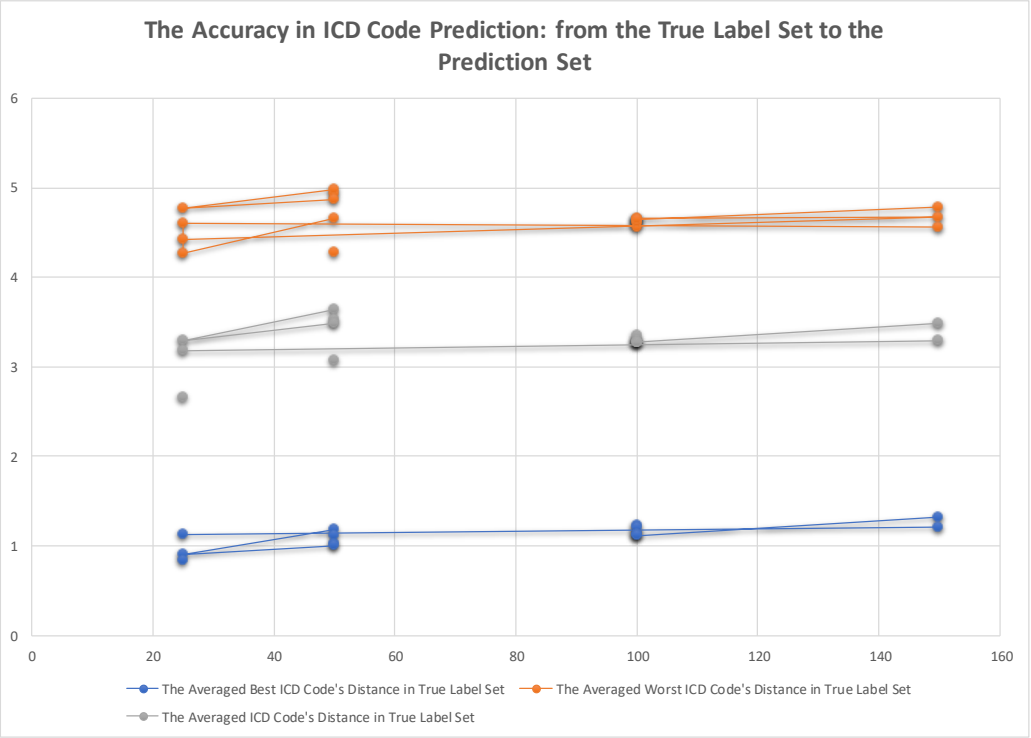

FIGURE 6.21: The accuracy in ICD code prediction when measuring from the true label set to the predicted code set.

6.5.4.1 Discussion: A Curse from the Sparsity

We can make an interesting observation here in Fig: 6.20 and Fig: 6.21: the prediction distance slightly increases with the increase of the training size. Intuitively, the increased training size should improve the prediction result, however, the experimental data proves the reverse. This observation attracted our attention and we did a thorough examining for the possible cause. It turns out that, due to the sparsity of the heterogeneous input EHR data, the feature space is also becoming more sparse, which make the optimizer more difficult to handle larger number of features concurrently. This can be directly reflected to the form of trained feature vector, where the heavily optimised feature weights either gather in small portions or are very sparsely distributed. Apparently, it is a challenging problem and would likely to exist in many general machine learning problems, particularly in a scenario where handling heterogeneous data is needed.

6.5.4.2 Discussion: A Perspective of Information Extraction

Given the experimental results on real-world record-based data, the TRF framework demonstrated the capability of extracting information from the input structure to a non-isomorphic format determined by the output structure, while at the same time utilising structurally embedded existing knowledge implicit in the output structure. The locality preserving property is a core underlying assumption which greatly limits the applicability of CRF for general information extraction. By relaxing the assumption, the proposed TRF can handle a much wider range of structured prediction problems, e.g., where the output can have non-isomorphic information extraction-style structures. No existing linear or semi-linear style CRF setting in the literature could handle output structures presented in this chapter, or the templates required for structured prediction, due to the lack of capabilities in modelling the information transition process as in TRF.

6.6 Conclusion

In this section, we report our experimental work and discuss the results. In summary, the effectiveness and efficiency of this TRF framework can be proved by the accuracy and time complexity represented in the data, particularly when considering the candidate ICD label set has more than 40k codes in the XML file we imported into the prediction framework. We also made several interesting observations regarding the performance and sparsity issue generally existing in the machine learning problems. Moreover, the experimental results motivate us to further improve the overall efficiency, particularly on the training set construction. In addition, we also demonstrate that the novel TRF model and its related training and inference techniques are effective. Our implementation can handle arbitrary connected general graph where all other traditional CRF/MRF implementations would fail.

We thus move forward to the last part of this thesis to further conclude our work presented.

7 Discussion and Future Work

In this thesis, we develop a novel probabilistic learning framework to tackle the ontology-assisted structured status prediction problem for heterogeneous data. We formalise the prediction task for EHR related problem and then develop and propose a novel probabilistic graphical model to represent this type of problems. To achieve this, we examine the knowledge abstraction and embedding process, and identify the latent connection between feature functions and the domain knowledge in a semantic context. We build up the rules for bridging the feature functions to the structural CI information provided by a graph. We also develop methodologies for representing domain knowledge with an equivalent set of feature functions.

Given the problem modelling and structural characteristics, we identify a special type of probabilistic graphical model to describe this general type of prediction problems. However, all the existing training and inference techniques fail in tackling this level of complexity. We thus develop a novel framework to do efficient training and inference.

The effectiveness and efficiency of the proposed theoretical contributions are examined and proved in the experiment chapter. Empirical study shows that the proposed model can capture the desired structural information and make accurate predictions, even on an extremely large candidate set.

In summary, the main contributions of this work include:

- exploratory data analysis and prediction learning on health record data
- ontology representation and domain knowledge embedding for structured prediction models

- heterogeneous input data modelling in structured prediction models
- an analysis of the *discriminative* learning approach to probabilistic graphical models highlighting the implicit structure on *both* the input and output components of the models
- the introduction of a new form of discriminatively learned probabilistic graphical model, the transitional random field (TRF), that relaxes this implicit structural restriction
- in terms of inference for TRFs, a key insight is how to relax the implicit *locality* constraint of CRFs during inference, by making use in TRFs of available structure in the data in a *general* way
- the derivation of a new training algorithm for TRFs, based on similarity, which avoids the computation of complex partition functions
- the implementation of an algorithm to learn TRFs using this framework
- results from an application of this algorithm to the challenging task of structural prediction of ICD-10 codes on a real sample of heterogeneous EHR data.

The work presented in this thesis could be regarded as an extension to the existing machine learning tech framework and the implementation for empirical study is also a possible tool for medical professionals for research in their domain.

7.1 Further Work

Our further work will be focused on the improvement of feature construction between input and output of TRF model. As pointed out in the experiment chapter, a complex feature function setting and the heterogenous input data have potential influence to other parts of classic learning framework, e.g., the optimization process. It would be interesting to address this sparsity feature problem in the optimization process as identified in this thesis.

We will also apply TRF to other general machine learning problems for further study, particularly where we can have rigid non-isomorphic structures on both sides of the input and output. Methodologies for building more descriptive feature functions to bridging the input and output structures in a novel way will also be in the interest of our research.

Bibliography

[ABB+00] M Ashburner, C A Ball, J A Blake, D Botstein, H Butler, J M Cherry, A P Davis, K Dolinski, S S Dwight, J T Eppig, M A Harris, D P Hill, L Issel-Tarver, A Kasarskis, S Lewis, J C Matese, J E Richardson, M Ringwald, G M Rubin, G Sherlock, and Gene Ontology Consortium. Gene Ontology: Tool for The Unification of Biology. *Nature Genetics*, 25(1):25–29, may 2000.

[ACZ+13] Carlos A Alvarez, Christopher A Clark, Song Zhang, Ethan A Halm, John J Shannon, Carlos E Girod, Lauren Cooper, and Ruben Amarasingham. Predicting out of intensive care unit cardiopulmonary arrest or death using electronic medical record data. *BMC medical informatics and decision making*, 13(1):1, 2013.

[AEP07] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-Task Feature Learning. In *Advances in Neural Information Processing Systems*, volume 19, page 41, 2007.

[ARS05] Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. Emotions from text: machine learning for text-based emotion prediction. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05*, pages 579–586. Association for Computational Linguistics, 2005.

[ATH+13] Peter C. Austin, Jack V. Tu, Jennifer E. Ho, Daniel Levy, and Douglas S. Lee. Using methods from the data mining and machine learning literature for disease classification and prediction: A case study examining

classification of heart failure sub-types. *Journal of clinical epidemiology*, 66(4):398–407, 2013.

[BC01] G Boysen and H Christensen. Stroke severity determines body temperature in acute stroke. *Stroke*, 32(2):413–417, 2001.

[BEP+08] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[Bes74] Julian Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192—-236, 1974.

[BHML16] Ghazaleh Beigi, Xia Hu, Ross Maciejewski, and Huan Liu. An Overview of Sentiment Analysis in Social Media and Its Applications in Disaster Relief. In *Sentiment Analysis and Ontology Engineering*, pages 313–340. Springer, 2016.

[Bis07] Christopher M. Bishop. *Pattern Recognition And Machine Learning*. Number 4 in Information science and statistics. Springer-Verlag New York, 1 edition, 2007.

[BK11] Wei Bi and James T Kwok. Multi-label classification on tree-and DAG-structured hierarchies. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 17–24, Bellevue, Washington, USA, 2011. ACM.

[BKGOM11] Aziz A Boxwala, Jihoon Kim, Janice M Grillo, and Lucila Ohno-Machado. Using statistical and machine learning to help institutions detect suspicious access to electronic health records. *Journal of the American Medical Informatics Association (JAMIA)*, 18(4):498–505, 2011.

[BKR11] Andrew Blake, Pushmeet Kohli, and Carsten Rother. *Markov random fields for vision and image processing*. Mit Press, 2011.

[BM98] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM, 1998.

[BRM14] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. Continuous conditional neural fields for structured regression. In *European Conference on Computer Vision*, pages 593–608. Springer, 2014.

[BRR+13] Dustin W. Ballard, Adina S. Rauchwerger, Mary E. Reed, David R. Vinson, Dustin G. Mark, Steven R. Offerman, Uli K. Chettipally, Ilana Graetz, Peter Dayan, and Nathan Kuppermann. Emergency physicians' knowledge and attitudes of clinical decision support in the electronic health record: A survey-based study. *Academic Emergency Medicine*, 20(4):352–360, 2013.

[BSM79] David P Byar, Mary E Sears, and William L McGuire. Relationship between estrogen receptor values and clinical data in predicting the response to endocrine therapy for patients with advanced breast cancer. *European Journal of Cancer (1965)*, 15(3):299–310, 1979.

[BZ08] Riccardo Bellazzi and Blaz Zupan. Predictive data mining in clinical medicine: Current issues and guidelines. *International Journal of Medical Informatics*, 77(2):81–97, 2008.

[CA15] Karla Caballero and Ram Akella. Dynamically Modeling Patient's Health State from Electronic Medical Records: A Time Series Approach. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, pages 69–78, 2015.

[CAM76] J K Clayton, J A Anderson, and G P McNicol. Preoperative prediction of postoperative deep vein thrombosis. *Br Med J*, 2(6041):910–912, 1976.

[CAR97] RICH CARUANA. Multitask Learning. *Machine Learning*, 28:41–75, 1997.

[CBS+16a] Edward Choi, Mohammad Taha Bahadori, Andy Schuetz, Walter F. Stewart, Jimeng Sun, Andy Schuetz, Walter F. Stewart, Jimeng Sun, Andy Schuetz, Walter F. Stewart, and Jimeng Sun. Doctor AI: Predicting Clinical Events via Recurrent Neural Networks. *Proceedings of Machine Learning for Healthcare 2016 JMLR W&C Track*, 56:1–12, 2016.

[CBS+16b] Edward Choi, Mohammad Taha Bahadori, Elizabeth Searles, Catherine Coffey, Michael Thompson, James Bost, Javier Tejedor-Sojo, and Jimeng Sun. Multi-layer Representation Learning for Medical Concepts. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 1495–1504, 2016.

[CBWB12] Paul E Cotter, Vikas K Bhalla, Stephen J Wallis, and Richard W S Biram. Predicting readmissions: poor performance of the LACE index in an older UK population. *Age and ageing*, 41(6):784–789, 2012.

[CCCM10] Lwc W C Chan, T Chan, Lf F Cheng, and Ws S Mak. Machine learning of patient similarity: A case study on predicting survival in cancer patient after locoregional chemotherapy. In *Bioinformatics and Biomedicine Workshops (BIBMW), 2010 IEEE International Conference on*, pages 467–470. IEEE, Ieee, dec 2010.

[CFB05] H Christensen, A Fogh Christensen, and G Boysen. Abnormalities on ECG and telemetry predict stroke outcome at 3 months. *Neurocrit. Care*, 234(1-2):99–103, 2005.

[CGB+15] You Chen, Joydeep Ghosh, Cosmin Adrian Bejan, Carl A. Gunter, Siddharth Gupta, Abel Kho, David Liebovitz, Jimeng Sun, Joshua Denny, and Bradley Malin. Building bridges across electronic health record systems through inferred phenotypic topics. *Journal of Biomedical Informatics*, 55:82–93, 2015.

[Cha07] Sung-hyuk Cha. Comprehensive Survey on Distance / Similarity Measures between Probability Density Functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4):300–307, 2007.

[Che08] Samson Cheung. Proof of hammersley-clifford theorem. Technical report, UNIVERSITY OF KENTUCKY, 2008.

[CJ83] George R Cross and Anil K Jain. Markov random field texture models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, pages 25–39, 1983.

[Cli90] Peter Clifford. Markov random fields in statistics. *Disorder in physical systems: A volume in honour of John M. Hammersley*, pages 19–32, 1990.

[CLSB11] L.W.C. Chan, Y. Liu, C.R. Shyu, and I.F.F. Benzie. A SNOMED supported ontological vector model for subclinical disorder detection using EHR similarity. *Engineering Applications of Artificial Intelligence*, 24(8):1398–1409, dec 2011.

[CPCC+16] Xiongcai Cai, Oscar Perez-Concha, Enrico Coiera, Fernando Martin-Sanchez, Richard Day, David Roffe, and Blanca Gallego. Real-time

prediction of mortality, readmission, and length of stay using electronic health record data. *Journal of the American Medical Informatics Association - JAMIA*, 23(3):553—-561, 2016.

[CR14] Mark Chignell and Mahsa Rouzbahman. A Search Engine for Structured Health Data. Technical report, University of Toronto, jun 2014.

[CSC+13] Mark E Cowen, Robert L Strawderman, Jennifer L Czerwinski, Mary Jo Smith, and Lakshmi K Halasyamani. Mortality predictions on admission as a context for organizing care activities. *Journal of hospital medicine*, 8(5):229–235, 2013.

[DBA+15] Wuyang Dai, Theodora S Brisimi, William G Adams, Theofanie Mela, Venkatesh Saligrama, and Ioannis Ch Paschalidis. Prediction of hospitalization due to heart diseases by supervised learning methods. *International journal of medical informatics*, 84(3):189—-197, 2015.

[DJW16] Y. Dai, W. Jiang, and G. Wang. Building Bayesian Inference Graphs for Healthcare Statistic Evidence. In *Proceedings of the International Conference on Parallel Processing Workshops*, pages 415—-420, 2016.

[dLLRN98] Luciano R. S. de Lima, Alberto H. F. Laender, and Berthier A. Ribeiro-Neto. A hierarchical approach to the automatic categorization of medical documents. In *Proceedings of the seventh international conference on Information and knowledge management*, pages 132–139. ACM, 1998.

[DLN+09] B B Dean, J Lam, J L Natoli, Q Butler, D Aguilar, and R J Nordyke. Review: use of electronic medical records for health outcomes research: a literature review. *Med Care Res Rev*, 66:611–638, 2009.

[DMR+00] Suzanne L Dawson, Bradley N Manktelow, Thompson G Robinson, Ronney B Panerai, and John F Potter. Which parameters of beat-to-beat blood pressure and variability best predict early outcome after acute ischemic stroke? *Stroke*, 31(2):463–468, 2000.

[Don06] Kevin Donnelly. SNOMED-CT: The advanced terminology and coding system for eHealth. *Studies in Health Technology and Informatics*, 121:279 – 290, 2006.

[EA12] Philippe Esling and Carlos Agon. Time-series data mining. *ACM Computing Surveys*, 45(1):1–34, nov 2012.

[ENL+16] Huseyin Melih Elibol, Vincent Nguyen, Scott Linderman, Matthew Johnson, Amna Hashmi, and Finale Doshi-Velez. Cross-Corpora Unsupervised Learning of Trajectories in Autism Spectrum Disorders. *Journal of Machine Learning Research*, 17(133):1–38, 2016.

[EW01] Andre Elisseeff and Jason Weston. A kernel method for multi-labelled classification. In *Advances in neural information processing systems*, pages 681–687, 2001.

[FC93] Chun Hsiung Fang and Fan Ren Chang. Analysis of stability robustness for generalized state-space systems with structured perturbations. *Systems and Control Letters*, 21(2):109–114, 1993.

[FHLB08] Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, and Klaus Brinker. Multilabel classification via calibrated label ranking. *Machine Learning*, 73(2):133–153, 2008.

[FJJ+12] Fei Wang, Jianying Hu, Jimeng Sun, F Wang, J Hu, and J Sun. Medical prognosis based on patient similarity and expert feedback. In *21st International Conference on Pattern Recognition*, pages 1799–1802, 2012.

[FMG+12] Roberto Forero, Geoff McDonnell, Blanca Gallego, Sally McCarthy, Mohammed Mohsin, Chris Shanley, Frank Formby, and Ken Hillman. A Literature Review on Care at the End-of-Life in the Emergency Department. *Emergency Medicine International*, 2012:1–11, 2012.

[FPM16] E Fersini, F A Pozzi, and E Messina. Approval network: a novel approach for sentiment analysis in social networks. *World Wide Web*, pages 1–24, 2016.

[FPY+16] Ruogu Fang, Samira Pouyanfar, Yimin Yang, Shu-Ching Chen, and S. S. Iyengar. Computational Health Informatics in the Big Data Age: a survey. *ACM Computing Surveys (CSUR)*, 49(1):1–36, 2016.

[Fra16] Abraham Jacob Frandsen. *Machine Learning for Disease Prediction*. All theses and dissertations, for master of science, Brigham Young University, 2016.

[FS08] Richárd Farkas and György Szarvas. Automatic construction of rule-based ICD-9-CM coding systems. *BMC bioinformatics*, 9(3):S10, 2008.

[FSCH16] Joseph Futoma, Mark Sendak, C Blake Cameron, and Katherine Heller. Predicting Disease Progression with a Model for Multivariate Longitudinal Clinical Data. In *Proceedings of the 1st Machine Learning for Healthcare Conference*, pages 42—-54, 2016.

[GAM+05] Amit X Garg, Neill K J Adhikari, Heather McDonald, M Patricia Rosas-Arellano, P J Devereaux, Joseph Beyene, Justina Sam, and R Brian Haynes. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *Jama*, 293(10):1223–1238, 2005.

[Gar13] E Gardner. The HIT approach to big data. *Health data management*, 21(3):34–36, 2013.

[GBH09] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1:12, 2009.

[GBZW93] Thomas Galski, Richard L Bruno, Richard Zorowitz, and John Walker. Predicting length of stay, functional outcome, and aftercare in the rehabilitation of stroke patients. The dominant role of higher-order cognition. *Stroke*, 24(12):1794–1800, 1993.

[GN87] Michael R Genesereth and Nils J Nilsson. *Logical Foundations of Artificial Intelligence*, volume 55. Springer, 1987.

[GNDV+14] Marzyeh Ghassemi, Tristan Naumann, Finale Doshi-Velez, Nicole Brimmer, Rohit Joshi, Anna Rumshisky, and Peter Szolovits. Unfolding physiological state: Mortality modelling in intensive care units. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 75–84. ACM, 2014.

[GNPI17] Benjamin A Goldstein, Ann Marie Navar, Michael J Pencina, and John PA Ioannidis. Opportunities and challenges in developing risk prediction models with electronic health records data: a systematic review. *Journal of the American Medical Informatics Association : JAMIA*, 24(1):198–208, 2017.

[Gon16] André Ricardo Gonçalves. *Sparse and Structural Multitask Learning*. PhD thesis, the University of Campinas, 2016.

[GRB13] P Gilbert, M D Rutland, and D Brockopp. Redesigning the work of case management: testing a predictive model for readmission. *The American journal of managed care*, 19(10 Spec No):eS19—-eSP25, 2013.

[Gru93] Thomas R Gruber. A Translation Approach to Portable Ontology Specifications. *Knowledge acquisition*, 5(April):199–220, 1993.

[GSN+04] A R Gujjar, T N Sathyaprabha, D Nagaraja, K Thennarasu, and N Pradhan. Heart rate variability and outcome in acute severe stroke: role of power spectral analysis. *Neurocriti. Care*, 1(3):347–353, 2004.

[GSR+13] Assaf Gottlieb, Gideon Y Stein, Eytan Ruppin, Russ B Altman, and Roded Sharan. A method for inferring medical diagnoses from patient similarities. *BMC medicine*, 11(1):194, jan 2013.

[GSRS11] Assaf Gottlieb, GY Gideon Y Stein, Eytan Ruppin, and Roded Sharan. PREDICT: a method for inferring novel drug indications with application to personalized medicine. *Molecular Systems Biology*, 7(1):496, 2011.

[GT05] Tracy D Gunter and Nicolas P Terry. The emergence of national electronic health record architectures in the United States and Australia: models, costs, and questions. *Journal of Medical Internet Research*, 7(1):e3, 2005.

[HA13] George Hripcsak and David J Albers. Next-generation phenotyping of electronic health records. *Journal of the American Medical Informatics Association : JAMIA*, 20(1):117—-121, 2013.

[Ham94] James Douglas Hamilton. *Time series analysis*, volume 2. Princeton university press, 1994.

[HC71] John M Hammersley and Peter Clifford. Markov Fields on Finite Graphs and Lattices, 1971.

[HDD15] Zhengxing Huang, Wei Dong, and Huilong Duan. A probabilistic topic model for clinical risk stratification from electronic health records. *Journal of Biomedical Informatics*, 58:28–36, 2015.

[HHHS98] Dereck L Hunt, R Brian Haynes, Steven E Hanna, and Kristina Smith. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. *Jama*, 280(15):1339–1346, 1998.

[HHS00] C Hajat, S Hajat, and P Sharma. Effects of poststroke pyrexia on stroke outcome: a meta-analysis of studies in patients. *J. Intensive Care Med.*, 31(2):410–414, 2000.

[HLG12] Joyce C Ho, Cheng H Lee, and Joydeep Ghosh. Imputation-Enhanced Prediction of Septic Shock in ICU Patients Categories and Subject Descriptors. *HI-KDD 2012: ACM SIGKDD Workshop on Health Informatics*, 2012.

[HLS13] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*, volume 398. John Wiley & Sons, 2013.

[HNC65] Frank Harary, Robert Z Norman, and Dorwin Cartwright. Structural models, 1965.

[HNN+14] Ilkka Huopaniemi, Girish Nadkarni, Rajiv Nadukuru, Vaneet Lotay, Steve Ellis, Omri Gottesman, and Erwin P Bottinger. Disease progression subtype discovery from longitudinal EMR data with a majority of missing values and unknown initial time points. In *AMIA Annual Symposium Proceedings*, volume 2014, pages 709–18, 2014.

[Hug09] C Hug. *Detecting hazardous intensive care patient episodes using real-time mortality models*. PhD thesis, MIT, 2009.

[Hwa99] CH Hwang. Incompletely and imprecisely speaking: Using dynamic ontologies for representing and retrieving information. *Proceedings of the 6th International Workshop on ...*, pages 1–12, 1999.

[HY14] Robert E Hoyt and Ann K Yoshihashi. *Health Informatics: Practical guide for healthcare and information technology professionals*. Lulu. com, 6 edition, 2014.

[HYBV05] Lynette Hirschman, Alexander Yeh, Christian Blaschke, and Alfonso Valencia. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, 6(Suppl 1):S1, 2005.

[HZCP04] Xuming He, Richard S. Zemel, and Miguel Á. Carreira-Perpiñán. Multiscale conditional random fields for image labeling. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - CVPR'04*, volume 2, pages 695–702, 2004.

[Inv88] W H O MONICA Project Principal Investigators. The World Health Organization MONICA Project (monitoring trends and determinants in cardiovascular disease): a major international collaboration. *J. Clin. Epidemiol.*, 41(2):105–114, 1988.

[Jan16] Bas Janssen. *Determining truth in tweets using feature based supervised statistical classifiers*. PhD thesis, University of Twente, 2016.

[Jay65] Edward T Jaynes. Gibbs vs Boltzmann entropies. *American Journal of Physics*, 33(5):391–398, 1965.

[JG11] Heng Ji and Ralph Grishman. Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1148–1158. Association for Computational Linguistics, 2011.

[JGN+16] A. E. W. Johnson, M. M. Ghassemi, S. Nemati, K. E. Niehaus, D. A. Clifton, and G. D. Clifford. Machine Learning and Decision Support in Critical Care. *Proceedings of the IEEE*, 104(2):444–466, 2016.

[JJB12] Peter B. Jensen, Lars J. Jensen, and Søren Brunak. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405, 2012.

[JJQK+16] Fleur Jeanquartier, Claire Jean-Quartier, Max Kotlyar, Tomas Tokar, Anne-Christin Hauschild, Igor Jurisica, and Andreas Holzinger. Machine Learning for In Silico Modeling of Tumor Growth. In *Machine Learning for Health Informatics*, pages 415–434. Springer, 2016.

[JP90] G M Joseph and V L Patel. Domain Knowledge and Hypothesis Generation in Diagnostic Reasoning. *Medical Decision Making*, 10(1):31, 1990.

[JPS+16] Alistair E W Johnson, Tom J Pollard, Lu Shen, Li-Wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[JTB+76] Bond Jennett, G Teasdale, R Braakman, J Minderhoud, and R Knill-Jones. Predicting outcome in individual patients after severe head injury. *The Lancet*, 307(7968):1031–1034, 1976.

[JTYY10] Shuiwang Ji, Lei Tang, Shipeng Yu, and Jieping Ye. A shared-subspace learning framework for multi-label classification. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(2):1–29, 2010.

[KCP11] Mohammed Khalilia, Sounak Chakraborty, and Mihail Popescu. Predicting disease risks from highly imbalanced data using random forest. *BMC medical informatics and decision making*, 11(51), 2011.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT Press, Cambridge, 2009.

[KHBL05] K Kawamoto, C A Houlihan, E A Balas, and D F Lobach. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. *BMJ*, 330:765, 2005.

[Kim14] Minyoung Kim. Conditional ordinal random fields for structured ordinal-valued label prediction. *Data mining and knowledge discovery*, 28(2):378–401, 2014.

[Kim16] Minyoung Kim. Sparse Conditional Copula Models for Structured Output Regression. *Pattern Recognition*, 2016.

[KK13] Philipp Krähenbühl and Vladlen Koltun. Parameter Learning and Convergent Inference for Dense Random Fields. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, volume 28, pages 513—-521, 2013.

[KKV13] Basel Kayyali, David Knott, and Steve Van Kuiken. The big-data revolution in US health care: Accelerating value and innovation. *Mc Kinsey & Company*, pages 1–13, 2013.

[KKZ12] Michał Kacprzak, Michał Kidawa, and Marzenna Zielińska. Fever in myocardial infarction: is it still common, is it still predictive? *Cardiol J*, 19(4):369–373, 2012.

[KPQ02] Zoltan Kato, Ting-Chuen Pong, and Song Guo Qiang. Multicue MRF image segmentation: Combining texture and color features. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 1, pages 660–663. IEEE, 2002.

[KRN+91] H A Katus, A Remppis, F J Neumann, T Scheffold, K W Diederich, G Vinar, A Noe, G Matern, and W Kuebler. Diagnostic efficiency of troponin T measurements in acute myocardial infarction. *Circulation*, 83:902–912, 1991.

[KSS+14] Dokyoon Kim, Hyunjung Shin, Kyung-Ah Sohn, Anurag Verma, Marylyn D Ritchie, and Ju Han Kim. Incorporating inter-relationships between different levels of genomic data into cancer clinical outcome prediction. *Methods*, 67(3):344–353, 2014.

[KWHS13] Edward H Kennedy, Wyndy L Wiitala, Rodney A Hayward, and Jeremy B Sussman. Improved cardiovascular risk prediction using nonparametric regression and electronic health record data. *Medical care*, 51(3):251–258, 2013.

[LAY+16] Muhammad K Lodhi, Rashid Ansari, Yingwei Yao, Gail M Keenan, Diana J Wilkie, and Ashfaq Khokhar. A Framework to Predict Outcome for Cancer Patients Using data from a Nursing EHR. In *2016 IEEE International Conference on Big Data (Big Data) A*, pages 3387–3395, 2016.

[LC95] Leah S Larkey and W Bruce Croft. Automatic assignment of icd9 codes to discharge summaries. Technical report, Technical report, University of Massachusetts at Amherst, Amherst, MA, 1995.

[Li01] Stan Z. Li. *Markov Random Field Modeling in Image Analysis*. Computer Science Workbench. Springer Science {&} Business Media, 3rd ed edition, 2001.

[Li12] S.Z. Li. *Markov Random Field Modeling in Computer Vision*. Computer Science Workbench. Springer Science & Business Media, 2012.

[LIT92] Pat Langley, Wayne Iba, and Kevin Thomposn. An Analysis of Bayesian Classifers. *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 223–228, 1992.

[Liu15] Fayao Liu. *Learning Structured Prediction Models in Computer Vision*. PhD thesis, The University of Adelaide, 2015.

[LJ09] Changki LEE and Myung-Gil JANG. Fast Training of Structured SVM Using Fixed-Threshold Sequential Minimal Optimization. *ETRI journal*, 31(2):121–128, 2009.

[LLS+15] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *AAAI*, pages 2181–2187, 2015.

[LMP01] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning - ICML'01*, ICML '01, pages 282–289, San Francisco, CA, USA, jun 2001. Morgan Kaufmann Publishers Inc.

[LPZ08] Bo Long, S Yu Philip, and Zhongfei (Mark) Zhang. A General Model for Multiple View Unsupervised Learning. In *SDM*, pages 822–833, 2008.

[LRKT09] ubor Ladický, Chris Russell, Pushmeet Kohli, and Philip H.S. Torr. Associative hierarchical CRFs for object class image segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 739–746, 2009.

[LRPV16] Cheng Li, Santu Rana, Dinh Phung, and Svetha Venkatesh. Hierarchical Bayesian nonparametric models for knowledge discovery from electronic medical records. *Knowledge-Based Systems*, 99:168–182, 2016.

[LSF00] Yves a. Lussier, Lyudmila Shagina, and Carol Friedman. Automating ICD-9-CM Encoding Using Medical Language Processing: A Feasibility Study. *Proceedings of the AMIA Symposium*, 1:1072, 2000.

[LUW+14] Stefan Lang, Nikolaus Umlauf, Peter Wechselberger, Kenneth Harttgen, and Thomas Kneib. Multilevel structured additive regression. *Statistics and Computing*, 24(2):223–238, 2014.

[LZL+16] Runzhi Li, Hongling Zhao, Yusong Lin, Andrew Maxwell, Chaoyang Zhang, and Yusong Lin. Multi-label classification for intelligent health risk prediction. In *Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on*, pages 986–993. IEEE, 2016.

[LZSI12] Manhua Liu, Daoqiang Zhang, Dinggang Shen, and Alzheimer's Disease Neuroimaging Initiative. Ensemble Sparse Classification of Alzheimer's Disease. *NeuroImage*, 60(2):1106—-1116, 2012.

[MBP05] Andrew Mccallum, Kedar Bellare, and Fernando Pereira. A conditional random field for discriminatively-trained finite-state string edit distance. In *Conference on Uncertainty in AI (UAI)*, jul 2005.

[MD13] Travis B Murdoch and Allan S Detsky. The inevitable application of big data to health care. *Jama*, 309(13):1351–1352, 2013.

[MDLS+13] M. A. McAdams-DeMarco, A. Law, M.L. Salter, E. Chow, M. Grams, J. Walston, and D.L. Segev. Frailty and early hospital readmission after kidney transplantation. *American Journal of Transplantation*, 13(8):2091–2095, 2013.

[MFBG+03] J Martí-Fàbregas, R Belvís, E Guardia, D Cocho, J Muñoz, L Marruecos, and J-L Martí-Vilalta. Prognostic value of Pulsatility Index in acute intracerebral hemorrhage. *Neurology*, 61(8):1051–6, 2003.

[Mil95] George A Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[MKE+11] D M Mann, J L Kannry, D Edonyabo, A C Li, J Arciniega, J Stulman, L Romero, J Wisnivesky, R Adler, and T G McGinn. Rationale, design, and implementation protocol of an electronic health record integrated clinical prediction rule (iCPR) randomized trial in primary care. *Implementation science : IS*, 6(109), 2011.

[MKR+16] Sidra Minhas, Aasia Khanum, Farhan Riaz, Atif Alvi, and Shoab A. Khan. A Non Parametric Approach for Mild Cognitive Impairment to AD Conversion Prediction: Results on Longitudinal Data. *IEEE Journal of Biomedical and Health Informatics*, page 1, 2016.

[MKY11] Abdullah Mueen, Eamonn Keogh, and Neal Young. Logical-shapelets: an expressive primitive for time series classification. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1154–1162. ACM, 2011.

[MMMS+13] Mar Marcos, Jose A Maldonado, Begoña Martínez-Salvador, Diego Boscá, and Montserrat Robles. Interoperability of clinical decision-support systems and electronic health records using archetypes: a case study in clinical trial eligibility. *Journal of biomedical informatics*, 46(4):676–689, 2013.

[Mou74] John Moussouris. Gibbs and Markov random systems with constraints. *Journal of statistical physics*, 10(1):11–33, 1974.

[MS99] Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT Press, 1999.

[MS02] Alexander Maedche and Steffen Staab. Ontology Learning for the Semantic Web. *Kluwer Academic Publishers, USA*, page 18, 2002.

[Mur12] Kevin P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[MV01] Alexander Maedche and Raphael Volz. The ontology extraction & maintenance framework Text-To-Onto. *Proc. Workshop on Integrating Data ...*, pages 1–12, 2001.

[MWHT14] Robert Moskovitch, Colin Walsh, George Hripsack, and Nicholas Tatonetti. Prediction of Biomedical Events via Time Intervals Mining. *ACM KDD Workshop on Connected Health in Big Data Era*, 2014.

[NGL10] Sebastian Nowozin, Peter V. Gehler, and Christoph H. Lampert. On parameter learning in CRF-based approaches to object class image segmentation. In *11th European Conference on Computer Vision (ECCV '10)*, pages 98–111. Springer, 2010.

[NKY+15] Nozomi Nori, Hisashi Kashima, Kazuto Yamashita, Hiroshi Ikai, and Yuichi Imanaka. Simultaneous modeling of multiple diseases for mortality prediction in acute hospital care. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2015.

[NL11] Sebastian Nowozin and Christoph H. Lampert. Structured Learning and Prediction in Computer Vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(3-4):185–365, 2011.

[NZM+16] Liqiang Nie, Luming Zhang, Lei Meng, Xuemeng Song, Xiaojun Chang, and Xuelong Li. Modeling Disease Progression via Multisource Multitask Learners: A Case Study With Alzheimer's Disease. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–12, 2016.

[OE16] Ziad Obermeyer and Ezekiel J. Emanuel. Predicting the Future Big Data, Machine Learning, and Clinical Medicine. *The New England journal of medicine*, 375(13):1216–1219, 2016.

[OFST+03] J Oliveira-Filho, S C S Silva, C C Trabuco, B B Pedreira, E U Sousa, and A Bacellar. Detrimental effect of blood pressure reduction in the first 24 hours of acute stroke onset. *Neurology*, 61(8):1047–1051, 2003.

[OKMI07] Shigeyuki Oba, Motoaki Kawanabe, Klaus-Robert Müller, and Shin Ishii. Heterogeneous Component Analysis. In J C Platt, D Koller, Y Singer, and S T Roweis, editors, *Advances in Neural Information Processing Systems*, pages 1097–1104. Curran Associates, Inc., 2007.

[OLC+16] Richard J Oentaryo, Ee-Peng Lim, Freddy Chong Tat Chua, Jia-Wei Low, and David Lo. Collective Semi-Supervised Learning for User Profiling in Social Media. *arXiv preprint arXiv:1606.07707*, 2016.

[OLSH12] Tim O'Reilly, Mike Loukides, Julie Steele, and Colin Hill. *How data science is transforming health care.* " O'Reilly Media, Inc.", 2012.

[PBX09] Jian Peng, Liefeng Bo, and Jinbo Xu. Conditional neural fields. In *Advances in neural information processing systems*, pages 1419–1427, 2009.

[PE15] Rimma Pivovarov and Noémie Elhadad. Automated methods for the summarization of electronic health records. *Journal of the American Medical Informatics Association*, 22(5):938–947, 2015.

[PGAJ06] Callum B Pearce, Steve R Gunn, Adil Ahmed, and Colin D Johnson. Machine learning can improve prediction of severity in acute pancreatitis using admission values of APACHE II score and C-reactive protein. *Pancreatology*, 6(1-2):123–131, 2006.

[PK11] Mihail Popescu and Mohammad Khalilia. Improving disease prediction using ICD-9 ontological features. *IEEE International Conference on Fuzzy Systems*, pages 1805–1809, 2011.

[PMGC13] Jay Pujara, Hui Miao, Lise Getoor, and William Cohen. Knowledge graph identification. In *International Semantic Web Conference*, pages 542–557. Springer, 2013.

[PP02] Athanasios Papoulis and S Unnikrishna Pillai. *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

[PPN+14] Adler Perotte, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Noémie Elhadad. Diagnosis code assignment: models and evaluation metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237, 2014.

[PSC+14] Peggy L. Peissig, Vitor Santos Costa, Michael D. Caldwell, Carla Rottscheit, Richard L. Berg, Eneida A. Mendonca, and David Page. Relational machine learning for electronic health record-driven phenotyping. *Journal of Biomedical Informatics*, 52:260–270, 2014.

[QLZ+09] Tao Qin, Ty Tie-Yan Liu, Xd Xu-Dong Zhang, Ds De-Sheng Wang, and Hang Li. Global ranking using continuous conditional random fields.

In *Advances in neural information processing systems*, pages 1281–1288, 2009.

[Ran57] J Rankin. Cerebral vascular accidents in patients over the age of 60. II. Prognosis. *Scott. Med. J.*, 2(5):200–215, 1957.

[RB05] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 860–867. IEEE, 2005.

[RBS+15] Narges Razavian, Saul Blecker, Ann Marie Schmidt, Aaron Smith-McLallen, Somesh Nigam, and David Sontag. Population-Level Prediction of Type 2 Diabetes From Claims Data and Analysis of Risk Factors. *Big Data*, 3(4):277–287, 2015.

[RC95] A Rangarajan and R Chellappa. Markov random field models in image processing. In *The Handbook of Brain Theory and Neural Networks*, pages 564–567. MIT Press, 1995.

[RC14] Mahsa Rouzbahman and Mark Chignell. Predicting ICU Death with Summarized Data: The Emerging Health Data Search Engine. Technical report, University of Toronto, 2014.

[RCR16] Ruben Rodrigues, Hugo Costa, and Miguel Rocha. Development of a Machine Learning Framework for Biomedical Text Mining. In *10th International Conference on Practical Applications of Computational Biology & Bioinformatics*, pages 41–49. Springer, 2016.

[RH05] Havard Rue and Leonhard Held. *Gaussian Markov random fields: theory and applications*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press, 2005.

[RKH+09] Martin A Ritter, Peter Kimmeyer, Peter U Heuschmann, Rainer Dziewas, Ralf Dittrich, Darius G Nabavi, and E Bernd Ringelstein. Blood Pressure Threshold Violations in the First 24 Hours After Admission for Acute Stroke Frequency, Timing, Predictors, and Impact on Clinical Outcome. *Stroke*, 40(2):462–468, 2009.

[RKNP16] Luc De Raedt, Kristian Kersting, Sriraam Natarajan, and David Poole. *Statistical Relational Artificial Intelligence: Logic, Probability, and Computation*. Morgan & Claypool Publishers, 2016.

[RLF09] N L Reaven, J E Lovett, and S E Funk. Brain injury and fever: hospital length of stay and cost outcomes. *J. Intensive Care Med.*, 24(2):131–139, 2009.

[RLM+06] Daniel L Rubin, Suzanna E Lewis, Chris J Mungall, Sima Misra, Monte Westerfield, Michael Ashburner, Ida Sim, Christopher G Chute, Harold Solbrig, Margaret-Anne Storey, and Others. The National Center for Biomedical Ontology: Advancing Biomedicine through Structured. *Info: Lawrence Berkeley National Laboratory*, 2006.

[RMH17] Jesse Read, Luca Martino, and Jaakko Hollmén. Multi-label methods for prediction with sequential data. *Pattern Recognition*, 63:45–55, 2017.

[RRB13] Michael J Rothman, Steven I Rothman, and Joseph Beals. Development and validation of a continuous measure of patient condition using the electronic medical record. *Journal of biomedical informatics*, 46(5):837–848, 2013.

[RS11] Max J Romano and Randall S Stafford. Electronic health records and clinical decision support systems: impact on national ambulatory care quality. *Archives of internal medicine*, 171(10):897–903, 2011.

[Sar06] Sunita Sarawagi. Efficient inference on sequence segmentation models. *Proceedings of the 23rd international conference on ...*, 2006.

[SASS11] Jyoti Soni, Ujma Ansari, Dipesh Sharma, and Sunita Soni. Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction. *International Journal of Computer Applications*, 17(8):43–48, 2011.

[SBH+07] Charles Safran, Meryl Bloomrosen, W Edward Hammond, Steven Labkoff, Suzanne Markel-Fox, Paul C Tang, Don E Detmer, and Others. Toward a national framework for the secondary use of health data: an American Medical Informatics Association White Paper. *Journal of the American Medical Informatics Association*, 14(1):1–9, 2007.

[SC04] Sunita Sarawagi and William W. Cohen. Semi-Markov conditional random fields for information extraction. In *In Advances in Neural Information Processing Systems 17*, volume 17, pages 1185—-1192, 2004.

[SCMD13] Daniela Stojanova, Michelangelo Ceci, Donato Malerba, and Saso Dzeroski. Using PPI network autocorrelation in hierarchical multi-label classification trees for gene function prediction. *BMC bioinformatics*, 14:285, 2013.

[SGM13] Mollie Shulan, Kelly Gao, and Crystal Dea Moore. Predicting 30-day all-cause hospital readmissions. *Health care management science*, 16(2):167–175, 2013.

[She73] S Sherman. Markov random fields and Gibbs random fields. *Israel Journal of Mathematics*, 14(1):92–103, 1973.

[SHL+12] Jimeng Sun, Jianying Hu, Dijun Luo, Marianthi Markatou, Fei Wang, Shahram Edabollahi, Steven E Steinhubl, Zahra Daar, and Walter F Stewart. Combining knowledge and data driven insights for identifying risk factors using electronic health records. In *Annual Symposium proceedings / AMIA Symposium*, volume 2012, pages 901–10. American Medical Informatics Association, 2012.

[Sin12] Amit Singhal. Introducing the knowledge graph: things, not strings. *Official google blog*, 2012.

[SK94] Ali H. Sayed and Thomas Kailath. A State-Space Approach to Adaptive RLS Filtering. *IEEE Signal Processing Magazine*, 11(3):18–60, 1994.

[SKJ84] D J Spiegelhalter and R P Knill-Jones. Statistical and knowledge-based approaches to clinical decision-support systems, with an application in gastroenterology. *J R Stat Soc Ser A (General)*, 147:35–77, 1984.

[SLF+10] Xiaoxiao Shi, Qi Liu, Wei Fan, Qiang Yang, and S Yu Philip. Predictive Modeling with Heterogeneous Sources. In *SDM*, pages 814–825, 2010.

[SLL+16] Elyne Scheurwegs, Kim Luyckx, Leon Luyten, Walter Daelemans, and Tim Van den Bulcke. Data integration of structured and unstructured sources for assigning clinical codes to patient stays. *Journal of the American Medical Informatics Association*, 23(e1):11–19, 2016.

[SLW+15] Peter Schulam, Colin Ligon, Robert Wise, Laura Hummers, Fredrick Wigley, and Suchi Saria. A Framework for Individualized Prognosis of Disease Trajectories in Complex , Chronic Diseases : Application to Scleroderma , an Autoimmune Disease. *AMIA Annual Symposium Proceedings*, pages 143–144, 2015.

[SM07] Charles Sutton and Andrew McCallum. An introduction to conditional random fields for relational learning. In Lise Getoor and Ben Taskar, editors, *Introduction to Statistical Relational Learning*, chapter 4, pages 93–128. MIT Press, 2007, illustrate edition, 2007.

[SM12] Charles Sutton and Andrew McCallum. An Introduction to Conditional Random Fields. *Foundations and Trends in Machine Learning*, 4(4):267–373, 2012.

[SMR07] Charles Sutton, Andrew McCallum, and Khashayar Rohanimanesh. Dynamic Conditional Random Fields: Factorized Probabilistic Models for Labeling and Segmenting Sequence Data. *The Journal of Machine Learning Research*, 8:693—-723, may 2007.

[SP03] Fei Sha and Fernando Pereira. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1 (NAACL '03)*, pages 134–141. Association for Computational Linguistics, 2003.

[SPGP12] Xiaoxiao Shi, Jean-Francois Paiement, David Grangier, and S Yu Philip. Learning from Heterogeneous Sources via Gradient Boosting Consensus. In *SDM*, pages 224–235, 2012.

[Spi71] Frank Spitzer. Markov random fields and Gibbs ensembles. *The American Mathematical Monthly*, 78(2):142–154, 1971.

[SR13] Jimeng Sun and Ck Reddy. Big data analytics for healthcare. *SIAM International conference on Knowledge discovery and data . . .*, page 1525, 2013.

[SRFL+14] Chaitanya Shivade, Preethi Raghavan, Eric Fosler-Lussier, Peter J Embi, Noemie Elhadad, Stephen B Johnson, and Albert M Lai. A review of approaches to identifying patient phenotype cohorts using electronic health records. *Journal of the American Medical Informatics Association*, 21(2):221–230, 2014.

[SRM04] Charles Sutton, Khashayar Rohanimanesh, and Andrew McCallum. Dynamic conditional random fields: factorized probabilistic models for labeling and segmenting sequence data. In *Proceedings of the twenty-first international conference on Machine learning (ICML '04)*, page 99, New York, New York, USA, jul 2004. ACM Press.

[SS13] Chandrima Sarkar and Jaideep Srivastava. Impact of density of lab data in EHR for prediction of potentially preventable events. In *Proceedings - 2013 IEEE International Conference on Healthcare Informatics, ICHI 2013*, pages 529–534, 2013.

[SS15] Peter Schulam and Suchi Saria. A Framework for Individualizing Predictions of Disease Trajectories by Exploiting Multi-Resolution Structure. In *Advances in Neural Information Processing Systems (NIPS)*, pages 748–756, 2015.

[SSP+80] R J Schneider, K Seibert, S Passe, C Little, T Gee, B J Lee Iii, V Mike, and C W Young. Prognostic significance of serum lactate dehydrogenase in malignant lymphoma. *Cancer*, 46:139–143, 1980.

[SWS+00] Arnold W M Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1349–1380, 2000.

[TDF16] Niall Twomey, Tom Diethe, and Peter Flach. On the need for structure modelling in sequence prediction. *Machine Learning*, 104(2):291–314, 2016.

[TDP+16] Vahid Taslimitehrani, Guozhu Dong, Naveen L. Pereira, Maryam Panahiazar, and Jyotishman Pathak. Developing EHR-driven heart failure risk prediction models using CPXR(Log) with the probabilistic loss function. *Journal of Biomedical Informatics*, 60:260–269, 2016.

[TGK04] Ben Taskar, Carlos Guestrin, and Daphne Koller. Max-margin Markov networks. In *Advances in Neural Information Processing Systems 16 - NIPS'03*, volume 16, pages 25–32, 2004.

[TJ02] S Tatikonda and Michael I Jordan. Loopy belief propagation and Gibbs measures. In *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence UAI2002*, volume 18, pages 493–500, 2002.

[TKV10] Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. Mining Multi-label Data. In *Data Mining and Knowledge Discovery Handbook*, chapter 34, pages 667–685. Springer, 2 edition, 2010.

[TSNJ14] Ying P Tabak, Xiaowu Sun, Carlos M Nunez, and Richard S Johannes. Using electronic health record data to develop inpatient mortality predictive model: Acute Laboratory Risk of Mortality Score (ALaRMS).

Journal of the American Medical Informatics Association, 21(3):455–463, 2014.

[Tu96] Jack V Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of clinical epidemiology*, 49(11):1225–1231, 1996.

[Van10] Erik Vanmarcke. *Random fields: analysis and synthesis*. World Scientific, 2010.

[VWB+16] David M. Vock, Julian Wolfson, Sunayan Bandyopadhyay, Gediminas Adomavicius, Paul E. Johnson, Gabriela Vazquez-Benitez, and Patrick J. O'Connor. Adapting machine learning techniques to censored time-to-event health record data: A general-purpose approach using inverse probability of censoring weighting. *Journal of Biomedical Informatics*, 61:119–131, 2016.

[vWEGF10] Carl van Walraven, Gabriel J Escobar, John D Greene, and Alan J Forster. The Kaiser Permanente inpatient risk adjustment methodology was valid in an external patient population. *Journal of clinical epidemiology*, 63(7):798–803, 2010.

[WBK06] Siri Krishan Wasan, Vasudha Bhatnagar, and Harleen Kaur. The impact of data mining techniques on medical diagnostics. *Data Science Journal*, 5(October):119–126, 2006.

[WD09] Daya C Wimalasuriya and Dejing Dou. Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, pages 1–15, 2009.

[Wei00] Yair Weiss. Correctness of local probability in graphical models with loops. *Neural computation*, 12:1–41, 2000.

[WH03] Adam B. Wilcox and George Hripcsak. The role of domain knowledge in automating medical text report classification. *Journal of the American Medical Informatics Association*, 10(4):330–338, 2003.

[WHB+88] Homer R. Warner, Peter Haug, Omar Bouhaddou, Michael Lincoln, Homer Warner Jr, Dean Sorenson, John W. Williamson, and Chinli Fan. ILIAD as an expert consultant to teach differential diagnosis. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pages 371–376. American Medical Informatics Association, 1988.

[WM85] George Y Wong and William M Mason. The hierarchical logistic regression model for multilevel analysis. *Journal of the American Statistical Association*, 80(391):513–524, 1985.

[WMD08] Michael Wick, Andrew Mccallum, and Anhai Doan. A Discriminative Approach to Ontology Mapping. In *The International Workshop on New Trends in Information Integration, NTII' 08*, pages 16–19, 2008.

[Won] Andrew Wong. Natural History and Determinants of Changes in Physiological Variables after Ischaemic Stroke.

[WPBPM13] Anna Wróblewska, Grzegorz Protaziuk, Robert Bembenik, and Teresa Podsiadły-Marczykowska. Associations between Texts and Ontology. In Robert Bembenik, Lukasz Skonieczny, Henryk Rybinski, Marzena Kryszkiewicz, and Marek Niezgodka, editors, *Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions*, pages 305–321. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[WRS10] Jionglin Wu, Jason Roy, and Walter F. Stewart. Prediction modeling using EHR data: challenges, strategies, and a comparison of machine learning approaches. *Medical Care*, 48(6):106–113, 2010.

[WS08] A Wright and D F Sittig. A four-phase model of the evolution of clinical decision support architectures. *Int J Med Inform*, 77:641–649, 2008.

[WSW14] Xiang Wang, David Sontag, and Fei Wang. Unsupervised Learning of Disease Progression Models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 85—-94, 2014.

[WWH14] Xiang Wang, Fei Wang, and Jianying Hu. A multi-task learning framework for joint disease risk prediction and comorbidity discovery. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 220–225. IEEE, 2014.

[WZFC14a] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge Graph and Text Jointly Embedding. In *EMNLP*, pages 1591–1601. Citeseer, 2014.

[WZFC14b] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge Graph Embedding by Translating on Hyperplanes. In *AAAI Conference on Artificial Intelligence*, pages 1112–1119. Citeseer, 2014.

[YFRJ07] Yisong Yue, Thomas Finley, Filip Radlinski, and Thorsten Joachims. A support vector method for optimizing average precision. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 271–278. ACM, 2007.

[YIN+13] Kazuto Yamashita, Hiroshi Ikai, Masaji Nishimura, Kiyohide Fushimi, Yuichi Imanaka, and Others. Effect of certified training facilities for intensive care specialists on mortality in Japan. *Critical Care and Resuscitation*, 15(1):28, 2013.

[YK08] M Yong and M Kaste. Association of characteristics of blood pressure profiles and stroke outcomes in the {ECASS-II} trial. *Stroke*, 39(2):366–372, 2008.

[YK09] Lexiang Ye and Eamonn Keogh. Time series shapelets: a new primitive for data mining. In *... conference on Knowledge discovery and data mining*, pages 947–956. ACM, 2009.

[YLCW09] Nan Ye, WS Lee, HL Chieu, and Dan Wu. Conditional random fields with high-order features for sequence labeling. In *Advances in Neural Information ...*, 2009.

[YQF+03] Qing-Hai Ye, Lun-Xiu Qin, Marshonna Forgues, Ping He, Jin Woo Kim, Amy C Peng, Richard Simon, Yan Li, Ana I Robles, Yidong Chen, Zeng-Chen Ma, Zhi-Quan Wu, Sheng-Long Ye, Yin-Kun Liu, Zhao-You Tang, and Xin Wei Wang. Predicting hepatitis B virus-positive metastatic hepatocellular carcinomas using gene expression profiling and supervised machine learning. *Nature medicine*, 9(4):416–423, 2003.

[ZBS01] Yongyue Zhang, Michael Brady, and Stephen Smith. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *Medical Imaging, IEEE Transactions on*, 20(1):45–57, 2001.

[ZC12] Yimeng Zhang and Tsuhan Chen. Efficient Inference for Fully Connected CRFs with Stationarity. *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[Zha02] Lei Zhang. *Knowledge graph theory and structural parsing*. PhD thesis, Twente University, 2002.

[ZLNY12] Jiayu Zhou, Jun Liu, Vaibhav A. Narayan, and Jieping Ye. Modeling Disease Progression via Fused Sparse Group Lasso. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1095—-1103, 2012.

[ZLNY13] Jiayu Zhou, Jun Liu, Vaibhav A. Narayan, and Jieping Ye. Modeling disease progression via multi-task learning. *NeuroImage*, 78:233–248, 2013.

[ZN09] Geoffrey Zweig and Patrick Nguyen. A segmental CRF approach to large vocabulary continuous speech recognition. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pages 152–157. IEEE, 2009.

[ZW11] Di Zhao and Chunhua Weng. Combining PubMed knowledge and EHR data to develop a weighted bayesian network for pancreatic cancer prediction. *Journal of Biomedical Informatics*, 44(5):859–868, 2011.

[ZXYP13] Qing Zhang, Yang Xie, Pengjie Ye, and Chaoyi Pang. Acute ischaemic stroke prediction from physiological time series patterns. *The Australasian medical journal*, 6(5):280–6, jan 2013.

[ZZ14] Min Ling Zhang and Zhi Hua Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.