UNIVERSITY OF CALIFORNIA
SANTA CRUZ

LEARNING STRUCTURED AND CAUSAL PROBABILISTIC MODELS FOR COMPUTATIONAL SCIENCE

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Dhanya Sridhar

September 2018

The Dissertation of Dhanya Sridhar
is approved:

Lise Getoor, Chair

Marilyn Walker

Kristian Kersting

Lori Kletzer
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Learning Structured and Causal Probabilistic Models for Computational Science

by

Dhanya Sridhar

The drive to understand human phenomena such as our behavior and biology guides scientific discovery in the social and biological sciences. Today's wealth of observational and experimental data presents both opportunities and challenges for machine learning methods to facilitate these discoveries around human behavior and biology. Social media sites provide observational data, capturing snapshots of how users feel towards current events, engage in discourse with one another, and reflect on behavioral factors that affect their mood. These rich textual data support socio-behavioral modeling and understanding. In biology, large-scale experimental datasets are available, coupled with extensive efforts to extract and curate scientific ontologies and knowledge bases. Such empirical data enables inferences in pharmaceutical sciences and genetics. While standard machine learning methods build probabilistic models using social media posts or gene expression levels, they fall short on handling three important challenges in these problems. First, in socio-behavioral and biological domains, inferences are interrelated and require collective reasoning. Second, prior knowledge from multiple sources such as textual or experiment evidence are abundant and probabilistic methods must fuse these signals of varying fidelity. Third, to advance discoveries in social and biological sciences, computational methods must go beyond predictive performance. In both domains, experts seek new insights and knowledge, requiring techniques to discover patterns and causal relationships directly from data.

My dissertation addresses the challenges of computational science domains by developing a unified probabilistic framework that: 1) exploits useful structure in

the domain to make collective inferences; 2) fuses several sources of signals; 2) discovers causal structure; 4) enables learning of complex, structured models directly from data. I validate this framework on important scientific modeling problems such as online debate and dialogue, mood and behavioral choices, interactions between drug treatments, and gene regulation. In this thesis, I first develop structural patterns for collective inference by evaluating several modeling choices for online debates. My findings illustrate the harms of naïve collective reasoning while showing the benefits of jointly modeling debate interactions and users' stances. I extend these collective patterns to fuse several sources of biological information which lead to state-of-the-art performance in drug-drug interaction prediction. To go beyond predictive performance, I combine multiple statistical signals to infer causal networks of gene regulation from measurements of gene expression and estimate causal effects in dialogue. Finally, I develop algorithms that learn these modeling patterns directly from data, showing the benefits of discovering complex dependencies in the drug-drug interaction prediction domain. The technical contributions highlighted in my thesis lay the foundation for applying structured and causal models to computational science. I conclude by outlining promising areas of future research that stem from my work and further bolster probabilistic methods for scientific domains.

To my parents and in memory of my grandmother

Acknowledgments

Most of this thesis reflects the research from my years as a student; I hope that these few pages offer a reflection of the people that made those years not only possible but enjoyable. For the culmination of these past several years, I owe a debt of gratitude to people from all facets of my life.

My advisor Lise Getoor deserves much of the credit and a continuing thanks for her guidance and mentorship. I began as a rambling grad student with little research know-how and a proclivity for finding tenuous at best connections between ideas. Lise helped me channel that energy to pick interesting problems, carry out careful research, and forge connections with others about my work. Lise's knack for presenting and framing research, both in talks and technical writing, are among the most important lessons I've gleaned. I have only just begun to scratch the surface of academia, but Lise's brilliance, creativity, and dedication to her students is something to aspire towards. I cannot thank Lise enough for her support, encouragement, and thoughtful advice.

I'd like to thank my committee members Marilyn Walker and Kristian Kersting, whose feedback and insight on much of my work has been key for this thesis. Lyn is a collaborator whose sharpness and attention to detail has been instrumental throughout. I've always been in awe of Kristian's brilliance, and he has been kind enough to accommodate my deluge of questions and keenness to discuss ideas at conferences. I thank both of them immensely and hope to stay in touch.

It's another testament to Lise that our group, LINQS, are not just my lab mates but close friends. I have had the pleasure of working with several generations of talented and wonderful LINQS members and postdocs. When I began as a completely inexperienced student, then senior students Ben London, Jay Pujara, Steve Bach and our postdoc Bert Huang gently helped me along, obliged

to emergency Skype calls with a distressed me on the other end, and encouraged me when I was plagued with doubts. This theme of support continued as Jimmy Foulds joined our group, and sparked one of our key lines of research with his perceptiveness. Jimmy's expertise was critical in helping me sharpen my technical foundation, and his good-natured guidance and friendship made our work together fun and productive. After Jay continued on as a postdoc, the research and tutorials we developed together were some of the most enjoyable. I'm especially grateful to Jay's guidance throughout as he become a mentor for me. The discipline and research skills I've learned from working with him are invaluable. He deserves a special thanks for fielding my barging into his office to chat and patiently editing out my unnecessarily flowery wording even at 2 AM before a deadline. Finally, Golnoosh Farnadi has continued this legacy of fantastic postdocs that have brainstormed and spent several late nights to conquer research beasts with me. I also thank former and current LINQS members Shobeir Fakhraei, Arti Ramesh, Shachi Kumar, Varun Embar, Sriram Srinivasan, and Eriq Augustine for their friendship and collaboration. I hope the current and future LINQS members enjoy our group as much as I did. Our LINQS crew reunions at conferences are already some of my favorite moments, and I look forward to many more.

Alongside me during my time in LINQS, I made three amazing friends that I'd like to especially thank: Sabina Tomkins, Golnoosh Farnadi and Pigi Kouki. I admire these women for their work ethic, technical prowess, but especially for their resilience in handling any challenge that came their way. Pigi, Sabina and Golnoosh became my close friends that often restored my sanity, buoyed me through my low points and celebrated with me in my high moments. Time spent with Sabina and Golnoosh were among the highlights of my years in Santa Cruz. I cannot wait to see the great things that each of them will invariably achieve, and

I will look forward to our reunions in various places across the globe.

My friends both in and outside of Santa Cruz have succeeded in ensuring that I balanced my research with phone calls, spirited volleyball matches and outdoor adventures. My biggest thanks to go some of my oldest and unwavering friends: Jillian Kermani, Rachel Miura, Dani and Gabi Drummond, Lauren Crawford, and Devan Tracy. Staying in close touch with these friends always gives me lots to look forward to, and I'm eager for many more adventures. Within Santa Cruz, I enjoyed reprieve from research with my board games crew: Sean Smith, Jay Pujara, Golnoosh Farnadi, Peter Cotrell; my running, hiking and climbing buddies: Soja-Marie Morgens, Andrew Wills, Aaron Springer, Emily Lovell, Jay Pujara, Sabina Tomkins; machine learning volleyball group: Holakou Rahmanian, Michal Derezinski, Sabina Tomkins, Params Raman, Sriram Srinivasan; and Sara Nasab. I especially thank Emily, Soja, Sabina and Golnoosh for your constant friendship during these years.

I had basically an extension of my own family that lent an immense amount of support these past many years. Mary Crawford and Roger Chaffin gave me not just a home that became a sanctuary, but also a family here in Santa Cruz. Words cannot thank them enough and express what they mean to me. Phil and Irene Whitfield were constant cheerleaders, and I have no doubt they will continue to be. My biggest thanks for pulling through these years of grad school with me are for my partner and teammate, Shawn Whitfield whose support, humor and infinite patience enrich my life daily.

Finally, and without a doubt the most enduring gratitude is for my family. I thank my brother, Dilip, for his good humor, lightheartedness and sage insights, making time for me when his own collegiate years keep him busy. I owe everything to my parents, Pavithra and Sridhar. They instilled in me discipline and

curiosity; they have provided me with each opportunity that eventually brought me to grad school. Their unshakable confidence that I would persevere in the face of challenges often keeps me going. In my years of grad school, their constant flexibility, adaptability and gracious support, even when I didn't realize I needed it, will be some of my most profound memories of these years. Their love means everything to me. This work is in memory of my grandmother, my first best friend and role model for what it means to be a strong, sharp and self-assured woman. She was one of my biggest cheerleaders and supporters throughout my PhD, and her memory motivates me daily.

I appreciate the generous President's dissertation year fellowship which funded my final year. Portions of this work were supported by the National Science Foundation (NSF), under grant numbers CCF-1740850, IIS1218488, and IIS-1703331; by the Intelligence Advanced Research Projects Activity (IARPA), via Department of Interior National Business Center (DoI/NBC) contract number D12PC00337; and by the Defense Advanced Research Projects Agency (DARPA). The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF, IARPA, DoI/NBC, DARPA, or the U.S. Government.

Chapter 1

Introduction

The drive to understand human life underscores much scientific research in disciplines such as social science and systems biology. These fields shed light on socio-behavioral and biological phenomena, producing a wealth of experimental and observational data. This abundance of data supports probabilistic models which learn meaningful associations between variables to infer new facts, facilitating further scientific study. However, inference problems in socio-behavioral and biological settings, which I denote *computational science*, challenge the standard assumptions of probabilistic methods, requiring new modeling paradigms and frameworks. In this thesis, I address problems in sociological and biological settings to formalize the challenges of computational science, identifying three desiderata for probabilistic methods. Building on a powerful class of structured probabilistic models, I develop a unified framework to support these needs, addressing real problems in socio-behavioral and biological domains.

Computational science tasks in both socio-behavioral and biological domains are supported by data that are relational, representing rich interdependencies between entities of interests such as users or genes. For socio-behavioral settings, relational data is typically observational, capturing snapshots of behaviors

or interactions. In contrast, biological settings often rely on empirical data from high-throughput experiments and their findings. User behavior data can be gathered at scale from online sources such as social media sites and mood or activity tracking platforms. Social media sites such as Twitter, Facebook or Reddit enables users to reflect on their opinions while discussing and debating with one another. Representing these interactions between users is inherently relational. Mood and activity logging applications allow users to monitor their daily choices both through measurements and textual entries [84, 58, 138]. In these platforms, dependencies across time capture relational structure. In these socio-behavioral domains where direct experimentation is prohibitive, probabilistic models play an important role in using these observational data to shed light on behaviors and attitudes. In this thesis, I focus on multiple social science tasks and settings such as inferring users' stances towards current political issues, attitudes between users, and links between users' physical and mental well-being.

In contrast to observational user behavior data, in biological domains, experimental results from large-scale assays or screens combined with highly curated databases of existing knowledge provide empirical data for computational science problems. For example, assays that produce gene expression measurements support inferences of gene regulatory patterns, and known interactions between drug treatments combined with measurements of their molecular structure similarity inform the prediction of novel drug-drug interactions. Again, links between entities such as drugs and genes induce relational structure that is important in inference tasks. In biological domains, probabilistic models disentangle meaningful patterns and associations from varied, heterogeneous experimental data. I validate the approaches developed in this thesis on two different biological tasks of inferring interactions between drug treatments and regulatory networks of genes.

In both of these prototypical computational science domains, standard probabilistic approaches to classification or regression problems have been proposed. For example, logistic regression and support vector machines classify users' stances on topics that they debate on online forums [157]. Similarly, using measurements of similarity between drug treatments from molecular structure or known side-effects, logistic regression ranks probable interactions between drugs [55]. However, to address computational science problems more broadly, standard probabilistic models still face several shortcomings and require additional assumptions. I formalize the challenges of computational science below and outline my technical contributions to meet the needs of socio-behavioral and biological modeling problems.

1.1 Challenges in Computational Science

The standard probabilistic modeling pipeline proceeds as follows: 1) specifying assumptions about the data and functional form of the model; 2) fitting model parameters with available data; and 3) making inferences on unseen data. In this pipeline, we typically further assume that the model structure is known and the inferences elucidate correlations and not causation, which requires stronger assumptions. Along each of these dimensions, computational science problems necessitate sophisticated choices, culminating in three desiderata:

Interrelated Inferences. Inferences in scientific domains are typically interrelated, e.g. the most probable stance of a social media user toward a political issue depends on probable stances for others with whom the user debates. When predicting interactions between drug treatments, similar drugs are likely to have common sets of interactions. Traditionally, modeling methods assume that given features *local* to a entity such as text from users' posts or cell response measure-

ments for drugs, inferences such as political views or interactions are independent, neglecting the signals from related predictions. In contrast, for computational science tasks, it is important to lift these limiting independence assumptions and reason with not only local features but also over the inferences for related variables such as the stance of another user. These are referred to as *collective* modeling assumptions, and support the interrelated inference imperative to socio-behavioral and biological settings.

Multiple Sources of Information. Training data and observations for computational science problems most often combine several sources of evidence with varying fidelity for the task. As an example, to infer regulatory links between genes, we not only have assays from a single experiment, but from multiple, potentially overlapping experiments. Additionally, curated knowledge bases provide information on taxonomy categories associated with genes or known protein-protein interactions that cover a subset of genes. In social media settings, to understand users' political biases, text data from posts can be combined with multiple signals such as liking, sharing, commenting between posts, and use of hash-tags or political slogans. Each signal varies in reliability and coverage, explaining different regions of the inference space. Fusing several sources of information when specifying and training probabilistic models is critical in scientific domains.

Discovery of Causal Knowledge and Complex Patterns. For tasks such as predicting new drug interactions or identifying the polarity of online discussions, a probabilistic model trained with informative signals for the task suffices to make inferences and find associations between variables. However, for several computational science problems, expert domain knowledge is limited and the goal of analysis is to discover complex patterns and causal relationships. Cause-and-

effect relations lie in contrast to standard statistical associations. For example, a probabilistic model trained on social media posts to recognize opinions on topics might exploit the association that using the phrase "second amendment" correlates to taking an anti stance on the gun control issue. However, the phrase usage does not cause a person to take a particular view. Causation is a stronger statement which indicates that a change in the cause always changes the outcome. In both socio-behavioral and biological tasks, supporting causal reasoning is crucial for true scientific discovery. For example, using gene expression measurements, biologists need to know which genes cause other genes to change their expression levels to develop new disease treatments. To support causal inferences, probabilistic methods require both different modeling assumptions and reasoning frameworks.

Another example of adapting probabilistic methods for discovery is learning complex patterns which inform model structure directly from data. Consider a complex database of relationships between drug therapies, their gene targets, enzymes, and transporters. To effectively model interactions between drugs, we require an understanding of the underlying patterns and rules that govern the relationships between these entities. Such long-range patterns are difficult to discern, even for domain experts, and discovering these structural dependencies enables proper specification of the model. Causal and structure discovery tasks both require different assumptions and probabilistic objectives that conventional modeling frameworks overlook.

These desiderata drive the need for methods that can model complex interdependencies, fuse signals and support discovery. Computational science domains require more flexible approaches than standard probabilistic models such as regression- or factorization-based methods that assume strict independences and homogeneity in the training data. In the next section, I first highlight a connection

between the structure in computational science data and a class of probabilistic models that are well-suited to capture structured representations. I review these structured approaches that lay the foundation for modeling computational science domains, and introduce my technical contributions for applying these models to real scientific problems.

1.2 Structured Probabilistic Approaches

Using an accurate representation of the data can often guide the selection of a modeling approach. The standard data representation input to the modeling pipeline is a flat table of entities and their attributes. For example, when predicting the stance of users towards a topic from social media posts, the rows of this table correspond to users and its columns include counts of each word that appears in the entire post corpus. This choice refers to the standard bag-of-words representation often used to predict characteristics from textual data. In contrast, for computational science domains, a useful abstraction is a complex graph which I refer to as *relational data graph*. Relational representations have been long studied across multiple research areas that span databases to formal logic. Relational models of data include both attributes of potentially multiple types of entities and relationships between them. These models can be compactly represented both with multiple tables, as in a relational database, and with graphs, which is the view I emphasize and use in this thesis.

A relational data graph $\mathcal{G}_{\mathcal{R}}$ is defined by: 1) vertex set $\mathbf{V} = \{V_1 \cup \ldots \cup V_K\}$ which contains K subsets that represent different types of vertices; and 2) edge set $\mathbf{E} = \{E_1 \cup \ldots \cup E_L\}$ that consists of L edge type subsets. As an example, in the social media domain, $\mathcal{G}_{\mathcal{R}}$ contains two types of vertices, for users and their posts, and the edge types might represent two kinds of interactions: liking and replying

to posts. In pharmacological settings, the vertices of $\mathcal{G}_{\mathcal{R}}$ are drug treatments, gene targets, and enzymes; the edge types may include interactions between drugs, similarities between genes, or interactions between enzymes and drugs. In general, E_i can represent hyperedges of the form $(v_1, \ldots, v_k)$ that connect k vertices. For simplicity, I often refer to binary edges between two vertices.

Each edge type subset $E_i = \{(v_j, v_k) | v_j \in V_D, v_k \in V_R\}$ has different vertex types as its domain and range. In our running social media example, users like posts while users may reply to other users. These result in two edge types where the "likes" relation has vertices of type user as its domain and those of type post as its range, while the "reply" relation is between two user vertices. Additionally, each vertex $v_i \in V_j$ of type j has a set of attributes, or annotations, which is denoted $v_i.x_k$ for all attributes $x_k \in X_j$ valid for type j. In our social media relational data graph, attributes for user vertices might include age, demographic information, or political views. Each of these user attributes are variables whose values depend on the type of attribute. For example, political views may be represented as categorical variables while age is continuous.

Similarly, an edge $e_i = (v_j, v_k)$ of type m can also take discrete, binary or continuous values to encode the strength or affinity of the tie between v_j and v_k. For example, in the pharmacological data graph, edges that capture similarity between two genes can have continuous values that represent the degree of closeness. Many useful measures such as the cosine similarity between two gene DNA sequences naturally output continuous values. In the social media setting, the reply edges between posts may have discrete values {agree, disagree, neutral} which indicate the agreement polarity of the interaction. An important advantage of the relational data graph representation is that the annotations, edge values and structure encode important dependencies in the domain. For example, the likes

edges in the social network domain indicate probabilistic dependencies between the corresponding users' stance attributes. The structure of $\mathcal{G}_{\mathcal{R}}$ and the information it encodes subsume the flat table representation and motivates a more sophisticated class of probabilistic models.

The first requirement for modeling relational data graphs is capturing probabilistic dependencies between variables. This need for joint models points to the family of probabilistic graphical models (PGM). A PGM is defined by an undirected or directed graph $G = (V, E)$ with variables V and edges E that encode statistical dependencies between these variables. For example, in the biological setting, a directed edge $E = (V_i, V_j)$ between genes V_i and V_j indicates that changing the expression levels of gene V_i affects the expression levels of gene V_j. The graph structure encodes a joint distribution that allows each variable to influence the value of other variables in inference. In the graphical model that captures influences between genes, finding the most probable expression level assignment to all gene variables requires considering the dependencies between genes to find a parsimonious set of values. Thus, the joint distribution viewpoint of PGMs suits our first criteria for modeling relational data graphs, which requires representing probabilistic dependencies between variables.

Although the probabilistic semantics of PGMs are useful, relational data graphs require a richer modeling representation beyond $G = (V, E)$ to capture repeated substructures in $\mathcal{G}_{\mathcal{R}}$ such as the same-stance views shared by users who like one another's posts. Representing this pattern separately for each individual user and post increases the dimensionality of G. Moreover, since $\mathcal{G}_{\mathcal{R}}$ is defined in terms of relational semantics, a graphical model whose structure can be directly defined with a relational language would ease the difficulty of modeling these complex data. To overcome these limitation, the class of statistical relational

learning (SRL) methods combine richer representations such as first-order logic or relational models with PGMs [75, 83, 150, 7, 126, 73]. SRL approaches specify the underlying PGM structure with weighted logical clauses or relational constraints that can be instantiated from data graphs $\mathcal{G}_{\mathcal{R}}$, capturing repeated patterns and substructures. Recently introduced and popular SRL frameworks such as Markov logic [126], Bayesian logic programs [75], and probabilistic soft logic [7] use weighted logical clauses to specify models, and have been successfully applied to domains from information extraction to natural language processing [12, 118]. First-order logic provides a powerful language for describing models of relational data graphs with clauses that constrain assignments to attributes and edge values of interest. For example, in our social media example, first-order logical clauses can encode preferences for same political view labels for users that like one another's posts. Moreover, this first-order clause is invoked for all such pairs of users that are incident on a "likes" edge, capturing the recurring pattern.

SRL methods thus provide a promising modeling framework for the relational data graphs in computational science domains. However, the goal is to learn these SRL models from training data and use them to infer attributes such as political views or interactions between drug treatments. The expressivity of SRL methods comes with a computational cost for both inference and learning, which are typically NP-hard in arbitrary and cyclic PGMs. One exception is probabilistic soft logic (PSL), which circumvents otherwise NP-hard inference by applying continuous relaxations that admit polynomial-time, efficient inference [7]. This scalability combined with its expressivity makes PSL an attractive candidate for extending and developing a unified computational science framework. In the next section, I detail the contributions of this thesis in fulfilling the desiderata of computational science.

1.3 Contributions

As motivated in the previous section, the PSL framework for structured probabilistic models is a promising step towards meeting the desiderata of computational science. However, given relational data graphs for complex biological or socio-behavioral problems, we still require a unifying framework which builds on PSL to fully address the challenges of these domains. My thesis develops this computational science framework based on PSL by making four foundational contributions:

1. General patterns for modeling the structure common in computational science domains;
2. Methods for fusing multiple sources of information with collective reasoning;
3. New constraints that support the assumptions of causal inference and discovery;
4. Algorithms for learning complex model structure directly from data.

I devote a chapter to each of these overarching tasks, demonstrating the proposed techniques on real biological and socio-behavioral tasks. To address social science settings, I study tasks such as modeling debate and dialogue on online forums, identifying the causal effects of exercise on mood, and understanding the impact of dialogue styles on user sentiment. I also validate my proposed approaches in biological science settings such as predicting drug-drug interactions, inferring gene regulatory networks, and learning biological patterns of drug interaction. The technical contributions of my thesis culminate in a unified framework for computational science problems. This thesis is organized as follows:

In Chapter 2, I first review probabilistic graphical models and statistical relational learning methods in detail to lay the theoretical foundation of my work. In

this chapter, I formalize the two core problems of probabilistic models: inference and learning. I build on these formalisms in subsequent chapters.

In Chapter 3, I propose extensible modeling patterns for relational graphs from online debate forums to infer users' stances towards topics and issues they discuss. I evaluate the ramifications of several modeling choices and develop a joint PSL approach that combines users' text and discussion patterns to reason about both user stance and user-user edges. The joint approach provides a useful template for subsequent computational science tasks and outperforms multiple competing methods in predicting online debate forum stance for users. Much of this work is published in Sridhar et al. [141, 140].

Chapter 4 focuses on methods for fusing multiple information sources with collective inference in the multi-relational pharmaceutical domain. I study the task of predicting novel drug-drug interactions and propose a PSL modeling pattern that combines known interactions and multiple similarity signals between drug treatments to propagate information across predictions. I show that this approach achieves state-of-the-art performance and yields predictions that are validated by the literature. These key findings are published in Sridhar et al. [142].

In Chapter 5, I develop a novel framework, CausPSL, that discovers a graph of cause-and-effect relationships given observed measurements of variables of interest. I encode logical characterizations of causal graphs as constraints in PSL, fusing statistical signals, graph structure penalties and domain knowledge from side information. I apply CausPSL to inferring gene regulatory networks and protein signaling pathways and demonstrate the scalability, performance and robustness of the method. Much of this work appears in Sridhar et al. [143], Sridhar and Getoor [139].

Chapter 6 extends the focus on causality by considering a complementary prob-

lem of estimating causal effects between a single outcome and its potential causes. In Chapter 6, motivated by the benefits of fusing data sources, I address causal inference problems in settings with varying degrees of textual data. I consider two socio-behavioral domains: 1) a mood logging application with variable measurements and text entries; and 2) discussions from online forums. First, in the mood logging domain, I evaluate several modeling choices to develop a methodology for improving causal estimation by combining text data. Second, I focus on text alone and analyze the causal effect of linguistic tone on sentiment in online forum discussions. These studies highlight the importance of considering structure and fusing sources in socio-behavioral causal inference problems. The contributions on mood modeling appear in Sridhar et al. [144].

Chapter 7 covers the discovery of PSL model structure directly from data. The task of learning PSL clauses contrasts against Chapter 5, which focuses on discovering the structure of causal graphs. This discovery problem searches over a different space of models which present scalability and formulation challenges. In this chapter, I formalize the task of structure learning for PSL based on prior work for SRL methods. I propose an efficient data-driven structure learning algorithm that exploits relational patterns in the data to discover PSL models for given tasks. I demonstrate the effectiveness of this proposed structure learning approach for drug-drug interaction prediction given a complex relational graph of enzymes, genes, transporters and drugs. Much of this work is published in Embar et al. [43].

Chapter 2

Preliminaries

Chapter 2 provides an in-depth review of the structured approaches highlighted in Chapter 1. First, I describe the formulation of probabilistic graphical models (PGM), which lay the theoretical foundation for the statistical relational learning (SRL) paradigm. Next, I outline SRL methods, focusing on those that use weighted first-order logic. This review provides the groundwork for describing hinge-loss Markov random fields (HL-MRF), a special family of PGMs that applies a particular convex relaxation to logical satisfaction and probabilistic soft logic (PSL), the language for defining these models. These three formalisms underpin the technical contributions of my thesis and will be referenced throughout.

2.1 Probabilistic Graphical Models

PGMs define joint distributions over variables which are parameterized by an undirected or directed graph where edges between variables denote statistical dependence. Formally, we are given a set $X = \{X_1, \ldots, X_n\}$ of random variables. Each variable X_i takes a value x_i from the domain $\mathcal{X}$. $\mathcal{X}$ can be discrete, binary or real-valued, making X_i a categorical, Boolean or continuous variable. The vector

$\mathbf{x} =< x_i, \ldots, x_n >$ denotes the joint assignment where each $X_i = x_i$. The goal is to define a distribution $P(\mathbf{x})$ over the joint assignment to all variables X. A probabilistic graphical model is defined by a graph $G = (X, E)$ whose vertices correspond to the variables X. An edge $e_i \in E$ is of the form (X_i, X_j) and indicates that $P(\mathbf{x})$ should model probabilistic dependencies between X_i and X_j. Consequently, an important property of PGMs is that the graph structure thus entails conditional independences between variables, allowing the joint distribution to be compact without requiring all 2^n dependences. PGMs are characterized by undirected or directed acyclic graphs (DAG). When G is undirected, the resulting PGM is referred to as a Markov random field (MRF); when G is a DAG, the resultant graphical model is a Bayesian network (BN). Below, I describe each of these formalisms and their conditional independence semantics in detail.

2.1.1 Markov Random Fields

Given an undirected graph $G = (X, E)$, and the corresponding set of maximal cliques $C = \{c_i, \ldots, c_M\}$ formed by the edges in E, a MRF defines the joint distribution over $\mathbf{x}$ as:

$$\begin{aligned} P(\mathbf{x}) &= \frac{1}{Z} \prod_{k=1}^{M} \phi_k(\mathbf{x}_k) \\ Z &= \sum_{x_i \in X} \prod_{i=1}^{k} \phi_k(\mathbf{x}_k) \\ \phi_k(\mathbf{x}_k) &= \exp(\lambda_k^T f_k(\mathbf{x}_k)) \end{aligned} \tag{2.1}$$

where $X_k = \{x_j | x_j \in c_k\}$ is the set of all variables that participate in the k-th clique, and vector $\mathbf{x}_k$ is the assignment to X_k variables. Z is a normalization constant, denoted the log partition function, and requires exponentially many sums to compute. Thus, evaluating Z is intractable and in practice, several useful

approximations exist. The final component of an MRF are ϕ_k functions, known as the clique potentials. These clique potentials have the property that $\log(\phi_k)$ is linear and ϕ_k is defined by a vector of feature functions $f_k(X_k)$ and weight vector λ_k. Each function $f_k^i(\mathbf{x_k})$ assigns a real-value in $(0, \infty)$ to $\mathbf{x_k}$ that measures the compatibility of this assignment to the variables X_k. Intuitively, higher scoring assignment configurations are exponentially more probable under the distribution. The set of all weight vectors $\Lambda = \{\lambda_k\}_{k=1}^M$ are the parameters of the MRF.

To understand the independences entailed by the graph G which defines an MRF, we consider $N(X_i)$, the neighbors of variable X_i in G (variables connected to X_i by an edge). The local Markov property of distribution P with respect to G indicates that each X_i is conditionally independent of variables $X \setminus X_i$ given its neighbors $N(X_i)$, denoted $X_i \perp\!\!\!\perp X \setminus X_i | N(X_i)$. In MRFs, the neighbors of X_i, $N(X_i)$, denote its Markov blanket, the set of variables required to render X_i conditionally independent of other variables in the graph. For BNs, their directed counterparts, the graph induces a different factorization of the joint distribution and as a consequence, entails different conditional independences, as I show below.

2.1.2 Bayesian Networks

Given a DAG $G = (X, E)$ and a function $\pi(X_i)$ that maps X_i to its parents in G (variables with edges incoming to X_i), a BN defines the joint distribution over X as:

$$P(\mathbf{x}) = \prod_{i=1}^{n} p(x_i | \pi(X_i)) \tag{2.2}$$

where conditional probabilities $p(x_i|\pi(X_i))$ parameterize the distribution. When variables X are categorical or Boolean, these conditional probabilities can be

represented by tables. If variables X are continuous, each p has a functional form with coefficients which correspond to the parameters of the joint distribution. The BN defined by G fulfills two conditional independence properties: the local and global Markov property. The local Markov property of the BN defined by G is that each $X_i \in X$ is independent of its non-descendants in G conditioned on its parents $\pi(X_i)$. The global Markov property of a distribution defined by G relies on all other conditional independences entailed by G. The independence entailment criteria on the graph is known as d-separation and builds on the notion of blocked paths. Below, I introduce necessary definitions and formalize the global Markov property based on these terms.

Definition 1. A **path** p from X_i to X_j in G is an ordered set of edges defined by the sequence of vertices $Z =< X_u \dots X_v >$ such that $(X_i, X_u), (X_v, X_j) \in E$ and all other contiguous $X_l, X_{l+1} \in Z$ are connected by an edge directed in either direction.

Definition 2. A variable X_w along path p is a **collider** if p consists of two incoming edges into X_w.

Definition 3. A path p between X_i to X_j is **blocked** by a set of variables Z when there exists a variable X_w along p such that: 1) X_w is not a collider and X_w in Z; or 2) X_w is a collider and neither X_w nor its descendants are in Z.

Definition 4. Variables X_i and X_j are **d-separated** by Z in G if and only if every path from X_i to X_j is blocked by Z.

Definition 5. The BN $p(X)$ defined by DAG G satisfies the global Markov property that for all subsets of X: U, V and Z such that U and V are d-separated by Z, U and V are conditionally independent in $p(\mathbf{x})$ given Z.

These definitions, when combined with assumptions defined in Chapter 5, imply constraints on valid DAGs G when given a set of variable observations. BNs and MRFs are probabilistic models that support inference and given training data, require learning. Below, I formalize both of these tasks in the context of PGMs.

2.1.3 Inference and Learning

Probabilistic inference is a widely studied problem that includes several forms of queries including marginal inference, which finds $P(x_i)$ from $P(\mathbf{x})$, and conditional inference, which finds $P(X_U|X_E)$ for unknown variables X_U given evidence variables X_E. In this thesis, I focus on the Maximum a Posteriori (MAP) inference problem which finds the mode assignment of $P(\mathbf{x})$. Formally, MAP inference corresponds to the optimization:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} P(\mathbf{x}) \tag{2.3}$$

It is standard to maximize $\log P(\mathbf{x})$, which gives an equivalent solution. In MRFs, the normalization constant can be ignored and $\log \prod_{k=1}^{M} \phi_k(\mathbf{x_k})$ is instead maximized. In general, MAP inference in graphical models is NP-hard. However, for BNs that are structured as trees or chains, there exist efficient and exact dynamic programming algorithms for MAP inference. In MRFs with arbitrary cycles, MAP inference is typically solved with approximate algorithms that find local optima. Before performing MAP inference, the parameters of the PGM are learned from training data, leading to the parameter estimation, or learning, problem.

BNs and MRFs are parameterized by conditional probability distributions and the real-valued weights Λ of feature functions, respectively. For generality, I de-

note any model parameters with set Θ. Here, I focus on a particular class of parameter estimators obtained through maximum likelihood estimation (MLE), which comes with desirable statistical properties. In the context of MLE, the goal of the learning problem is to find the optimal parameters Θ^* so that the log likelihood of observed data is maximized. Formally, we are given n observations of assignments to all the variables $< \mathbf{x}^{(1)} \dots \mathbf{x}^{(n)} >$ and Θ^* is given by:

$$\Theta^* = \arg\max_{\Theta} \sum_{i=1}^{n} \log P(\mathbf{x}^{(i)}) + r(\Theta) \tag{2.4}$$

where the first term is the log likelihood of the observed data and $r(\Theta)$ is a regularization term that prevents over-fitting. An example of $r(\Theta)$ when Θ consists of real-valued weights λ_k is $||\Theta||_2$, the Euclidean or L_2 norm.

The choice of graphical model again affects the complexity of the learning problem. Here, for MRFs, since the log partition function Z depends on the parameters, it cannot be omitted from the optimization. In general, arbitrarily cyclic MRFs, learning the parameters requires approximations to the full log likelihood. Two important approximations are covered in Chapter 7. In BNs, the log likelihood decomposes into a sum over the conditional probability for each variable X_i, admitting exact and often closed-form solutions. Another important learning problem in PGMs is structure learning, which finds the underlying graph G (and the form of the feature functions f_k^i for MRFs) from observed data. Structure learning corresponds to model discovery and plays a critical role in computational science, as highlighted in Chapter 1. This problem will be covered in full detail in subsequent chapters of this thesis.

2.2 Statistical Relational Learning

One challenge in PGMs is defining conditional probability distributions or feature functions. Importantly, relational data graphs typically contain several repeated substructures and specifying a separate feature function for each of these is tedious. Statistical relational learning (SRL) methods overcome this challenge by defining the factors of a PGM with template relational patterns that are instantiated several times by the data. Examples of notable SRL frameworks which define both undirected and directed PGMs include probabilistic relational models [83], relational Markov networks [150], Bayesian logic programs [75], Markov logic networks [126], relational dependency networks [107], probabilistic soft logic [7] and most recently, relational logistic regression [73]. A powerful relational language is first-order logic, and several notable SRL methods such as Markov logic, Bayesian logic programs and probabilistic soft logic rely on weighted first-order logical clauses to define their underlying distributions. While Bayesian logic programs specify directed models, MLNs and PSL yield undirected MRF distributions. In my review of SRL, I discuss logic-based methods for MRFs, as the contributions of my thesis focus on these methods. For an more extensive and comprehensive review of SRL, I refer to the reader to Getoor and Taskar [51]. I first provide an overview of first-order logic before formalizing the MRFs defined by logical clauses.

An **atom** $p(\cdot)$ in first-order logic consists of a predicate p (e.g. WORKS, LIVES) over constants (e.g. `Alice, Bob`) or variables (e.g. A, B). An atom whose predicate arguments are all constants is a *ground* atom. A **literal** is an atom or its negation. A **clause** c is a formula $\wedge_i L_i \vee_j L_j$ where L_i and L_j are literals. Given n clauses $C = \{c_1 \dots c_n\}$ and real-valued weights $\mathbf{w} = \{w_1 \dots w_n\}$, a **model** $M_{C,\mathbf{w}} = \{(w_1, c_1) \dots (w_n, c_n)\}$ is a set of clause and weight pairs.

Given constants from a domain, we substitute the variables appearing in literals over C with these constants to obtain a set of *ground* clauses G_c for each clause $c \in C$. The corresponding set of ground atoms is the set of random variables X which constitute the vertices of graph G. The domain for each X_i is $\{0, 1\}$ since the variables correspond to logical atoms. The model $M_{C,\mathbf{w}}$ defines a distribution over $\mathbf{x}$ as:

$$P_{M_{C,\mathbf{w}}}(\mathbf{x}) = \frac{1}{Z} \exp(-\sum_{i=1}^{n} \sum_{G_{c_i}} w_i \phi_{c_i}(\mathbf{x}))$$
$$\text{where} \tag{2.5}$$
$$Z = \sum_{\mathbf{x}} \exp(-\sum_{i=1}^{n} \sum_{G_{c_i}} w_i \phi_{c_i}(\mathbf{x}))$$

Each ϕ_c instantiated from a clause c is a function over assignments $\mathbf{x}$ that returns 1 if r is satisfied by $\mathbf{x}$ and 0 otherwise. The underlying MRF factorizes over cliques that are induced by ground clauses ϕ_c. A key difference from general MRFs is that the first-order clauses capture repeated patterns by templating several instantiations ϕ_c. Both MAP inference and learning for most logic-templated SRL methods remain computationally intractable and require approximate algorithms such as Gibbs sampling or loopy belief propagation for inference and pseudolikelihood estimation for learning. In the next section, I introduce a particular class of undirected models, HL-MRFs, and PSL, a SRL method for defining these distributions. I show how MAP inference can solved in polynomial time due to the formulation of HL-MRFs, and define pseudolikelihood estimation in the context of HL-MRFs.

2.3 Hinge-loss Markov Random Fields and Probabilistic Soft Logic

In this section, I provide a succinct review of HL-MRFs and the framework for describing them, PSL. I refer the reader to [7] for a full description of these methods. PSL is a SRL framework that defines HL-MRFs, a special class of the undirected graphical model given by Equation 2.5. HL-MRFs are distributions over continuous variables X whose domain is $[0, 1]$. To obtain HL-MRFs from logical clauses, we apply a continuous relaxation of Boolean logic to the ground clauses to derive ϕ_c of the form:

$$\phi_c(\mathbf{x}) = \max\{1 - \sum_{i \in I^+} X_i - \sum_{i \in I^-} (1 - X_i), 0\}^p \tag{2.6}$$

where I^+ and I^- denote the set of non-negated and negated ground atoms in the clause and $p \in \{1, 2\}$. In contrast to ground Boolean clauses that are satisfied or violated (returning 0 or 1), a ground clause in soft logic assigns a continuous distance to satisfaction. The above relaxation of logical satisfaction is convex and follows from applying the Lukasiewicz t-norm to relax logical operators for continuous values. Intuitively, $\phi_c(\mathbf{x})$ corresponds to a linear or quadratic penalty for violating clause c.

PSL defines conditional distributions over the target variables for a particular task conditioned on the remaining evidence variables. Formally, given a set of *target predicates* $\mathbb{P}_T$, a PSL model $\tilde{M_{C,\mathbf{w}}}$ consists of non-negative weights $\mathbf{w} \in \mathbb{R}^+$ and disjunctive clauses $\wedge_i L_i \rightarrow \vee_i T_i$ where the predicate for literal T_i belongs to $\mathbb{P}_T$. In this thesis, the disjunctive clauses of this form are interchangeably referred to as rules. Given target atoms Y and a set of evidence atoms X where each X_i

is observed, a PSL model $\tilde{M_{C,\mathbf{w}}}$ defines an HL-MRF distribution of the form:

$$P_{\tilde{M_{C,\mathbf{w}}}}(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\exp(-\sum_{i=1}^{n}\sum_{G_{c_i}} w_i\phi_{c_i}(\mathbf{x},\mathbf{y}))$$

$$\text{where} \tag{2.7}$$

$$Z = \int_{\mathbf{y}} \exp(-\sum_{i=1}^{n}\sum_{G_{c_i}} w_i\phi_{c_i}(\mathbf{y},\mathbf{y}))$$

Following Equation 2.3, the MAP inference problem for HL-MRFs requires solving:

$$\mathbf{y}^* = \arg\min_{\mathbf{y}}(-\sum_{i=1}^{n}\sum_{G_{c_i}} w_i\phi_{c_i}(\mathbf{x},\mathbf{y})) \tag{2.8}$$

Each continuous hinge-loss penalty function ϕ_{c_i} is piecewise linear in the variables $\mathbf{y}$. Consequently, MAP inference for every HL-MRF is a convex optimization problem which can be solved in polynomial time. This result leads to exact MAP inference algorithms including the scalable consensus-based alternating direction method of multipliers (ADMM) method. Bach [5] unifies convex MAP inference in HL-MRFs with linear programming relaxations for discrete MRFs and randomized algorithms for weighted maximum satisfiability (MAX-SAT) problems. The efficiency of MAP inference makes PSL an attractive modeling choice.

As discussed in Section 2.2, the learning problem remains computationally challenging. To overcome the intractable likelihood score, pseudo-likelihood [13] (PLL) is an approximation that is commonly used across SRL structure learning and weight learning methods. For HL-MRFs, PLL $\hat{P}_{\tilde{M_{C,\mathbf{w}}}}$ approximates the

likelihood as:

$$\hat{P}_{M_{C,\mathbf{w}}^{\sim}}(\mathbf{y}|\mathbf{x}) = \prod_{Y_i \in \mathbf{Y}} \frac{1}{Z_i(\mathbf{y}, \mathbf{x})} \exp(-f_i(y_i, \mathbf{y}, \mathbf{x}))$$
$$\text{where} \tag{2.9}$$
$$Z_i(\mathbf{y}, \mathbf{x}) = \int_{y_i} \exp(-f_i(y_i, \mathbf{y}, \mathbf{y}))$$
$$f_i(y_i, \mathbf{y}, \mathbf{x}) = \sum_{c \in C} \sum_{j : Y_i \in G_c} w_j \phi_j(y_i, \mathbf{y}, \mathbf{x})$$

The notation $j : Y_i \in G_c$ selects ground clauses j where Y_i appears. This notation also corresponds to the Markov blanket of variable Y_i. PLL factorizes the likelihood as a product of local conditional distributions, yielding a log partition function $Z_i(\mathbf{y}, \mathbf{x})$ that requires only evaluating a single integral. Although even a one-dimensional integral can be challenging to compute numerically, it can be efficiently approximated with Monte Carlo methods.

In the subsequent chapters, I introduce PSL modeling patterns for multiple computational science tasks. In Chapter 7 where I formalize structure learning for PSL, I cover both fundamental learning problems in detail.

Chapter 3

Modeling Online Debates

As delineated in Section 1.3, to apply structured PSL models to computational science problems, the first important task is specifying useful modeling patterns that combine the information encoded by relational data graphs. The online debate forums domain introduced in Chapter 1 provides a rich testbed for developing these modeling patterns which can then be extended to other computational science problems, as shown in Chapter 4. In online debate forums, users participate in discussions, or threads, on various topics by writing posts to initiate a discussion, or reply to another user. The text in users' posts indicate both the user's position, or stance, on the topic and the polarity of his/her interaction with other users. The relational data graph for the online debate setting consists of users and their text via posts as vertices, and reply interactions with other users as edges. The reply interactions could include several types of interactions such as agreeing, disagreeing, supporting or opposing. This graph supports several socio-behavioral inference tasks.

With their rich textual data combined with user-user interactions, online debate forums present a valuable opportunity for the understanding and modeling of dialogue. To understand these debates, a key challenge is inferring the users'

stances, all of which are inherently interrelated as described in Chapter 1. Related work in online debate forums have shown the benefits of collective models that rely on structured approaches, but there are several modeling choices whose ramifications are not well understood. To develop general structural patterns, we require investigating these choices carefully and understanding their impacts on collective modeling. This chapter presents a unified framework based on PSL that enables the comprehensive study of collective models that: 1) fuse textual data and interaction information at user (author) or post-level granularity; 2) reason jointly about the polarity of interactions between users; and 3) learn more predictive classifiers using empirical losses that capture imbalances in the training data. We comprehensively evaluate the possible modeling choices on eight topics across two online debate corpora, finding accuracy improvements of up to 11.5 percentage points over a local classifier. The empirical highlights of this chapter emphasize the importance of carefully exploring modeling decisions when developing structured approaches. The modeling templates developed in this chapter inform collective models introduced throughout this thesis.

3.1 Debate Stance Classification

Social media sites such as Twitter, Reddit or Facebook provide a snapshot into users' opinions on a particular topic, broader ideologies, and attitudes toward one another. On such sites, debates emerge as a prominent pattern of discourse and dialogue. Understanding users' positions in these debates and their interactions with one another sheds light on larger political, behavioral and cultural trends. Computational methods that use these online debates to learn users' stances and interactions thus facilitate advances in sociology and political science. This motivates the problem which we refer to as *stance prediction* in online debate forums,

Dialogue Turns	Stance
User 1: 18. That's the smoking age thats the shooting age. Why do you think they call it ATF?	ANTI
User 2: Shooting age? I know 7 year old shooters. 18 should be the gun purchasing age, but there is really no "shooting" age.	ANTI
User 1: I know. I was just pointing out that the logic used to propose a 21 year "shooting age" was inconsistent.	ANTI
User 2: I see. I dont think its really fair that you can join the army at 18 and use handguns and military weapons, but you cant purchase a handgun until 21.	ANTI

Figure 3.1: Example of a debate dialogue turn between two users on the *gun control* topic, from 4FORUMS.COM.

which identifies users' opinion toward the topics they debate and discuss.

Machine learning methods have already been extensively applied to predict users' stance and disagreement between users in online debates [1, 99, 103, 91, 157, 159, 134, 9, 8, 56]. One line of work proposes sophisticated linguistic features that elicit useful signals from users' text alone, without considering the context of entire debate and discussion threads [1, 99, 159, 134]. Several *collective* approaches have improved upon these text-based models by making joint predictions that combine the networks of interactions across users [157, 9, 56, 91]. These collective models exploit both the structure of online debates and domain knowledge of argumentation, but still face several modeling decisions with important ramifications. Below, we emphasize three aspects of debate forum structure that motivate these modeling questions.

In these debate threads, users typically author multiple posts, replying to other users' posts and engaging in back-and-forth discussions. Each post encodes signals about the users' stance and their (dis)agreement towards others. When viewed from the lens of users, these threads are complex, potentially loopy graphs while

from the perspective of posts, the thread represents a tree structure. Typically, approaches for predicting stance from online debates focus on posts, with few methods highlighting preliminary benefits of constraining information at the level of authors [56]. In a separate line of work for stance prediction from congressional debate transcripts, the advantages of author-level modeling have been shown [18]. The first modeling question is whether to specify collective models at the post-level – treating posts as the units of interests – or at the author-level, aggregating post information for each user.

Debates inherently provoke disagreement, with replies between users often displaying negative polarity of sentiment. This domain knowledge underscores many collective approaches which enforce or encourage posts connected by reply interactions to have opposite stance values. However, many socio-political issues are nuanced, presenting multiple facets that lead to more complicated discourse and dialogue than simple disagreements. To properly capture this complexity, the second modeling choice is whether to infer (dis)agreement between users and stance jointly, exploiting the dependencies across these predictions when combining textual information with network structure.

Debates, especially online, are often biased – one side of a topic typically provokes stronger, more polarized reactions than the other. Users arguing for this particular stance are thus more vocal and represented better in the data. Additionally, inherently in debates, interactions between users of opposing stances are more common than between users of the same stance. These biases manifest as imbalance in the labels of training data. When learning probabilistic models in the presence of skewed data, the predictions from the trained model can result in high false positive or false negative rates and harm performance. If policy experts or lawmakers use these predictions in decision-making, these biased models can

have deleterious downstream effects on society. Motivated by data imbalance, the final modeling question examines how we can train probabilistic models to improve predictive performance in the presence of skewed labeled data. In particular, we focus on augmenting learning algorithms with terms that capture the imbalance structure in the data with appropriate loss terms.

In this work, we develop a unified debate stance classification framework for evaluating choices along each modeling dimension: 1) level of aggregating; 2) joint (dis)agreement modeling; 3) mitigating imbalance in learning. We use probabilistic soft logic [7], a declarative approach well-suited to specifying collective models using soft logical constraints. We propose post- and author-level PSL models to investigate the effects of jointly inferring (dis)agreement interactions between users on stance prediction. We introduce a new learning objective for PSL that augments penalty terms for mitigating the imbalance in training labels. We evaluate our framework on several topics two online debate forums, 4Forums and CreateDebate [158, 56]. In addition to empirical improvements of up to 11.5 percentage points of accuracy over simple classification approaches, our technical contributions include:

1. joint modeling framework in PSL that allows us to evaluate all choices
2. novel PSL learning algorithm which augments losses that capture systematic biases in the training data
3. comprehensive validation of all modeling questions across four topics from two debate forum sites

Our extensive experimental results emphasize the importance of aggregating information at the correct level, jointly modeling reply polarity and adjusting for the label imbalances.

3.2 Online Debate Forums

Online debate forums represent richly structured argumentative dialogues. On these forums, users debate with each other in discussion threads on a variety of topics or issues, such as *gun control*, *gay marriage*, and *marijuana legalization*. Each discussion consists of a number of posts, which are short text documents authored by users of the forum. A post is either a reply to a previous post, or it is the start (root) of a thread. As users engage with each other, a thread branches out into a tree of argumentative interactions between the users. Forum users often post numerous times and across multiple discussions and topics, which creates a potentially cyclic interaction graph. Online debates present different challenges than more controlled dialogic settings such as congressional debates. Posts are short and informal, there is limited external information about authors, and debate topics admit many modes of argumentation ranging from serious, to tangential, to sarcastic. The reply graph in online debates also has substantially different semantics to networks in other debate settings, such as the graph of speaker mentions in congressional debates. To illustrate this setting, Fig. 3.1 shows an example dialogue between two users who are debating their opinions on the topic of gun control.

In the context of online debate forums, *stance classification* [151, 133] is the task of assigning stance labels with respect to a discussion topic, either at the level of the user or the level of the post. Stance is typically treated as a binary classification problem, with labels PRO and ANTI. In Fig. 3.1, both users' stances toward gun control are ANTI.

We study datasets from two online debate websites: 4FORUMS.COM, from the Internet Argument Corpus [158], and CREATEDEBATE.COM [56]. Table 3.1 shows statistics about these datasets including the average number of users per discussion

	4Forums	CreateDebate
Users per topic	336	311
Posts per user, per topic	19	4
Words per user, per topic	2511	476
Words per post	134	124
Distinct reply links per user, per topic	6	3
Stance labels given for	Users	Posts
%Post-level reply links have opposite-stance users	71.6	73.9
%Author-level reply links have opposite-stance users	52.0	68.9

Table 3.1: Structural statistics averages for 4FORUMS and CREATEDEBATE.

topic and average number of posts authored. In the online debate forum corpora that we study, the presence of a reply, or even a textual disagreement between posts, does not necessarily indicate opposite stance (e.g. in gun control debates on 4Forums, 23% of disagreements correspond with same stance). These more nuanced debates necessitate richer modeling of replies.

For our unified framework, we specify a hinge-loss Markov random field to reason jointly about stance and reply-link polarity labels. We denote reply-link polarity as (dis)agreement, and obtain labels by considering same- or opposite-stance interactions between pairs of users. Table 3.3 and Table 3.2 show the ratio of positive to negative labels for both stance and disagreement in CREATEDEBATE.COM and 4FORUMS.COM, respectively. We see that in topics such as evolution, gay marriage and marijuana legalization, the PROside outweighs ANTIby up to three times, whereas in discussions around gun control, the ANTIside dominates. In the case of disagreement labels, in topics such as debating Obama on CREATEDEBATE.COM, users largely take opposing views whereas when discussing

Topic	Stance	Disagreement
Abortion	1.6	1.7
Evolution	3.6	0.9
Gay Marriage	3.0	1.3
Gun Control	0.5	1.1

Table 3.2: Ratio of positive to negative stance and disagreement labels in the 4forums dataset.

Topic	Stance	Disagreement
Abortion	1.5	2.3
Marijuana	3.2	1.5
Gay Rights	2.4	3.1
Obama	1.0	3.0

Table 3.3: Ratio of positive to negative stance and disagreement labels in the CreateDebate dataset.

gun control on 4FORUMS.COM, users with matching views interact with one another almost as much. These systematic biases in the training data can result in models that produce higher rates of false positives or false negatives, yielding both poorer classifier performance and imbalance in predictions. This motivates our final contribution around extending the learning algorithm with these appropriate loss functions to mitigate this skew in the results.

3.3 Related Work

Previous work on stance in online debates has shown that contextual information given by reply links is important for predicting stances [157], and that collective classification often outperforms methods which treat each post independently. Hasan and Ng [56] use conditional random fields (CRFs) to encourage opposite stances between sequences of posts, and Walker et al. [159] use MaxCut over explicitly given rebuttal links between posts to separate them into PRO and ANTI clusters. Sridhar et al. [141] use hinge-loss Markov random fields (HL-MRFs)

to encourage consistency between stance and disagreement predictions, evaluating several modeling choices within their framework.

While the first two approaches leverage rebuttal or reply links, they model reply links as being indicative of opposite stances. However, as shown in Fig. 3.1, responses—even rebuttals—can occur between users with the same stance, which suggests the benefit of a more nuanced treatment of reply links. The approach of Sridhar et al. [140] considers text-based agreement annotations between posts, though it requires that reply links are labeled, which can be restrictive. In contrast, Sridhar et al. [141] demonstrate the advantages of jointly inferring uncertain reply label predictions to improve stance classification. Recently, Dong et al. [34] also model both agreement and disagreement in interactions to constraint same- and opposite-stances between users in a generative model applied to predict stances on a news site's comments.

In the context of opinion subgroup discovery, Abu-Jbara and Radev [2] demonstrate the effectiveness of clustering users by opinion-target similarity. In contrast, Murakami and Raymond [103] use simple recurring patterns such as "*that's a good idea*" to categorize reply links as *agree*, *disagree* or *neutral*, prior to using MaxCut for subgroup clustering of comment streams on government websites. This approach improves over a MaxCut approach that casts all reply links as disagreements. Building on this work, Lu et al. [91] model unsupervised discovery of supporting and opposing groups of users for topics in online military forums. They improve upon a MaxCut baseline by formulating a linear program (LP) to combine multiple textual and reply-link signals, suggesting the benefits of jointly modeling textual and reply-link features. Recently, several approaches build on the joint stance constraints proposed by Sridhar et al. [141] to develop weakly supervised learning algorithms for stance prediction [69, 40] when ground truth

is limited or unavailable. Ebrahimi et al. [40] exploit textual similarity to encode relational bootstrapping constraints in HL-MRFs and predict stances when given only a few phrase-level annotations on new topics. Johnson and Goldwasser [69] also use HL-MRFs and include constraints based on temporality and domain knowledge of argument framing, also applying their approach to predicting stances on new and unseen topics.

In a different line of work, while Somasundaran and Wiebe [134] do not use relational information between users or posts, their approach shows the benefit of modeling opinions and their targets at a fine-grained level using relational sentiment analysis techniques. Similarly, Wang and Cardie [160] demonstrate the effectiveness of using sentiment analysis to identify disputes on Wikipedia Talk pages. In the congressional debate setting, approaches using CRFs and similar collective techniques such as minimum-cut have also leveraged reply link polarity for improvements in stance classification [151, 9, 8, 18]. However, these methods rely heavily on features specific to the congressional setting in order to predict link polarity, and make little use of textual features. In contrast, Abbott et al. [1] use a range of linguistic features from the text of posts and their parents to classify agreement or disagreement between posts on the online debate website 4Forums.com, without the goal of classifying stance.

Several approaches have been proposed to train joint classifiers in the presence of imbalanced labels, but have mainly studied problems in link prediction and knowledge base completion [167], named-entity recognition [53] and collaborative filtering [168]. These methods consider training CRFs and Markov logic networks (MLN) [126], templated random field models. Gimpel and Smith [53] first propose a framework for augmenting the pseudolikelihood training loss commonly used for CRFs with a cost function that capture empirical loss. The resulting training

All models:		Collective models only:	
$localPro(X1)$	$\rightarrow\ pro(X1)$	$disagree(X1, X2) \wedge pro(X1)$	$\rightarrow\ \neg\ pro(X2)$
$\neg\ localPro(X1)$	$\rightarrow\ \neg\ pro(X1)$	$disagree(X1, X2) \wedge \neg\ pro(X1)$	$\rightarrow\ pro(X2)$
		$\neg\ disagree(X1, X2) \wedge pro(X1)$	$\rightarrow\ pro(X2)$
		$\neg\ disagree(X1, X2) \wedge \neg\ pro(X1)$	$\rightarrow\ \neg\ pro(X2)$
		$disagree(X1, X2)$	$=\ 1$

Disagreement models only:	
$localDisagree(X1, X2)$	$\rightarrow\ disagree(X1, X2)$
$\neg\ localDisagree(X1, X2)$	$\rightarrow\ \neg\ disagree(X1, X2)$
$pro(X1) \wedge \neg\ pro(X2)$	$\rightarrow\ disagree(X1, X2)$
$pro(X1) \wedge pro(X2)$	$\rightarrow\ \neg\ disagree(X1, X2)$
$\neg\ pro(X1) \wedge \neg\ pro(X2)$	$\rightarrow\ \neg\ disagree(X1, X2)$

Figure 3.2: PSL rules to define the collective classification models, both for post-level and author-level models. Each X is an author or a post, depending on the level of granularity that the model is applied at. The $disagree(X_1, X_2)$ predicates apply to post reply links, and to pairs of authors connected by reply links.

algorithm provides a max-margin solution that trades off likelihood against loss functions that capture different structure in the problem. Yang et al. [167] extend this formulation for MLNs by introducing false-positive (FP) and false-negative (FN) costs that can be tuned for imbalanced link prediction problems, where experts can indicate a strong bias for type one or two error. In our work, we adapt FN and FP penalties for the continuous learning setting in HL-MRFs, motivated by the imbalance in stance and disagreement in online forums.

3.4 Modeling Debate Stance

We face multiple modeling decisions that may impact predictive performance when classifying stance in online debates. A key contribution of this work is the exploration of the ramifications of these choices. We consider the following variations on modeling: collective (**C**) versus local (**L**) classifiers, whether to explicitly model disagreement (**D**), and author-level (**A**) versus post-level (**P**) models. We describe each modeling approach below. We then introduce a novel

cost-penalized learning algorithm that handles the false positive and false negative imbalance in both stance and disagreement labels, for any modeling variant.

Collective versus Local. The first modeling distinction we consider is between classifiers that predict the stance of each user in isolation using only attributes for that user (local) and those that predict all users' stances jointly, exploiting also the stance labels and attributes of other users (collective). In both modeling cases, the content from debate forum posts provide noisy but important local signal about the post and its user's stances. The methods proposed in this work build upon the state-of-the-art local classification approach of Walker et al. [157], which trains a supervised classifier using features including n-grams, lexical category counts, and text lengths. We use logistic regression for our local classifiers which make independent stance predictions. These models will be referred to as *local* (**L**). In *collective* (**C**) classification approaches for stance prediction, the stance labels are all predicted jointly, leveraging relationships along the graph of replies. The simplest way to make use of reply links is to encode that the stance of posts (or authors) that reply to each other is likely to be opposite [159, 56]. Collective approaches attempt to find the most likely joint stance labeling that is consistent with both the local classifier's predictions and the alternation of stance along response threads. The alternating stance assumption is not necessarily a hard constraint, and may potentially be overridden by the local predictions. **C** and **L** models can be constructed with **A** or **P**-level granularity as described below, resulting in four modeling combinations.

Modeling Disagreement. As seen in Fig. 3.1 and Table 3.1, the assumption that reply links correspond to opposite stance is not always correct. This suggests the potential benefit of more nuanced models of agreement and disagreement. A

natural disagreement modeling approach is to predict the polarity of reply links jointly with stance.

There are two variants of reply link polarity to consider. In *textual disagreement*, replying posts are coded as expressing agreement or disagreement with the text of the parent post. This may not correspond to a disagreement in stance *relative to the thread topic*. Some forum interfaces support user self-labeling of post reply links as rebuttals or agreements, thereby explicitly providing textual disagreement labels for posts. Alternatively, in the *stance disagreement* variant, reply links denote either same or opposite *stance* between users (posts). In Fig. 3.1, User 1 and User 2 disagree in text but have the same stance. For collective modeling of stance and disagreement, it is useful to consider the stance disagreement variant which identifies opposite and same-stance reply links, and jointly encourage stance predictions to be consistent with the disagreement predictions.

As with the local classification of stance, we can construct local classifiers for stance disagreement. In this work, for each reply link instance, we use a copy of the local stance classification features for each author/post at the ends of the reply link. The linguistic features further include discourse markers such as “actually" and “because" from the disagreement classifier of Abbott et al. [1]. Additionally, we use textual disagreement as a feature for stance disagreementwhen available. When reply links are not explicitly labeled as rebuttals or agreements, or only rebuttals are known, we instead predict textual disagreement using the features given above, trained on a separate data set with textual-disagreement labels.

Finally, with a stance disagreement classifier in hand, we can build collective models that predict stance based on predicted stance disagreement polarity. We denote these models as *disagreement* (**D**). When applied at one of **A** or **P**-level modeling, this yields two more possible modeling configurations. These models

are certainly more complex than others we consider, but their design is consistent with intuition about the nature of discourse, so the added complexity may yield better accuracy.

Author-Level versus Post-Level. When modeling debates, stance classifiers can predict either the stance of a debate participant (i.e. an *author* (**A**)) [18], or the stance expressed by a specific dialogue act (i.e. a *post* (**P**)) [56]. The choice of prediction target may depend on the downstream goal, such as user modeling or the study of the dialogic expression of disagreement. From a philosophical perspective, authors are individuals who hold opinions, while posts are not. A post is simply a piece of text which may or may not express the opinions of its author.

Nevertheless, given a prediction target, either author or post, it may be beneficial to consider modeling at a different level of granularity. For example, Hasan and Ng [56] find that post-level prediction accuracy can be improved by "clamping" all posts by a given author to the same stance in order to smooth their labels. Alternatively, author-level predictions may potentially be improved by first treating each post separately, thereby effectively giving a classifier more training examples, i.e. the number of *posts* instead of the number of *authors*. With this procedure, a final author-level prediction can be obtained by averaging the predictions over the posts for the author, trading the noisiness of post-level instances against the smoothing afforded by the final aggregation. When designing a stance classifier, the modeler must decide the level of granularity for the prediction target and find the best model therein.

3.5 PSL Models

To study these choices, we build a flexible stance classification framework that implements the above variations using PSL. The models we introduce are specified by the PSL rules in Fig. 3.2, with both post-level and author-level models following the same design. We denote the different modeling choices with the letters defined in Section 3.4. First, local logistic regression classifiers output stance probabilities based on textual features of posts or authors. All of the models begin with these real-valued stance predictions, encoded by the observed predicate *localPro*(X_i). The rules listed for all models encourage the inferred global predictions *pro*(X_i) to match these local predictions.

This defines the *local classification* models **L**, which are HL-MRFs with node potentials and no edge potentials, and which are equivalent to the local classifiers. The collective models extend the **L** models by adding edge potentials which encourage the stance labels to respect disagreement relationships along reply links. Specifically, every reply link between authors (for author-level models) or between posts (for post-level models) x_1 and x_2 is associated with a latent variable *disagree*(x_1, x_2). The rules encourage the global stance variables to respect the polarity of the disagreement variables (same stance, or opposite stance) and while also trying to match the stance classifiers. For the models that do not explicitly model disagreement, it is assumed that every reply edge constitutes a disagreement, i.e. *disagree*$(x_1, x_2) = 1$. These models are denoted **C**.

Otherwise, the disagreement variables are encouraged to match binary-valued predictions from the local disagreement classifiers. We binarize the predictions of the disagreement classifiers to encourage propagation. The disagreement variables are modeled jointly with the stance variables, and label information propagates in both directions between stance and disagreement variables. The full joint

stance/disagreement collective models are denoted **D**. In the following, the models are denoted by pairs of letters according to their collectivity level and modeling granularity. For example, **AC** denotes collective classification performed at the author level, without joint modeling of disagreement. To train these models and use them for prediction, weight learning and MAP inference are performed using the structured perceptron algorithm and ADMM algorithm of Bach et al. [6].

3.6 Cost-Penalized Learning

An important contribution of this work is addressing the training data imbalance in online forum debates. To mitigate this issue, we propose a novel learning algorithm for HL-MRFs which we call cost-penalized maximum pseudolikelihood estimation (CP-MPLE). The CP-MPLE algorithm is a supervised learning method that learns rules weights for a proposed PSL model and extends cost-penalized learning in MLNs [167]. The key idea of our formulation is to augment standard learning with soft FP and FN costs based on continuous variables which can be tuned to balance the effect of positive or negative label skew. Formally, the CP-MPLE estimate optimizes a cost-augmented variant of log pseudolikelihood under an HL-MRF distribution [13, 7]:

$$\begin{aligned}\text{CM-MPLE} &= \sum_{Y_i \in \mathbf{Y}} f_i(y_i, \mathbf{y}, \mathbf{x}) - \log Z_i^{cost} \\ &= \sum_{Y_i \in \mathbf{Y}} f_i(y_i, \mathbf{y}, \mathbf{x}) - \log \int_{y_i'} \exp(-f_i(y_i', \mathbf{y}, \mathbf{x})) \exp(c(y_i', y_i)) \\ &\text{where} \\ f_i(y_i, \mathbf{y}, \mathbf{x}) &= \sum_{c \in C} \sum_{j : Y_i \in G_c} w_j \phi_j(y_i, \mathbf{y}, \mathbf{x})\end{aligned} \tag{3.1}$$

and

$$c(y_i\prime, y_i) = \begin{cases} c(y_i\prime, y_i) = \alpha(y_i' - y_i) & \text{if } y_i' > y_i \\ \beta(y_i - y_i') & \text{if } y_i' < y_i \end{cases}$$

Each normalization constant term Z_i^{cost} marginalizes out Y_i in each conditional distribution $P(Y_i|\mathbf{Y}, \mathbf{X})$, integrating over possible continuous assignments to Y_i, denoted y_i'. However, unlike standard pseudolikelihood, each Z_i^{cost} is multiplied by a cost $c(y_i', y_i)$ which is e^{α} and e^{β} depending on whether a possible assignment y_i' is a false positive or false negative. In contrast to the discrete formulation proposed by Yang et al. [167], based on continuous values y_i', we define soft variants of false positives and false negative assignments. Costs α and β are then multiplied by these degrees of FP and FN violation rather than using indicator functions. When FP cost $e^{\alpha} < 0$, false positive assignments to Y are penalized less, and if FN cost $e^{\beta} < 0$, false negative assignments contribute less error when adjusting the weights during training.

We follow Bach et al. [7] and perform gradient descent with the voted perceptron algorithm. The resulting weight updates are of the form:

$$\begin{aligned} \nabla_{w_c} &= \sum_{Y_i \in \mathbf{Y}} f_i^c(y_i, \mathbf{y}, \mathbf{x}) - \mathbb{E}_{P^{cost}}[f_i^c(y_i, \mathbf{y}, \mathbf{x})] \\ &= \sum_{Y_i \in \mathbf{Y}} f_i^c(y_i, \mathbf{y}, \mathbf{y}) - \int_{y_i'} f_i^c(y_i', \mathbf{y}, \mathbf{y}) \frac{\exp(f_i(y_i, \mathbf{y}, \mathbf{x}) + (c(y_i', y_i))}{\int_{y_i'} \exp(f_i^c(y_i, \mathbf{y}, \mathbf{x}) + c(y_i', y_i))} \end{aligned} \tag{3.2}$$

The expected value of distances to satisfaction of clause c is computed under the modified cost-penalized conditional distribution of $P(Y_i|\mathbf{Y}, \mathbf{X})$. Under the cost-penalized distribution, every possible assignment y_i' is multiplied by its cost, changing the expected value and thus the gradient. Intuitively, α and β modulate the effect that negative or positive examples have on the gradient. For example, if the stances in a topic are skewed towards ANTI, by increasing the penalty on

Models	4Forums			
	Abortion	Evolution	Gay Marriage	Gun Control
PL	61.9 ± 4.3	76.6 ± 3.9	72.0 ± 3.6	66.4 ± 4.6
PC	63.4 ± 5.9	74.6 ± 4.1	73.7 ± 4.3	68.3 ± 5.5
PD	63.0 ± 5.4	76.7 ± 4.2	73.7 ± 4.6	67.9 ± 5.0
AL	64.9 ± 4.2	77.3 ± 2.9	74.5 ± 2.9	67.1 ± 4.5
AC	**66.0 ± 5.0**	74.4 ± 4.2	75.7 ± 5.1	61.5 ± 5.6
AD	**65.8 ± 4.4**	**78.7 ± 3.3**	**77.1 ± 4.4**	67.1 ± 5.4

Table 3.4: Author stance classification accuracy and standard deviation for 4FORUMS, estimated via 5 repeats of 5-fold cross-validation. Bolded figures indicate statistically significant ($\alpha = 0.05$) improvement over AL, the baseline model for the author stance classification task.

Models	CreateDebate			
	Abortion	Gay Rights	Marijuana	Obama
PL	66.4 ± 5.2	70.2 ± 5.0	74.1 ± 6.5	**63.8 ± 8.7**
PC	**68.7 ± 5.7**	**72.6 ± 5.6**	75.4 ± 7.4	**66.1 ± 8.5**
PD	**69.5 ± 5.7**	**73.2 ± 5.9**	74.7 ± 7.0	**66.1 ± 8.5**
AL	65.2 ± 6.5	69.5 ± 4.4	74.0 ± 6.6	59.0 ± 7.5
AC	65.8 ± 7.0	**73.6 ± 3.5**	73.9 ± 7.6	**62.5 ± 8.3**
AD	**67.4 ± 7.5**	**74.0 ± 5.3**	74.8 ± 7.5	**63.0 ± 8.3**

Table 3.5: Author stance classification accuracy and standard deviation for CREATEDEBATE, estimated via 5 repeats of 5-fold cross-validation. Bolded figures indicate statistically significant ($\alpha = 0.05$) improvement over AL, the baseline model for the author stance classification task.

false negatives and decreasing the penalty on false positives, the negative stance examples' influence on the learned model are reduced.

3.7 Experimental Results

The goals of our experimental evaluation are two-fold: 1) performing a comprehensive study of the merits of different modeling choices; and 2) validating both the classifier accuracy and balance benefits of our cost-penalized MPLE learning

Models	4Forums			
	Abortion	Evolution	Gay Marriage	Gun Control
PL	66.1 ± 2.5	72.4 ± 4.2	69.0 ± 2.7	67.8 ± 3.5
PC	**70.5 ± 2.5**	**74.1 ± 3.8**	**73.2 ± 3.1**	**69.1 ± 3.0**
PD	**69.7 ± 2.5**	**73.9 ± 4.0**	**72.5 ± 3.0**	**68.8 ± 3.0**
AL	**74.7 ± 7.1**	73.0 ± 5.7	70.3 ± 6.0	68.7 ± 5.3
AC	**76.8 ± 8.1**	68.3 ± 5.3	72.7 ± 11.1	46.9 ± 8.0
AD	**77.0 ± 8.9**	**80.3 ± 5.5**	**80.5 ± 8.5**	65.4 ± 8.3

Table 3.6: Post stance classification accuracy and standard deviations for 4FORUMS, estimated via 5 repeats of 5-fold cross-validation. Bolded figures indicate statistically significant ($\alpha = 0.05$) improvement over PL, the baseline model for the post stance classification task.

Models	CreateDebate			
	Abortion	Gay Rights	Marijuana	Obama
PL	60.2 ± 3.2	62.7 ± 4.4	68.1 ± 6.1	59.4 ± 6.0
PC	**62.8 ± 3.8**	**66.1 ± 4.9**	68.7 ± 7.9	**61.1 ± 6.6**
PD	**62.6 ± 4.1**	**66.2 ± 5.4**	69.1 ± 7.4	**61.0 ± 6.6**
AL	61.6 ± 9.8	63.7 ± 5.3	66.7 ± 6.7	59.7 ± 13.6
AC	63.4 ± 12.4	**71.2 ± 8.4**	66.9 ± 9.0	63.7 ± 15.6
AD	**66.8 ± 12.2**	**72.7 ± 8.9**	69.0 ± 8.3	63.5 ± 16.3

Table 3.7: Post stance classification accuracy and standard deviations for CREATEDEBATE, estimated via 5 repeats of 5-fold cross-validation. Bolded figures indicate statistically significant ($\alpha = 0.05$) improvement over PL, the baseline model for the post stance classification task.

algorithm. To evaluate the modeling choices outlined in Section 3.4, we study eight topics from 4FORUMS.COM [158] and CREATEDEBATE.COM [56], for classification tasks at both the author level and the post level. Our collective models (**C**) provide a comparison against the CRF approach proposed by Hasan and Ng [56]. We perform extensive cross-validation evaluation to find the best performing modeling choices. We build on this model by examining the effects on both performance and the imbalance in predictions of various learning algorithms, especially our CP-MPLE approach. We select FP and FN cost parameters within

our cross-validation framework to evaluate these merits.

On average, each topic-wise data set contains hundreds of authors and thousands of posts. The 4Forums data sets are annotated for stance at the author level, while CreateDebate has stance labels at the post level. To perform post-level evaluations on 4Forums we apply author labels to the posts of each author, and on CreateDebate we computed author labels by selecting the majority label of their posts. For 4Forums, since post-level stance labels correspond directly to author-level stance labels, we use averages of post-level predictions as the local classifier output for authors. Section 3.2 includes an overview of these debate forum data sets.

In the experiments, we measure classification accuracy across five repeats of 5-fold cross-validation. In each fold, we ran logistic regression using the scikit-learn software package,[1] using the default settings, except for the L1 regularization trade-off parameter C which was tuned on a within-fold hold-out set consisting of 20% of the discussions within the fold. For PSL models, weight learning was performed on the same in-fold tuning sets. We trained via 700 iterations of all learning algorithms, and ran the ADMM MAP inference algorithm to convergence at test time. For CP-MPLE, we tune α and β cost parameters within the same cross-validation framework. We search over all combinations of α from -20 to 20 in intervals of 5.0 and β from -10.0 to 10.0 in intervals of 2.0. These ranges were selected through exploration on a separate development dataset. We hold out each fold in turn and select best performing parameters on the remaining folds. We also perform five repeats of this experiment. When evaluating the impact on classifier performance, we select parameters based on accuracy and when studying imbalance, we optimize for the F_1 score. F_1 score is a natural metric for evaluating the trade-offs between false negatives and false positives

[1] Available at `http://scikit-learn.org/`.

since it captures the balance between precision and recall. On average, weight learning and inference took around 1 minute per fold.

3.7.1 Evaluating Modeling Choices

The full results for author-level and post-level predictions are given in tables 3.4-3.5 and tables 3.6-3.7, respectively. In the tables, entries in bold identify statistically significant differences from the local classifier baseline under a paired t-test with significance level $\alpha = 0.05$. These results are summarized in Fig. 3.3, which shows box plots for the six possible models, computed over the final cross-validated accuracy scores of each of the four data sets from each forum. The overall trends can be seen by reading the box plots in each figure from left to right. In general, collective models outperform local models, and modeling disagreement further improves accuracy. Author-level modeling is typically better than post-level, even for the post-level prediction task. The improvements shown by collective models and author-level models are consistent with Hasan and Ng [56]'s conclusion about the benefits of user-level constraints. This may suggest that posts only provide relatively noisy observations of the underlying author-level stance. Modeling at the author level results in more stable predictions, as noisy posts are pooled together. But here we also show that the full joint disagreement model at the author level, **AD**, performs the best overall, for both prediction tasks and for both forums, gaining up to 11.5 percentage points of post-level accuracy over the local post-level classifier.

A closer analysis reveals some subtleties. When comparing **D** models with **C** models in Fig. 3.3, disagreement modeling makes a much bigger difference at the author level than at the post level. This is likely impacted by the level of class imbalance for *disagreement* classification in the different levels of modeling.

Disagreement, rather than agreement, between authors prompts many responses. Thus, reply links are more likely disagreements when measured at the post level, as seen in Table 3.1. Therefore, enforcing disagreement may be a better assumption at the post level, and the nuanced disagreement model is not necessary in this case. The overall improvements in accuracy from disagreement modeling for post-level models were small.

On the other hand, the assumption that reply edges constitute disagreement is less accurate when modeling at the author level (see Table 3.1). In this case, the full joint disagreement model is necessary to obtain good performance. In an extreme example, the two datasets with the lowest disagreement rates at the author level are evolution (44.4%) and gun control (50.7%) from 4Forums. The **AC** classifier performed very poorly for these data sets, dropping to 46.9% accuracy in one instance, as the "opposite stance" assumption did not hold (Tables 3.4 and 3.6). The full joint disagreement model **AD** performed much better, in fact achieving an outstanding accuracy rates of 80.3% and 80.5% for posts on evolution and gay marriage respectively. To illustrate the benefits of author-level disagreement modeling, Fig. 3.4 shows a post for an author whose stance towards gun control is correctly predicted by **AD** but not the **AC** model, along with a subsequent reply. The authors largely agree with each other's views, which the joint disagreement model leverages, while the simpler collective model encourages opposite stance due to the presence of reply links between them.

3.7.2 Evaluating CP-MPLE

Our first validation of the different modeling choices elucidates the benefits of joint modeling with the **AD** approach. We build on this finding and turn our evaluation to the choice of learning algorithm. As shown in Section 3.2, online

Learning	4Forums			
	Abortion	Evolution	Gay Marriage	Gun Control
MLE	65.8 ± 4.4	78.7 ± 3.3	77.1 ± 4.4	67.1 ± 5.4
MPLE	66.7 ± 4.8	78.8 ± 3.4	77.0 ± 3.6	67.2 ± 5.5
CP-MPLE	**67.0 ± 2.6**	**80.8 ± 1.0**	**78.9 ± 1.6**	**68.3 ± 1.4**

Table 3.8: Accuracy results on **AD** model across four topics of 4FORUMS.COM. CP-MPLE improves performance significantly.

Models	CreateDebate			
	Abortion	Gay Rights	Marijuana	Obama
MLE	67.4 ± 7.5	74.0 ± 5.3	74.8 ± 7.5	63.0 ± 8.3
MPLE	68.5 ± 7.8	74.2 ± 5.7	75.7 ± 8.2	65.7 ± 10.5
CP-MPLE	**71.4 ± 3.8**	**76.5 ± 2.5**	**77.3 ± 3.1**	**68.9 ± 4.0**

Table 3.9: Accuracy results for **AD** model on CREATEDEBATE.COMhighlights the greatest significant gains from CP-MPLE.

debate sites exhibit biases in labels. Often, both the stances on a topic and pairwise disagreements are skewed. The goal of this experiment is to validate the benefits of CP-MPLE on both prediction accuracy and balance.

We first evaluate classification accuracy on the author stance prediction task across all eight topics from both forums. We compare CP-MPLE against the structured voted perceptron algorithm (MLE) and maximum pseudolikelihood learning (MPLE). Table 3.8 shows the accuracy across 4FORUMS.COM topics for all weight learning methods after we select cost parameters for CP-MPLE using cross-validation. Table 3.9 shows the performance across CREATEDEBATE.COM topics. Throughout this evaluation, bolded figures represent significant improvements at a statistical significance level of 0.05 over MLE.

The results show that CP-MPLE improves upon the accuracy of MLE in all topics across both online forums, with best gains of 9% in the Obama topic of CREATEDEBATE.COM. Indeed, we see on average greater gains in the CREAT-

Learning	4Forums			
	Abortion	Evolution	Gay Marriage	Gun Control
MLE	74.4 ± 3.9	87.7 ± 1.8	86.1 ± 2.7	17.2 ± 12.8
MPLE	74.1 ± 4.2	87.9 ± 2.1	86.3 ± 2.7	18.9 ± 12.8
CP-MPLE	**74.7 ± 2.4**	**88.6 ± 1.0**	**86.7 ± 1.1**	**30.9 ± 3.3**

Table 3.10: F_1 scores for **AD** model on 4Forums.comshows trade-off between precision and recall of predictions across learning methods. CP-MPLE yields most balanced predictions.

Models	CreateDebate			
	Abortion	Gay Rights	Marijuana	Obama
MLE	73.9 ± 7.4	82.8 ± 3.9	84.7 ± 5.2	59.5 ± 9.6
MPLE	74.6 ± 8.0	83.6 ± 3.6	85.4 ± 5.7	63.8 ± 11.6
CP-MPLE	**77.2 ± 3.2**	**84.5 ± 1.6**	**86.2 ± 2.0**	**67.1 ± 4.2**

Table 3.11: F_1 scores for **AD** model on CreateDebate.comshows that CP-MPLE gives strongest improvements in this imbalanced domain.

eDebate.com topics where CP-MPLE improves up to four accuracy points over MLE in both abortion and Obama topics. In 4Forums.com topics, the best gains are achieved in evolution and gay marriage topics. Across both forums, the topics where CP-MPLE boosts performance most correlate to the most skewed topics in Table 3.2, matching our intuition of the cost parameters. We also see that standard MPLE algorithm performs comparably with MLE. Although MPLE sees accuracy gains in four topics, the increased variance makes these gains insignificant. This suggests that pseudolikelihood alone remains insufficient to improve upon the structured perceptron algorithm. However, the CP-MPLE algorithm which augments empirical loss terms that capture FP and FN cost gives us the best predictive performance.

As a second line of investigation, we evaluate the balance in stance predictions by computing average F_1 scores to compare the learning methods across both

forums. Table 3.10 shows these F_1 results for 4FORUMS.COM and Table 3.11 presents the corresponding evaluation for CREATEDEBATE.COM. We see that in 4FORUMS.COM, MPE and MPLE only achieve an F_1 of up to 18.9 on the gun control topic which Table 3.2 shows is skewed towards the ANTI stance. CP-MPLE improves the F_1 to 30.9, which is promising though it leaves room for further gain. Interestingly, again we see that the greatest and most significant gains are achieved across topics in CREATEDEBATE.COM, with F_1 improvements of up to 13% on the Obama topic. The results demonstrate that CP-MPLE can better mitigate the imbalance in stance prediction, especially on topics that exhibit a greater skew in the training labels, as shown by Table 3.3.

To summarize our conclusions from all experiments, the results suggest that author-level modeling is the preferred strategy, regardless of the prediction task. In this scenario, it is essential to explicitly model disagreement in the collective classifier. Our top performing **AD** model statistically significantly outperforms the respective prediction task baseline on 6 out of 8 topics for both tasks with p-values less than 0.001. Based on our experimental results, we recommend the full author-disagreement model **AD** as the classifier of choice. Our second key finding is that the CP-MPLE learning algorithm further improves **AD** performance due to the presence of imbalanced labels.

3.8 Discussion

The prediction of user stance in online debate forums is a valuable task, and modeling debate dialogue is complex and requires many decisions such collective or non-collective reasoning, nuanced or naive use of disagreement information, and post versus author-level modeling granularity. We systematically explore each choice, and in doing so build a unified joint framework that incorporates each

salient decision. Our method uses a hinge-loss Markov random field to encourage consistency between local classifier predictions for stance and disagreement information. We find that modeling at the author level gives better predictive performance regardless of the granularity of the prediction task, and that nuanced disagreement modeling is of particular importance for author-level collective modeling. The resulting collective classifier gives improved predictive performance over both the simple non-collective and standard collective approaches, with a running time overhead of only a few minutes. We show that these performance gains can be further improved with our novel learning algorithm which explicitly encodes empirical loss in its objective. Finally, we demonstrate that our learning algorithm mitigates the imbalance in predictions especially in skewed training data settings, which benefits the downstream usage of these predictions in policy or decision-making.

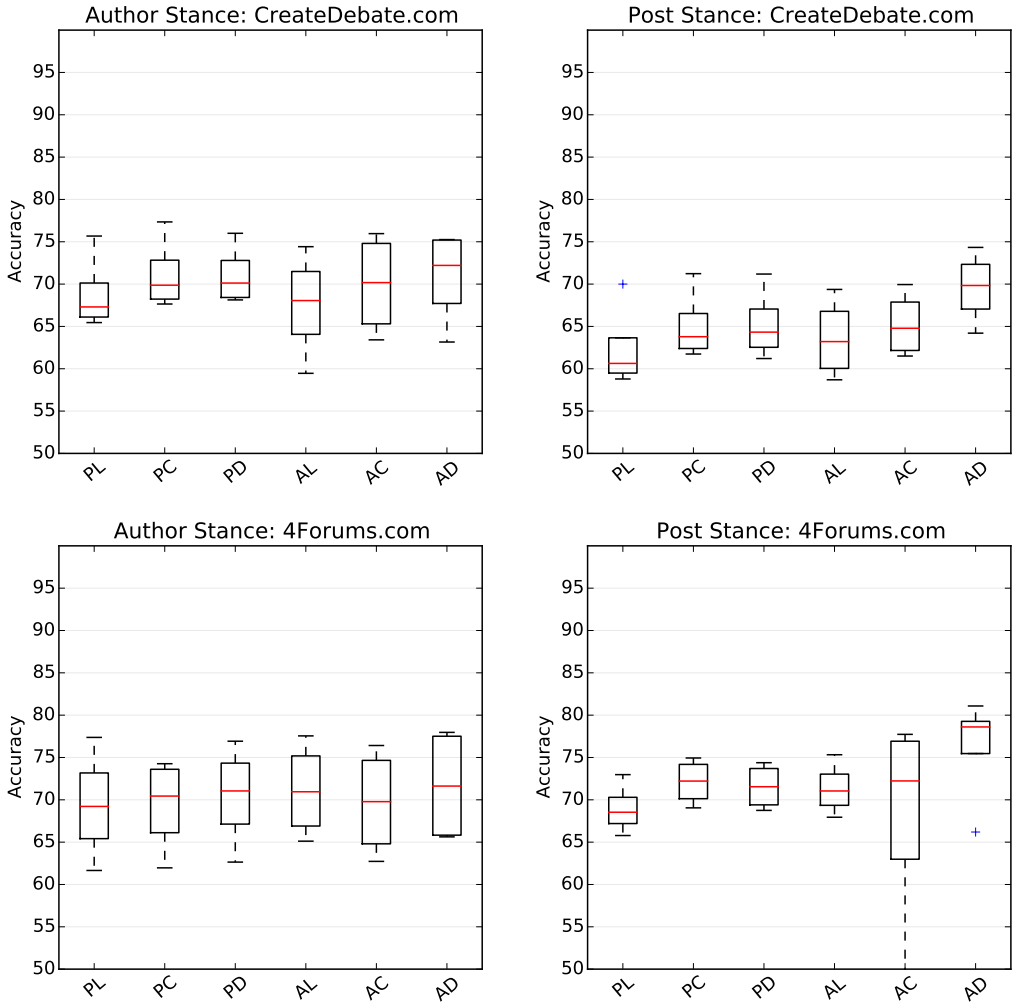


Figure 3.3: Overall accuracies per model for the author stance prediction task, computed over the final results for each of the four data sets per forum. Note that we expect significant variation in these plots, as the data sets are of varying degrees of difficulty.

Text	Stance
Post: I agree with everything except the last part. Safe gun storage is very important, and sensible storage requirements have two important factors.	ANTI
Reply: I can agree with this. And in case it seemed otherwise, I know full well how to store guns safely, and why it's necessary. My point was that I don't like the idea of such a law, especially when you consider the problem of enforcement.	ANTI

Figure 3.4: A post-reply pair by 4FORUMS.COM authors whose gun control stance is correctly predicted by **AD**, but not by **AC**.

Chapter 4

Fusing Multiple Sources

In the previous chapter, we developed useful collective modeling templates in the context of online debate forums and stance prediction. In this chapter, we turn our focus to another important computational science domain with a similarly complex relational data graph. We consider pharmacological graph where vertices are drug treatments. We observe several types of edges between these drug treatments, including adverse or beneficial drug-drug interactions and similarities between drugs based on molecular, chemical or annotation-based traits. In contrast to the debate forum graph in the previous chapter, this graph is multi-relational, requiring structured approaches to fuse these multiple sources of information.

In this domain, an important problem is predicting new interactions between drugs based on current knowledge of interactions and multiple drug similarities. As concurrent use of multiple medications becomes ubiquitous among patients, it is crucial to characterize both adverse and synergistic interactions between drugs. Probabilistic models of putative drug-drug interactions (DDIs) can guide in vitro testing and cut down significant cost and effort. With the abundance of experimental data characterizing drugs and their associated targets, such methods

must effectively fuse multiple sources of information and perform inference over the network of drugs. In this chapter, we build on the collective modeling patterns in Chapter 3 and propose a probabilistic approach for jointly inferring unknown DDIs from a network of multiple drug-based similarities and known interactions. We compare against two methods including a state-of-the-art DDI prediction system across three experiments and show best performing improvements of more than 50% in AUPR over both baselines. We find five novel interactions validated by external sources among the top-ranked predictions of our model. These results highlight the importance of fusing sources of varying reliability when developing collective probabilistic models. In the following chapter, we extend this idea to combining several constraints types in PSL and sources of prior information when discovering causal knowledge.

4.1 Drug-drug Interaction

Increasingly, patients use multiple pharamceutical drugs simultaneously to treat their illnesses. Interactions between drugs can result in reduced efficacy of one or more drugs, and in some cases, even deletrious side-effects. The risk of adverse effects is higher in demographics like the elderly that commonly take multiple medications at once. On the other hand, certain drugs interact to produce synergistic effects that are more effective in combatting diseases like cancer [104, 23]. Crowther et al. [26] characterizes a drug-drug interaction (DDI) as a drug effect that is greater or less than expected in the presence of another drug. While it remains crucial to verify potential DDIs *in vitro*, it is prohibitively expensive to exhaustively test all possible interactions. Therefore, computational modeling and predictive methods provide a viable way to identify the most salient potential interactions for downstream experimental validation [171].

Interactions between drugs are classified as *pharmacokinetic* and *pharmacodynamic*. Computational and mathematical modeling methods rely on current understanding of the mechanisms underlying each of these types of interactions and are specific to each interaction type. A pharmacokinetic interaction with a drug affects the process by which the other drug is absorbed, distributed, metabolized or excreted in the body [26]. On the other hand, drugs acting on the same receptor, site of action, or physiological system constitutes a pharmacodynamic interaction. While pharmacokinetic interactions are usually associated with an adverse or exaggerated response, pharmacodynamic interactions are implicated in both synergistic and detrimental effects. Many pharmacokinetic interactions are facilitated by the enzyme family *Cytochrome P450* (CYP) and extensive but incomplete knowledge of its mechanisms have been used for computational modeling of pharmacokinetic interactions [41]. Similarly, prior work has applied mathematical modeling of known drug response mechanisms to simulate and predict pharmacodynamic interactions [71, 68].

In contrast to computational modeling, statistical and predictive methods leverage data and evidence from related experiments as domain knowledge and biological priors. Recent advancements in high-throughput experimentation have generated a wealth of biological characterizations of drug compounds and their target genes [46]. A key challenge for statistical models of drug-drug interactions, or the closely related problem of drug-target interactions, lies in fusing or combining information from multiple data sources. Much related work has developed ways of computing similarity scores between drugs or pairs of drugs to be used as features for machine learning classifiers [20, 55, 4, 156, 155]. Sophisticated algorithms such as restricted Boltzmann machines and matrix factorization are especially effective in combining two types of similarities by learning latent rep-

resentations of the entities [163, 19, 54], however, they do not inherently support multiple similarities in the same model. Statistical methods for DDI prediction are more generalizable as they do not rely on extensive expert knowledge of each mode of interaction. Although Park et al. [109], Huang et al. [59] apply their predictive methods only to pharmacodynamic interactions, statisicals models can be easily extended to both types of interactions. To the best of our knowledge, [55] present state-of-the-art results for drug-drug interaction prediction of both pharmacokinetic and pharmacodynamic interactions with their INDI system for combining similarity measures to use features for a local logistic regression classifier.

However, these similarity-based methods neglect the structural information encoded in the biological network of drugs and their interactions. Two general types of approaches have been studied for adding network information to similarity-based features: methods that compute additional network-based features and methods that perform inference directly over the structure of the network. We refer to these as network-similarity methods and network-based inference methods, respectively. Both kinds of approaches begin by formulating a graph of drugs and their interactions. Network-similarity methods proceed by computing *relational* features, based on the local neighborhoods of drugs such as neighborhood overlap and other well-studied network attributes [20, 19, 59]. The relational features supplement the similarity information given as input to a classifier. In contrast, network-based inference methods reason over the graph structure when predicting interactions.

Multiple network-based inference approaches have been introduced for the closely related problem of drug-target interaction prediction. Bleakley and Yamanishi [15] formulate the problem of inferring missing links in a bipartite graph of drugs and targets, and introduce a model that uses local bipartite structure

for prediction. Cheng et al. [21], Mei et al. [96] similarly leverage local bipartite topology for inference and Park et al. [109] introduce a random walk approach for reasoning over the network of drugs and targets. However, local network-based features cannot enforce global constraints based on the full graph of entities. Given local relational features, current network-based inference methods follow traditional machine learning algorithms in assuming the instances to be independent and identically distributed. In recent work, Fakhraei et al. [45, 44] improve upon existing bipartite drug-target interaction prediction approaches using the probabilistic programming framework *Probabilistic Soft Logic* (PSL) to jointly classify all interactions, fusing similarity relations and global network information.

In this chapter, we formalize the problem of network-based drug-drug interaction prediction using multiple similarity relations. We *collectively* predict drug-drug interactions, considering statistical dependencies between predictions along with knowledge of observed interactions using Probabilistic Soft Logic. We apply our collective approach to predict DDIs on three kinds of interactions: (1) CYP-related interactions (2) non-CYP related interactions (3) general interactions documented by Drugbank [165]. For all settings, we evaluate our collective approach against two non-collective methods including state-of-the-art INDI system of Gottlieb et al. [55] and a non-collective PSL model. Our model achieves statistically significant improvement up to 5% in area under the ROC (AUC) results from Gottlieb et al. [55]. We further assess area under the precision-recall curve (AUPR) for all methods and show that our collective DDI prediction approach significantly outperforms the state-of-the-art baseline method by up to 50%. Finally, we present important novel DDIs predicted by our approach that are validated in literature.

4.2 Datasets

We use two datasets for our experimental evaluation. The first dataset, released by Gottlieb et al. [55], includes pairwise interactions between 807 drugs, with the drug IDs anonymized . We constructed the second dataset by extracting interactions from Drugbank for the 315 drugs used by Fakhraei et al. [45], Perlman et al. [116], where Drugbank IDs are provided for additional validation. The following section described interaction types and similarities used in these datasets.

4.2.1 Drug Interaction Data

For the first dataset, Gottlieb et al. [55] download 10,702 interactions from DrugBank and 70,099 interactions listed as moderate or high from Drugs.com website[165]. The dataset contains two types of interactions: (1) CYP-related interactions (CRDs), where both drugs are metabolized by the same cytochrome P450 (CYP) enzyme (2) non-CYP-related interactions, where no CYP is shared between the drugs (NCRDs). After filtering and processing, the final dataset includes 10,106 CRD and 45,737 NCRD DDIs [55] across 807 drugs.

For the second dataset, we download interactions from DrugBank version 4.3 for the 315 drugs used by Perlman et al. [116], Fakhraei et al. [45]. We cross referenced Drugbank IDs released for the 315 drugs to extract the listed drug interactions, resulting in 4293 known interactions.

4.2.2 Drug Similarity Data

Both datasets contain seven drug-drug similarities. Four of these similarity measures are drug-based: Chemical-based, Ligand-based, Side-effect-based, Annotation-based. Three similarities are between drug targets and computed

by aggregating over known targets for the drugs: Sequence-based, PPI network-based, and Gene Ontology-based. In the first dataset, Gottlieb et al. [55] average maximal similarities between the associated targets for drugs that have more than one target. In the second dataset, we average over all possible pairwise similarities between target genes for drugs that have multiple targets.

Following section provides a brief description of the methods in Gottlieb et al. [55], Perlman et al. [116] for similarity extraction:

Chemical-based is the Jaccard similarity, or closely related Dice similarity, of molecular fingerprints from pairs of drugs. Molecular fingerprints are retrieved from cheminformatics toolkits such as chemical development kit (CDK) [145] or RDKit using canonical SMILES[1]. Fingerprinting methods represent molecules as bit strings for fast similarity computation and are grouped into hashing-based and structural methods. Hashed fingerprints such as the Daylight method rely on hash functions to represent linear substructures of molecules as bit strings. Structural fingerprints such as MACCS, Atom-Pair, Morgan and Feature-Based Morgan methods use features of molecular substructures to compute bit strings. The Jaccard and Dice similarity scores between two sets X and Y are defined as

$$\text{Jaccard}(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}, \text{Dice}(X,Y) = \frac{2|X \cap Y|}{|X| + |Y|}$$

We obtain all fingerprints described above from RDKit and additionally, the hashed fingerprint from CDK computed with default values as used by Gottlieb et al. [55]. In our experiments, we use the hashed fingerprint from CDK after comparing performance of all fingerprinting methods on development data.

[1]Simplified Molecular Input Line Entry Specification

Ligand-based is the Jaccard similarity between the corresponding sets of protein-receptor families for each drug pair. The protein-receptor is obtained from the similarity ensemble approach (SEA) search tool [74] Drugs' canonical SMILES compared with a collection of ligands[2].

Side-effect-based is the Jaccard similarity score between common side-effects for each pair of drugs.

Annotation-based is the Resnik semantic similarity [125] of Drugs' ATC codes mapped to the World Health Organization ATC classification system [132].

Sequence-based is the Smith-Waterman sequence alignment score between the corresponding drug targets (proteins). They are normalized via dividing the pair-wise score by the geometric mean of the alignment scores of each sequence against itself, suggested in [15].

Protein-protein interaction network-based is the distance between pairs of corresponding drug targets using their corresponding proteins in the human protein-protein interactions network via an all-pairs shortest path algorithm.

Gene Ontology-based is the Resnik semantic similarity [125] between Gene Ontology annotations of drugs' corresponding targets.

For more detailed descriptions of these similarities, refer to Perlman et al. [116], Gottlieb et al. [55].

[2]A substance that binds with a biomolecule to serve a biological purpose.

4.3 Problem Statement

We consider the problem of inferring new edges in a partially observed graph of interactions between drug vertices by leveraging multiple known similarity relations between vertices. We are given a set of drugs $D = \{D_1 \dots D_n\}$. We observe a set of interaction edges between the drugs denoted by $n \times n$ interaction matrix I where $I_{ij} = 1$ indicates an interaction between d_i and d_j and is 0 indicates an unobserved or missing edge. Additionally we are given a set of $n \times n$ drug-drug similarity relations encoded by tables $\{M_1 \dots M_k\}$ where $M_{l_{ij}} \in [0, 1]$ and indicates similarity between d_i and d_j according to biological similarity l.

We define a drug network as a multigraph $G = (V, \mathbf{E})$ where $V = D$ is the vertex set of drugs and $\mathbf{E} = \{M_1 \dots M_k\} \cup I$ is the collection of multiple edge types given by the similarity relations and the interaction matrix I. The drug-drug interaction prediction problem is to use all the information encoded in G to predict the unobserved interaction edges between drug vertices in G.

4.4 Approach

4.4.1 Collective Drug-drug Interaction

Given all the information G, we want to infer interaction values for missing edges $U = \{(d_i, d_j) | I_{ij} = 0\}$. Many techniques have been studied for inference of missing links but here we focus on the intersection of two well known approaches: network-based methods and collective probabilistic methods. Generally, network-based inference techniques make use of the structure of G by considering the local neighborhoods for each d_i and d_j in edges we want to infer. For example, network-based methods might include set similarity of the neighbors of d_i and d_j along with the local edge similarites encoded in G. Collective probabilistic methods

learn joint distributions $P(U, G)$ to infer the most probable joint assignment to all edges in U thereby leveraging statistical dependencies between prediction targets as well as the observations in G. Network-based collective methods combine the two techniques by parametrizing $P(U, G)$ according to structural features of G. Collective prediction methods have been shown to work well in the closely related setting of drug-target interaction prediction [45]. Below we describe hinge-loss Markov random fields and probabilistic soft logic, a framework for performing network based collective inference, and describe our model for drug-drug interaction prediction.

Drug-drug Interaction PSL Model

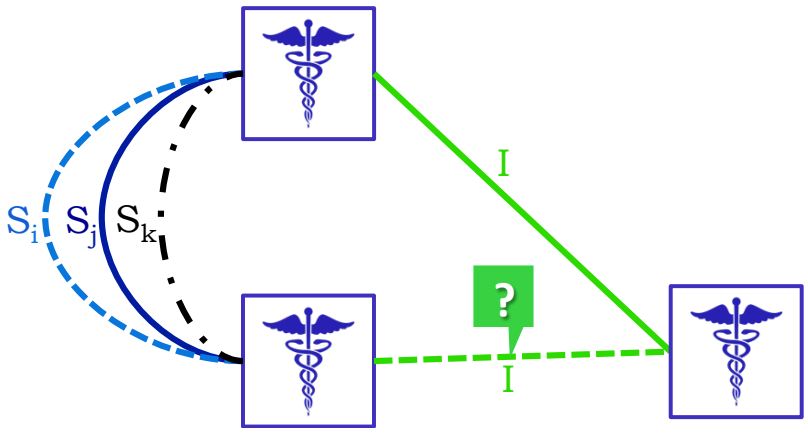


Figure 4.1: Triad-based drug-drug interaction prediction rules.

We propose a PSL model for collective drug-drug interaction prediction that fuses several sources of information. The rules of a PSL model capture beliefs or knowledge about the problem domain. For the drug-drug interaction domain encoded by drug network G, we assert that a drug is likely to be involved in an interaction if it is similar to another drug that is a known interactor. To model the notion of similarity, we are interested in fusing multiple sources of drug similarity.

We make this concrete in the full set of rules for drug-drug interaction prediction shown in figure 4.2.

$$
\begin{aligned}
w_1 : \textsc{Sim}_{Chemical}(D_1, D_2) \wedge \textsc{Interacts}(D_2, D_3) &\rightarrow \textsc{Interacts}(D_1, D_3) \\
w_2 : \textsc{Sim}_{Ligand}(D_1, D_2) \wedge \textsc{Interacts}(D_2, D_3) &\rightarrow \textsc{Interacts}(D_1, D_3) \\
&\dots \\
w_7 : \textsc{Sim}_{GO}(D_1, D_2) \wedge \textsc{Interacts}(D_2, D_3) &\rightarrow \textsc{Interacts}(D_1, D_3)
\end{aligned}
$$

Figure 4.2: PSL model for collective drug-drug interaction prediction.

where we have one rule for each drug similarity described in section 4.2, resulting in seven rules. We represent the prediction target with the $\textsc{Interacts}(D_1, D_3)$ predicate. Given a set of drugs d_1, d_2, and d_3 with known interaction between d_2 and d_3, the rule results in groundings of the form:

$$
\begin{aligned}
w_1 : \textsc{Sim}_{Chemical}(d_1, d_2) \wedge \textsc{Interacts}(d_2, d_3) &\rightarrow \textsc{Interacts}(d_1, d_3) \\
w_2 : \textsc{Sim}_{Chemical}(d_2, d_3) \wedge \textsc{Interacts}(d_3, d_1) &\rightarrow \textsc{Interacts}(d_2, d_1)
\end{aligned}
$$

Figure 4.3: Small subset of ground PSL rules.

We exclude multiple symmetric groundings for ease of exposition. The ground rules show the propagation of similarity information between target variables. The inferred value of $\textsc{Interacts}(d_1, d_3)$ also informs the value of $\textsc{Interact}(d_2, d_1)$. Following [45], we refer to these as 'triad rules' as they encourage triangle completion, or triadic closure. Figure 4.1 shows a schematic overview of the triad rules. The predicted interaction edge provides evidence for other inferences, resulting in a flow of information throughout the network. This form of collective prediction leverages the full structure of the drug network graph G while combining multiple sources of similarity information. To fully evaluate the impact of joint prediction, we describe below two baseline methods that work non-collectively and assume independence between predicted interactions.

4.4.2 Comparison Methods

We compare against two non-collective methods including the state-of-the-art INDI framework for inferring interactions between drugs [55]. We describe each of these below.

State-of-the-art INDI method

[55] introduce the INDI framework for novel drug-drug interaction prediction. They introduce a method for computing similarity scores between target interaction edges to known interaction edges based on the given drug-drug similarities. For each target drug-pair, each pairwise combination of similarities is considered for computing the similarity score to the most similar known drug interaction. The procedure effectively performs nearest neighbor search using different similarity distance measures. Each score is then used as a feature to train a logistic regression classifier. We refer to [55] for full details.

Non-collective PSL Model

To quantify the effect of collective prediction, we also use a non-collective PSL model that considers the dependencies between the target interactions and observed interactions only, as in the INDI method. Formally, we modify the triad rules above as follows: where we introduce the $\textsc{Interacts}_{Obs}$ predicate to limit

$$w_1 : \textsc{Sim}_{Chemical}(D_1, D_2) \wedge \textsc{Interacts}_{Obs}(D_2, D_3) \rightarrow \textsc{Interacts}(D_1, D_3)$$
$$\dots$$
$$w_7 : \textsc{Sim}_{GO}(D_1, D_2) \wedge \textsc{Interacts}_{Obs}(D_2, D_3) \rightarrow \textsc{Interacts}(D_1, D_3)$$

Figure 4.4: Non-collective PSL model for drug-drug interaction prediction.

the triadic closure of predicted interactions to known interactions only.

4.4.3 Experimental Setup

In order to validate our collective drug-drug interaction prediction method and compare against state-of-the-art methods, we perform experiments on the two drug interaction networks described in section 4.4. For each dataset, we perform ten-fold cross-validation across all pairs of interactions. We use eight folds as interaction evidence or observations, one fold as training labels to learn weights for the rules, and the final fold as a held-out test set. All similarities between drugs are used as evidence, or features, for the models. The similarity distributions are highly left-skewed, which is problematic for the soft truth interpretation used by PSL, as values below 0.5 do not highly affect the inference. We transform all similaritity values between drugs by taking the cube-root to normalize the distributions and allow for proper interpretation by PSL.

We compute area under the precision-recall curve (AUPR) for the positive class, area under the ROC (AUC) and F1 score on the test set. Link prediction tasks usually suffer from class imbalance as true positive links are sparse compared to true negatives. Related work on general link prediction and DDI prediction report AUC because it is more robust to the skewness than metrics such as accuracy. However, AUC is still sensitive to the high number of negative examples. For practical downstream biological validation of predicted DDIs, it is more important to have a reliable ranking of candidate positive interactions. The precision-recall curve better captures the effectiveness of models at discriminating true positive examples. F1 score is another measure of classification accuracy and can interpreted as a weighted average of precision and recall. Since PSL outputs real-valued truth scores and logistic regression produces class probabilities, we threshold the values to $\{0,1\}$ to compute the F1 score. We perform grid search over a range of threshold values between $[0,1]$ to obtain best-performing

thresholds.

We implement the INDI feature computation method in `Matlab` by extending a related implementation of the computation for the drug-target interaction prediction setting [45, 116]. We use the logistic regression classifier provided in the `glmfit` package with default settings. For our models, we use the open-source PSL framework. We run 700 iterations of the structured voted-perceptron weight learning algorithm in PSL and use default settings for the ADMM inference algorithm. We will make all code and datasets publicly available.

Blocking Methods for PSL

In a drug network with n drugs and n^2 interactions where PSL considers dependencies between pairs of interactions, the computational complexity reaches $O(n^4)$, which quickly becomes expensive for large networks. To make the approach scalable, we employ a common techniques to block unimportant links from being grounded out by the model. In the PSL triad rule setting, for each similarity i, we limit the possible $\textsc{Similar}_i(D_1, D_2)$ edges that are considered for each drug D_1. By blocking on the similarity links, we restrict the grounding of all possible triads to only the ones that are most likely.

To block similarities in the grounded out PSL models, for each drug, we perform nearest neighbor search to pick the top 15 most similar other drugs as evidence for $\textsc{Sim}_i(D_1, D_2)$. In the first drug dataset, for the CRD interaction experiments, we use a more restrictive blocking method to induce more sparsity since CRD interactions are rarer. When searching for the 15 nearest neighbors for each drug, we restrict ourselves to those drugs that have appeared in at least one observed interaction in the full network. In this sparser setting, some drugs may not appear in any $\textsc{Sim}_i(D_1, D_2)$ groundings. For those drugs, we additionally retrieve

five most similar other drugs using standard nearest neighbor search and include the pairs as evidence for $\text{SIM}_i(D_1, D_2)$. [45] provide more comprehensive analysis on techniques for blocking.

4.5 Experimental Results

4.5.1 Comparison to State-of-the-art Baselines

We compare our proposed collective PSL approach for DDI prediction to two baselines including the state-of-the-art INDI system with 10-fold cross-validation experiments. We apply the three methods to each fold and report average and standard deviations of our chosen metrics for each model. We refer to the INDI system as INDI, the non-collective PSL baseline as NC-PSL, and collective PSL model as PSL. Tables 4.1-4.3 present average and standard deviation for area under the precision-recall curve (AUPR) and area under the ROC (AUC) from cross-validation experiments on the three interaction types from two datasets: (1) CYP-related interactions (CRD) from Drugs.com and Drugbank [55] (2) Non-CYP-related interactions (NCRD) from Drugs.com and Drugbank [55] (3) General interactions from DrugBank. Bolded results highlight statistically significant improvement over both baselines with $\alpha = 0.05$. Figures 4.5, 4.6 and 4.7 show precision-recall curves of all methods plotted for interaction type settings (1), (2) and (3) respectively. Additionally, to assess the benefit of fusing multiple similarities, we compare against our collective PSL model implemented with single similarities. Table 4.4 shows AUPR for the collective PSL model for single similarities across interaction type settings (1), (2) and (3).

Table 4.1: Average AUPR, AUC and F1 scores (with best threshold t indicated), and standard deviation for 10 fold CV comparing all DDI prediction models for CRD interactions from dataset 1.

Method	**AUPR-Pos**	**AUROC**	**F1**
INDI	0.15 ± 0.007	0.92 ± 0.003	0.24 ± 0.005 ($t = 0.1$)
NC-PSL	0.15 ± 0.01	0.91 ± 0.004	0.23 ± 0.01 ($t = 0.8$)
PSL	**0.34 ± 0.02**	**0.96 ± 0.003**	**0.4 ± 0.02 ($t = 0.3$)**

Table 4.2: Average AUPR, AUC and F1 scores (with best threshold t indicated), and standard deviation for 10 fold CV comparing all DDI prediction models for NCRD interactions from dataset 1.

Method	**AUPR-Pos**	**AUROC**	**F1**
INDI	0.64 ± 0.01	0.95 ± 0.003	0.63 ± 0.01 ($t = 0.35$)
NC-PSL	0.70 ± 0.006	0.96 ± 0.001	0.62 ± 0.01 ($t = 0.9$)
PSL	**0.78 ± 0.006**	**0.97 ± 0.001**	**0.70 ± 0.01 ($t = 0.3$)**

Table 4.3: Average AUPR, AUC and F1 scores (with best threshold t indicated), and standard deviation for 10 fold CV comparing all DDI prediction models for general interactions from dataset 2.

Method	**AUPR-Pos**	**AUROC**	**F1**
INDI	0.47 ± 0.04	0.91 ± 0.01	0.51 ± 0.03 ($t = 0.2$)
NC-PSL	0.56 ± 0.04	0.95 ± 0.006	0.6 ± 0.03 ($t = 0.5$)
PSL	**0.69 ± 0.02**	**0.96 ± 0.006**	**0.67 ± 0.02 ($t = 0.4$)**

Table 4.4: Average AUPR and standard deviation for 10 fold CV for single similarity collective DDI prediction models across all interaction types

Similarity	**CRD**	**NCRD**	**General**
ATC	0.18 ± 0.01	0.73 ± 0.01	0.68 ± 0.02
Chemical	0.32 ± 0.02	0.58 ± 0.01	0.46 ± 0.04
Distance	0.31 ± 0.03	0.63 ± 0.004	0.35 ± 0.04
Gene Ontology	0.33 ± 0.02	0.63 ± 0.004	0.39 ± 0.04
Ligand	0.18 ± 0.01	0.67 ± 0.01	0.37 ± 0.03
Sequence	0.29 ± 0.02	0.63 ± 0.004	0.37 ± 0.04
Side Effect	0.30 ± 0.01	0.56 ± 0.01	0.51 ± 0.03

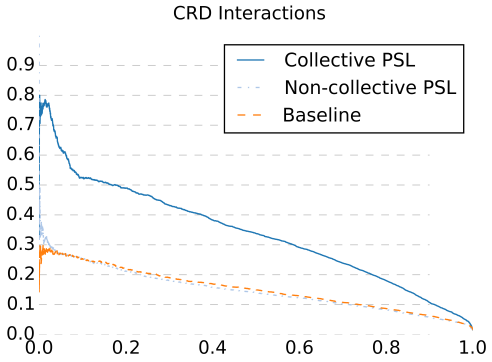


Figure 4.5: Precision-recall curves comparing all DDI prediction models on CRD Interactions dataset.

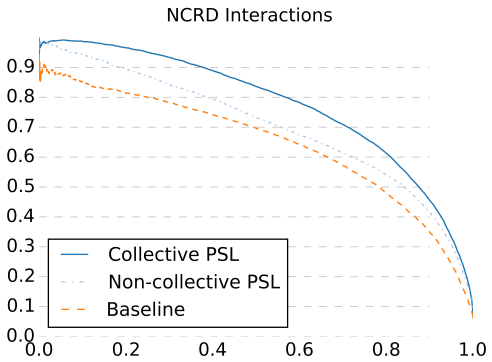


Figure 4.6: Precision-recall curves comparing all DDI models on NCRD Interactions dataset.

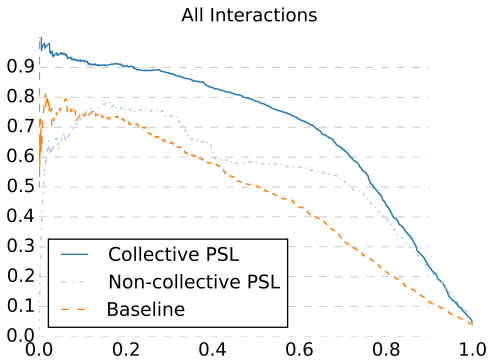


Figure 4.7: Precision-recall curves comparing all DDI models on general interactions dataset.

[55] report AUC results consistent with our evaluation of the INDI system as given in tables 4.1 and 4.2. Our collective PSL model statistically significantly outperforms both baselines in AUC, AUPR and F1-score for all three interaction type prediction experiments. For AUPR, in the best case CRD interactions setting, our collective model improves up to 50% in AUPR over the state-of-the-art INDI system and non-collective PSL model, from 0.15 to 0.34. For the NCRD and general interactions, the collective PSL approach sees gains of up to 20% in AUPR over both baselines. The collective model improves up to 0.05 in AUC over the state-of-the-art INDI method, with AUC as high as 0.97 for the NCRD interaction setting, signficantly improving over the 0.95 achieved by the INDI system. For F1-score, our collective model improves close to 50% over the INDI and non-collective baselines for the CRD setting and up to 30% for NCRD and general interaction settings. Interestingly, the non-collective PSL method performs at least as well as the INDI system in setting (1) and for settings (2) and (3), significantly outperforms the INDI system in AUPR and AUC. This improvement by the non-collective PSL model demonstrates the method's effectiveness in combining multiple similarities as well as or better than the state-of-the-art similarity

Table 4.5: Top ranked PSL model predictions for interactions unknown in DrugBank

Rank	Drug Bank IDs	Drug Bank IDs
1	**DB00870; DB01418**	**Suprofen and Acenocoumarol**
2	DB01067; DB00839	Glipizide and Tolazamide
3	**DB01297; DB00806**	**Practolol and Pentoxifylline**
4	DB00870; DB00806	Suprofen and Pentoxifylline
5	DB00272; DB01232	Betazole and Saquinavir
6	**DB00870; DB01032**	**Suprofen and Probenecid**
7	DB00939; DB01418	Meclofenamic acid and Acenocoumarol
8	**DB00414; DB01032**	**Acetohexamide and Probenecid**
9	DB01297; DB01392	Practolol and Yohimbine
10	**DB01097; DB01262**	**Leflunomide and Decitabine**

combination technique used by INDI. The gains achieved by the fully collective PSL model highlights the benefits of joint inference over the full drug-drug interaction network. Additionally, for all interaction settings, the multiple similarity collective approach significantly improves in AUPR over all individual similarity collective models. This result supports the findings of Fakhraei et al. [45], Gottlieb et al. [55] that multiple similarities benefit performance of both drug-target and drug-drug interaction prediction tasks.

4.5.2 Validation of Unseen Interaction Predictions

In order for statistical methods to be useful for domain experts, predictive models should produce highly probable novel interactions for subsequent in vitro testing. Thus, following Gottlieb et al. [55], Fakhraei et al. [45], Bleakley and Yamanishi [15], we compare top-ranked, unseen DDI predictions produced by our collective PSL model with evidence from medical and biological data sources. These predictions are novel with respect to Drugbank interactions used as training data and validate the ability of our collective approach to produce salient

interaction predictions given observations.

For this experiment, we use predicted drug-drug interactions from non-anonymized dataset 2 to cross-reference in literature. From our 10-fold cross validation experiments, we output the predictions and filter out those that are not present in Drugbank as verified interactions. We rank these new predictions and consider the top 10 interactions as shown in table 4.5. Bolded rows indicate drug pairs that are verified by literature or another database as interactors, or have substantive supporting evidence for potential interaction. We use the Interactions Checker tool provided by drugs.com (`http://drugs.com`) for validation, as these interactions were not used to train any of our models. Additionally, Drugbank provides the *BioInteractor* tool that uses drug-target, -enzyme and -transporter associations to predict highly probable interactions that are not included in the main database. Our collective PSL approach highly ranks five interactions that are substantiated by Interactions Checker or BioInteractor. Some interactions involve the following four drugs that are no longer FDA approved or used outside of the United States: Suprofen, Acenocoumarol (used worldwide but not in U.S.), Practolol and Acetohexamide. Because these drugs are presently less well-studied and documented, they arise naturally as test cases for our validation study.

The top predicted interaction is between Suprofen, a non-steroidal anti-inflammatory drug, and Acenocoumarol, an anticoagulant. BioInteractor characterizes the effect of Suprofen on Acenocoumarol as a CYP mediated pharmacokinetic interaction. The sixth most highly ranked prediction involving Suprofen and Probenecid, a uricosuric agent used to treat gout, is also classified by BioInteractor as a CYP related interaction. Acetohexamide is in the sulfonylurea class of compounds used to treat type-II diabetes and is predicted by our model to interact with Probenecid, which is highly protein-bound. Interactions Checker characterizes this particular

interaction as enhancing the hypoglycemic effects of sulfonyureas when taken together. The risk of using Probenecid together with Acetohexamide is high with the elderly, who are commonly treated simultaneously for gout and diabetes.

The collective PSL model also ranks Decitabine, used to treat Leukemia, and Leflunomide, used for rheumatoid arthritis treatment, as interactors. Interactions Checker indicates a major interaction between Leflunomide and Decitabine in conjunction since both are immunosuppressants and can have additive effects to increase risk of serious infection. Ranked third, Pentoxifyline, a vasodilator and anti-inflammatory used to improve blood circulation, is predicted to interact with Practolol, a beta-blocked formerly used to treat cardiac arrhythmias. Although Drugs.com does not list this particular interaction, the Interactions Checker lists moderate interaction between Propranolol, beta-blocker now used in place of Practolol, and Pentoxifyline. The prediction of an effect on Pentoxifyline by a drug chemically similar to Propranolol also demonstrates the effectiveness of the PSL triad rules. The propagation of likely interaction across drugs that are similar is also evident in the second ranked prediction of interaction between Glipizide and Tolazamide. Both are sulonyureas like Acetohexamide and are used to treat type-II diabetes. Though the drugs deliver similar responses, currently there is no strong evidence of their interaction.

We compare the predictions in Table 4.5 to the top ten novel interactions predicted by the INDI system. In contrast, only three out of the ten predictions made by INDI can be verified by BioInteractor or Interactions Checker: (1) Ciprofloxacin and Lomefloxacin (rank 2) (2) Methotrexate and Lomefloxacin (rank 4) (3) Mifepristone and Lomefloxacin (rank 6). There are no overlaps with the predictions ranked highly by PSL. Lomefloxacin and Ciprofloxacin both fight infection, Methotrexate treats cancers and Mifepristone ends pregancy. Interest-

ingly, both the collective PSL model and the INDI system predict interactions involving major cancer drugs, Decitabine and Methotrexate, respectively.

4.6 Discussion

In this work, we formulate the problem of collective drug-drug interaction prediction. We introduce a joint probabilistic approach using the PSL framework to fuse multiple sources of similarity information together with domain-knowledge of the network structure for this domain. The originality of this work lies in proposing and experimentally validating a highly scalable, collective probabilistic approach for DDI prediction that is easily extensible with different sources of information and similarity measures. We evaluate our approach on two datasets containing three types of interactions, including one extracted for this work with known Drugbank IDs for additional validations. We perform ten-fold cross-validation on all settings and see that our collective PSL model significantly outperforms two other similarity-based methods, including the state-of-the-art INDI system, on two important metrics for link prediction, AUPR and AUC. Our best performing PSL model improves more than 50% upon AUPR of both baselines and achieves a best AUC of 0.97. Moreover, the non-collective similarity-based method implemented in PSL also significantly outperforms INDI in two settings and performs comparably to INDI in other settings. This result also highlights the effectiveness of PSL as an extensible framework for similarity-based reasoning that enjoys the benefits of collective inference shown by the first result. Furthermore, the top then predictions of our best performing collective PSL methods contain five interactions that are unseen in Drugbank but substantiated by Drugs.com and the BioInteractor tool on Drugbank. This result signifies the usefulness of our collective approach for producing high-quality predictions that

can be verified experimentally downstream.

Another benefit of our collective PSL method is scalability and speed. The focus of the INDI method is combining similarities by computing interaction edge-based similarity score using a nearest-neighbor search approach. This feature computation is a computationally expensive procedure, requiring $O(n^4)$ passes over the drug entities. For a dataset containing 807 drugs, this computation takes approximately 12 hours on average per fold on a single 32GB machine with 4 cores. The comparable non-collective PSL model introduced in this work takes approximately 1 hour for a round of weight learning and inference per fold on the same machine. The collective PSL model completes computation for a fold in approximately 7 hours. The PSL framework admits highly efficient, polynomial-time inference and here, we further reduce computational complexity by blocking unnecessary groundings of the model. Scalability is crucial for link prediction tasks in increasingly massive biological networks, as new drugs are frequently introduced.

The task of DDI prediction is closely related to problems of predicting drug side effects, drug adverse reactions, and synergistic drug pairs. In fact, predicting synergistic drug interactions is just a specific subtask of the DDI prediction problem. Our collective approach for similarity-based reasoning in networks can be applied and generalized to all these related settings.

Chapter 5

Discovering Causal Structure

Chapter 3 and 4 focused on inference tasks in two complementary computational science relational data graphs. The collective modeling patterns developed for the multi-modal online debate forums domain benefits the multi-relational problem of inferring drug interactions amidst several sources of information. In these settings, we addressed the first two challenges of adapting PSL to computational science domains: 1) evaluating modeling decisions to develop templates for collective inference; and 2) fusing collective inference with multiple signals of varying fidelity. In both domains, collective PSL models outperformed comparable methods. However, prior work and knowledge of these domains informed our choice of model structure such as specifying appropriate rules for the task. In contrast, for several computational science tasks, domain knowledge is limited and the goals of probabilistic modeling lie in model structure discovery.

This chapter focuses on discovery of causal structure, the first of two important structure discovery tasks addressed in this thesis. Causality is a stronger notion that probabilistic dependency between two variables. While dependencies tell us that observing one variable's value gives us more information about another variable's value, causality tells us that changing a variable's value always changes

the value of the variable it affects. Going beyond single causal edges, scientific phenomena such as gene regulation, population genetics, or atmospheric patterns are best described by a graph of cause-and-effect relationships between key entities such as genes or air masses. Directed acyclic graphs whose edges encode causal relations are extensively applied and studied, based on foundational work by Pearl and Verma [112], Pearl et al. [113], Pearl [111]. In the context of scientific settings, the role of computational methods is to infer the structure of these causal graphs using observational data and domain knowledge. The work in this chapter identifies three key requirements for inferring the structure of causal networks for scientific discovery: (1) robustness to noise in observed measurements; (2) scalability to handle hundreds of variables; and (3) flexibility to encode domain knowledge and other structural constraints.

We first formalize the problem of joint probabilistic causal structure discovery. The approach introduced in this chapter builds on and significantly extends the templates developed in the preceding chapters, fusing sources of information and collectively propagating inferences but combining new forms of structural constraints. We propose CausPSL, an approach using PSL that exploits multiple statistical tests, supports efficient optimization over hundreds of variables, and can easily incorporate these structural constraints, including imperfect domain knowledge. We compare our method against multiple well-studied approaches on biological and synthetic datasets, showing improvements of up to 20% in F1-score over the best performing baseline in realistic settings.

Importantly, this chapter introduces notions of causality and model structure discovery, both of which are further developed in subsequent chapters. Chapter 6 addresses the complementary causal inference problem of estimating the effects of a single cause on an outcome for socio-behavioral problems where text modalities

are prominent forms of observational data. Chapter 7 returns to the problem of discovering graphical model structure but focuses on learning PSL models.

5.1 Causal Structure Discovery

The problem of causal structure discovery (CSD) consists of inferring a network of cause-and-effect relationships between many variables using observational data and domain knowledge. In contrast to the estimation of single causal relationships, CSD finds consistent causal graphs over all variables, exponentially increasing problem complexity. CSD is an important task for facilitating scientific discovery, such as determining regulatory networks amongst genes [47, 89] and understanding influences between atmospheric patterns to better forecast climate events [38].

Computational methods for causal structure discovery face several critical challenges. First, observational data is frequently noisy, containing spurious correlations between variables. Second, even with simplifying assumptions, CSD requires searching over exponentially many potential causal graphs, posing a scalability bottleneck. Finally, CSD requires incorporating heterogeneous domain knowledge of differing reliabilities, such as ontological and experimental evidence. Thus, successful CSD approaches must be robust, scalable, and flexible to succeed on real-world problems.

Existing methods for CSD have largely been evaluated in synthetic and low-noise settings that do not accurately represent the challenges of real-world domains. Traditional CSD approaches make locally greedy and iterative decisions, improving scalability at the cost of robustness. However, recent approaches based on logical satisfiability (SAT) [93] or linear programming (LP) [27] have shown the benefits of enforcing global constraints on the causal graph structure through joint inference.

In this chapter, we extend the joint inference view and propose a novel approach, CausPSL, that provides an attractive compromise between robustness to noise, scalability, and flexibility. We explore these trade-offs through extensive experimental evaluation on biological datasets, demonstrating significant performance gains on both real-world data and synthetic benchmarks. We formulate CSD as an inference problem by defining a joint probability distribution over causal graphs. Our approach defines this distribution by unifying constraints from statistical tests, side information, and domain knowledge. We implement CausPSL using the probabilistic soft logic (PSL) framework [7] which defines a hinge-loss Markov random field and supports efficient MAP inference. In experiments, we demonstrate several key strengths of CausPSL:

- **Robustness via Redundancy:** CausPSL exploits redundancy by using multiple statistical tests and soft constraints, mitigating noisy inputs.
- **Efficient Performance:** CausPSL scales to causal graphs with hundreds of variables via exact and efficient MAP inference.
- **Flexible Modeling:** CausPSL encodes both well-studied structural constraints and novel long- and short-range constraints with an easily extensible logical syntax.

We validate the features of CausPSL on realistic experimental settings including gene regulatory networks and protein signaling datasets, showing increases in F1-score of up to 20% over state-of-the-art CSD methods.

5.2 Preliminaries and Related Work

We focus on causal DAGs introduced by Pearl et al. [113], Pearl [111], Pearl and Verma [112]. To motivate modeling causal and ancestral structures using

independences in the data, we briefly review the foundations of conditional independences in DAGs. The two key concepts of d-separation and faithfulness, which are defined in Chapter 2, directly underpin constraint-based approaches to causal structure discovery. We contrast related work in constraint-based methods to our proposed approach.

5.2.1 Background on D-separation and Faithfulness

We recall that the causal DAG $\mathcal{G}^*$ encodes conditional independences in the data. The Markov condition, which states that each variable is conditionally independent of its non-descendants given its parents, directly identifies some conditional independences, and the DAG entails others. The graphical d-separation criteria determines the remaining conditional independences, as described in Chapter 2. Recall that V_i and V_j is d-separated by $\mathbf{Z_k}$ if all paths from V_i and V_j are blocked by $\mathbf{Z_k}$. We denote this relation $\mathcal{D}(V_i, V_j; \mathbf{Z_k})$. If $\langle \mathcal{G}, P \rangle$ are *faithful*, then $\mathcal{G}$ consists of all conditional independences in P and

$$\mathcal{D}(V_i, V_j; \mathbf{Z_k}) = V_i \perp\!\!\!\perp V_j | \mathbf{Z_k}$$

will hold for all i, j, k [72].

The faithfulness property forms the basis of constraint-based approaches for BN structure learning. Independence tests on the data are important as they constrain certain edge orientation along paths based on d-separation criteria. Marginal dependence relations identify adjacencies. Conditional dependencies are useful in distinguishing edge orientations, as described below. For example, consider path $A - B - C$ where undirected edges indicate associations between variables. We orient B as a collider with parents A and C if B is not in the conditioning set $\mathbf{Z}$ that makes A and C independent. Collider orientation follows

directly from criterion (2) above and is used in all constraint-based approaches.

5.2.2 Related Work on Constraint-based Approaches

Traditional constraint-based structure learning algorithms iteratively prune edges from a complete, undirected graph based on conditional independence tests of increasing set size between adjacent nodes and then iteratively apply rules based on d-separation and acyclicity to orient as many undirected edges as possible [136, 135, 25, 123]. The canonical PC algorithm assumes that no latent variables or confounders are present, and prunes edges without enforcing any consistency checks against conflicting independence tests [136]. Extensions of PC include Conservative PC (CPC) for avoiding edge orientations based on conflicting tests, Fast Causal Inference (FCI) for admitting latent variables, and order-independent PC for remaining robust to variable orderings used for iteration [123, 137, 135]. However, PC and its extensions remain sensitive to false positives and negatives since the algorithms enforce local rather global consistency and do not infer edges jointly.

The Max-Min Parents Children (MMPC) skeleton graph algorithm of [153] enforces consistency checks and conservatively selects association tests for use in edge-removal to prevent false positives and false negatives. However, MMPC still remains iterative and only outputs an undirected skeleton graph to be used downstream by score-based hill climbing algorithm [153]. In our probabilistic approach, we use multiple independence tests as input so that conflicting evidence does not rule out edges but only makes them less probable.

Constraint-based approaches can also easily be expressed using logic. The LoCI algorithm [24] is one such method that performs logical inference on clauses that represent d-separation constraints to discover causal structure. The COm-

bINE algorithm [152] casts causal structure discovery as an instance of satisfiability based on constraints generated from perfect interventions across overlapping sets of variables. Interventions elucidate ancestral relations since only ancestors can the change distribution of downstream variables. In contrast, we model imperfect and noisy interventions as evidence for ancestral edges which we leverage to infer direct causal relations using soft constraints.

Our work is most similar to constrained optimization approaches that formulate logical constraints from multiple, conflicting sets of conditional independence tests to identify causal or ancestral structure [63, 93, 62]. These approaches score edges by the number of independence statements they satisfy, weighted by the confidence scores of the statements. Conflicting independence tests are handled more robustly than traditional constraint-based methods. However, high confidence inputs will dominate without fully probabilistic reasoning. In our work, we directly encode a joint probability distribution over causal and ancestral edge predictions, combining both multiple independence tests and noisy evidence of ancestral relations to collectively propagate structural constraints between the two types of edges.

Score-based methods to CSD evaluate possible DAGs with penalized forms of likelihood. These approaches solve CSD efficiently by performing either greedy hill-climbing search [153, 22, 31] or constrained optimization using integer linear programs (ILP) [66, 27, 169, 10]. ILP methods can perform exact inference [10] but require constraints on the number of parents per variable, which are unknown or hard to justify in less understood biological domains.

5.3 Joint Probabilistic CSD

The input to causal structure discovery (CSD) is a set $\mathbf{V} = \{V_1 \dots V_n\}$ of n variables and m independent observations of $\mathbf{V}$. Here, we assume that the observations are drawn without selection bias or hidden confounders, as in PC and most score-based methods. The problem of CSD is to infer a directed acyclic graph (DAG) $\mathcal{G}^* = (\mathbf{V}, \mathbf{E})$ such that each edge $E_{ij} \in \mathbf{E}$ corresponds to V_i being a *direct cause* of V_j. If V_i is a direct cause of V_j, manipulating the value of V_i changes the marginal distribution of V_j. If V_i is an *ancestor* of V_k, there exists a directed path p, denoted by sequence of edges $V_i \to \cdots \to V_k$, from V_i to V_k. Ancestral structure is encoded by DAG $\mathcal{G}^*_{\mathcal{A}}$ where edges represent ancestral relations and correspond to the transitive closure of the causal graph $\mathcal{G}^*$. Typically, CSD methods output an equivalence class of $\mathcal{G}^*$ and $\mathcal{G}^*_{\mathcal{A}}$ that correspond to the optimal distribution.

The *joint probabilistic CSD* problem is to infer causal graph $\mathcal{G}^*$ together with the ancestral graph $\mathcal{G}^*_{\mathcal{A}}$. The problem requires defining a suitable joint meta-distribution $\mathcal{P}$ over the space of possible structures $\mathcal{G}$ and $\mathcal{G}_{\mathcal{A}}$. The inputs to $\mathcal{P}$ are random variables that capture structural and independence attributes of $\mathcal{G}$ and $\mathcal{G}_{\mathcal{A}}$. To avoid confusion with the random variables in our probabilistic model, henceforth, we refer to the domain variables $V \in \mathbf{V}$ as vertices.

$\mathbf{C}$ and $\mathbf{A}$ are the set of variables C_{ij} and A_{ij} for all V_i, V_j that denote the absence or presence of an ancestral or causal edge, respectively. The goal of inference is to find assignments for these variables. $\mathbf{U}$ is the set of observed variables U_{ij} associated with an undirected edge, or adjacency, from V_i to V_j for all V_i, V_j. $\mathbf{U}$ corresponds to the skeleton graph used in constraint-based methods. The set $\mathbf{M}$ of M_{ij} variables denotes marginal association between V_i and V_j where each M_{ij} is obtained by performing a statistical test of independence $V_i \perp\!\!\!\perp V_j$. Sim-

ilarly, $\mathbf{S_{ij}} = \{S_{ij}^{\mathbf{Z_1}} \dots S_{ij}^{\mathbf{Z_m}}\}$ denotes the set of variables that measure conditional association between V_i and V_j when conditioned on a non-empty subset of vertices $\mathbf{Z_k} \subset \mathbf{V}_{\setminus\{V_i,V_j\}}$. Each set $\mathbf{Z_k}$ has size between 1 and $|V|$ - 2. Each $S_{ij}^{\mathbf{Z_m}}$ corresponds to a statistical test for $V_i \perp\!\!\!\perp V_j|\mathbf{Z_m}$. Finally, we optionally observe $\mathbf{L} = \{L_{kl} \dots L_{st}\}$, local evidence that captures domain knowledge or side information about causal, ancestral or adjacency relations.

To solve this problem, the meta-distribution $\mathcal{P}(\mathbf{C}, \mathbf{A}|\mathbf{U}, \mathbf{S}, \mathbf{M}, \mathbf{L})$ is first fully defined. Then, we perform *maximum a posteriori* (MAP) inference over $\mathcal{P}$ to find an optimal joint assignment to variables $\mathbf{C}$ and $\mathbf{A}$.

5.4 CausPSL Approach

Defining meta-distribution $\mathcal{P}$ that relates $\mathcal{G}$ and $\mathcal{G}_{\mathcal{A}}$ requires a flexible modeling framework. To efficiently solve the joint probabilistic CSD problem, $\mathcal{P}$ must admit tractable inference. Our approach uses PSL, which offers both desired features.

5.4.1 CausPSL Model

CausPSL represents statistical tests, causal and ancestral relations as predicates to form orientation constraints in a HL-MRF using the rules shown in Table 5.1.

Predicates

The targets of joint probabilistic inference, C_{ij} and A_{ij}, are represented with predicates Causes(A, B) and Anc(A, B). We represent undirected edges U_{AB} with Adj(A, B).

We introduce Assoc(A, B) and Indep(A, B) to capture marginal association

Rule Type	Rules
Causal Orientation	C1) $\neg\textsc{Adj}(A,B) \rightarrow \neg\textsc{Causes}(A,B)$
	C2) $\textsc{Causes}(A,B) \rightarrow \neg\textsc{Causes}(B,A)$
	C3) $\textsc{Adj}(A,B) \wedge \textsc{Adj}(C,B) \wedge \neg\textsc{Adj}(A,C) \wedge \textsc{CondAssoc}(A,C,S) \wedge \textsc{InSet}(B,S) \rightarrow \textsc{Causes}(A,B)$
	C4) $\textsc{Adj}(A,B) \wedge \textsc{Adj}(C,B) \wedge \neg\textsc{Adj}(A,C) \wedge \textsc{CondAssoc}(A,C,S) \wedge \textsc{InSet}(B,S) \rightarrow \textsc{Causes}(C,B)$
	C5) $\textsc{Causes}(A,B) \wedge \textsc{Assoc}(A,C) \wedge \textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{Adj}(B,C) \rightarrow \textsc{Causes}(B,C)$
	C6) $\textsc{Causes}(A,B) \wedge \textsc{Causes}(B,C) \wedge \textsc{Adj}(A,C) \rightarrow \textsc{Causes}(A,C)$
Basic Joint Rules	J1) $\textsc{Causes}(A,B) \rightarrow \textsc{Anc}(A,B)$
	J2) $\neg\textsc{Anc}(A,B) \rightarrow \neg\textsc{Causes}(A,B)$
	J3) $\textsc{Anc}(A,B) \wedge \textsc{Anc}(B,C) \rightarrow \textsc{Anc}(A,C)$
	J4) $\textsc{Anc}(A,B) \wedge \textsc{Adj}(A,B) \rightarrow \textsc{Causes}(A,B)$
	J5) $\textsc{Adj}(A,B) \wedge \textsc{Adj}(B,C) \wedge \textsc{Assoc}(A,C) \wedge \textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{Causes}(B,A) \wedge \neg\textsc{Anc}(C,A) \rightarrow \textsc{Causes}(B,C)$
Ancestral Orientation	A1) $\textsc{Indep}(A,B) \rightarrow \neg\textsc{Anc}(A,B)$
	A2) $\textsc{Anc}(A,B) \rightarrow \neg\textsc{Anc}(B,A)$
	A3) $\textsc{Indep}(A,C) \wedge \textsc{CondAssoc}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{HasSize}(S,1) \rightarrow \neg\textsc{Anc}(B,A)$
	A4) $\textsc{Indep}(A,C) \wedge \textsc{CondAssoc}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{HasSize}(S,1) \rightarrow \neg\textsc{Anc}(B,C)$
	A5) $\textsc{Assoc}(A,C) \wedge \textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{HasSize}(S,1) \wedge \textsc{Anc}(B,C) \wedge \textsc{Anc}(B,A) \rightarrow \neg\textsc{Anc}(A,C)$
	A6) $\textsc{Assoc}(A,C) \wedge \textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{HasSize}(S,1) \wedge \textsc{Anc}(A,B) \wedge \textsc{Anc}(B,C) \rightarrow \textsc{Anc}(A,C)$
	A7) $\textsc{Assoc}(A,C) \wedge \textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \textsc{HasSize}(S,1) \wedge \textsc{Anc}(C,B) \wedge \textsc{Anc}(B,A) \rightarrow \textsc{Anc}(C,A)$
	A8) $\textsc{CondIndep}(A,C,S) \wedge \textsc{InSet}(B,S) \wedge \neg\textsc{Anc}(A,B) \wedge \textsc{HasSize}(S,1) \rightarrow \neg\textsc{Anc}(A,C)$

Table 5.1: PSL rules for causal and ancestral structure inference.

and independence, corresponding to M_{AB}. To denote conditional association and independence, we introduce $\textsc{CondAssoc}(A,B,S)$ and $\textsc{CondIndep}(A,B,S)$. S will be substituted with all possible conditioning sets $\mathbf{Z_m}$. These logical atoms correspond to the $\mathbf{S_{AB}}$. To obtain substitutions for these predicates, we enumerate pairwise marginal and conditional tests with all possible conditioning sets up to a maximum size. We threshold p values from statistical tests to determine whether independence statements are characterized as $\textsc{Assoc}, \textsc{CondAssoc}$ or $\textsc{Indep}, \textsc{CondIndep}$. We use $1-p$ as truth values for $\textsc{CondAssoc}$, $\textsc{Assoc}$ and p for $\textsc{CondIndep}$, $\textsc{Indep}$. Since adjacencies imply dependence between variables, we obtain $\textsc{Adj}(A,B)$ by retaining $\textsc{Assoc}(A,B)$ observations that are never conditionally independent. Finally, because orientation constraints require membership checks in conditioning sets S, we use auxiliary predicate $\textsc{InSet}(C,S)$ to indicate that vertex C is in conditioning set S.

$\textsc{Local}_\lambda(A, B)$ predicates denote evidence from source λ for causal, ancestral or undirected edge between vertices A and B and correspond to variables $\mathbf{L}$. Obtaining local evidence is domain-specific, and in our experimental evaluation, we show applications of both intervention-based and other side information.

Soft Constraints

The constraints which we fuse in our proposed approach arise from the graphical d-separation criteria presented in earlier sections. Broadly, these criteria provide a correspondence between the observed conditional independences in the data to valid paths in causal graphs. Table 5.1 shows the rules used in CausPSL. The causal orientation rules (C1-C6) follow from the three sound and complete PC rules [136] and the ancestral orientation rules (A1-A8) are derived from constraints used in the SAT-based ancestral causal inference (ACI) algorithm [93]. The basic joint rules (J1-J5) connect ancestral and causal edge predictions through fundamental relationships between the structures introduced in Section 5.3. These multiple types of well-studied constraints propagate consistency across predictions for CausPSL.

Causal Orientation Rules Rule C1 discourages causal edges between vertices that are not adjacent. Rule C2 penalizes simple cycles between two vertices. The remaining rules ensure that observed independences match those implied by the graph through d-separation. Rules C3 and C4 correspond to the PC rule which orients chain $V_i - V_j - V_k$ as $V_i \rightarrow V_j \leftarrow V_k$ if conditioning on V_j breaks the independence between V_i and V_k. Unlike in PC, in CausPSL, V_j appears in multiple conditioning sets. The redundancy recovers information when V_j is incorrectly missing from a separating set. Rule C5 captures the PC rule that orients path $V_i \rightarrow V_j - V_k$ as $V_i \rightarrow V_j \rightarrow V_k$ when $V_i \rightarrow V_j$ is probable and V_j

induces conditional independence between V_i and V_k. Rule C6 maps to the final PC rule, and if $V_i \to V_j \to V_k$ and $V_i - V_k$, orients $V_i \to V_k$ to avoid a cycle. PC applies these rules iteratively to fix edges whereas in CausPSL, the rules induce dependencies between causal edges to encourage parsimonious joint inferences.

Basic Joint Rules Rule J1 encodes that causal edges are also ancestral by definition and rule J2 is its contrapositive. Rule J3 encodes transitivity of ancestral edges, encouraging consistency across predictions. Rule J4 infers causal edges between probable ancestral edges that are adjacent. These four rules exactly encode the relationship between causal and ancestral graphs, and suffice to recover structure under perfect inputs. However, in noisy settings, we gain robustness by including additional joint constraints such as rule J5 and ancestral rules below to recover consistent explanations from conflicting inputs. Rule J5 orients chain $V_i - V_j - V_k$ as a diverging path $V_i \leftarrow V_j \to V_k$ when V_k is not likely an ancestor of V_i. Without ancestral constraints, statistical tests alone cannot distinguish between diverging and linear paths.

Ancestral Orientation Rules Ancestral rules A1 and A2 are analogous to their causal orientation counterparts. Rules A3 to A7 follow from lemmas relating minimal conditional (in)dependence to the existence or absence of ancestral edges [93, 24]. Minimal conditional independence is defined as $(X \perp\!\!\!\perp Y|\mathbf{W} \cup Z) \wedge \neg(X \perp\!\!\!\perp Y|\mathbf{W})$ and corresponds to ancestral edge existence between Z and X or Y. Similarly, minimal conditional dependence is $\neg(X \perp\!\!\!\perp Y|\mathbf{W} \cup Z) \wedge (X \perp\!\!\!\perp Y|\mathbf{W})$ and denotes ancestral edge absence between Z, and X and Y. For compactness, we encode minimal conditional (in)dependence by only comparing marginal associations to conditional tests of set size one. We model ancestral edge existence with three rules, A5 to A7, for each path orientation case: 1) $\leftarrow Z \to$ where Z is diverg-

ing, 2) $\leftarrow Z \leftarrow$ where Z is along linear path from X to Y, and 3) $\rightarrow Z \rightarrow$ where Z is along a linear path in the opposite direction. Rule A8 translates the first novel ancestral rule introduced in ACI [93]. Rules A5 to A8 introduce dependencies across ancestral edge predictions, requiring collective inferences.

5.5 Experimental Results

Our evaluation demonstrates three advantages of our method: the flexibility of combining multiple structural constraints, scalability for large causal networks, and robustness to noise.[1] We evaluate our model on standard synthetic data [63, 62, 93] and two real-world biological datasets. We compare against PC [136], the canonical constraint-based CSD method and Max-Min Hill Climbing (MMHC), a score-based hybrid approach that uses the max-min parents children (MMPC) graph pruning algorithm and has achieved state-of-the-art performance in multiple BN structure learning domains [153]. We also include comparisons against a bootstrapped variant of PC commonly used to improve robustness [122, 93]. In our experiments, scalability prevents us from comparing against the SAT-based CSD approach [63], which becomes prohibitively expensive for domains larger than eight variables.

Dataset	PC	MMHC	Bootstrapped PC
Synth	0.74 ± 0.09	0.76 ± 0.12	0.72 ± 0.11
DREAM20	0.15 ± 0.04	0.17 ± 0.05	0.18 ± 0.05
DREAM30	0.16 ± 0.03	0.2 ± 0.05	0.16 ± 0.04

Table 5.2: Average F_1 scores of methods across compared baselines.

[1] Code and data at: bitbucket.org/linqs/causpsl.

Dataset	CausPSL-PC	CausPSL-Joint	CausPSL-Anc	CausPSL
Synth	0.87 ± 0.06	0.87 ± 0.06	0.86 ± 0.06	**0.87 ± 0.06**
DREAM20	0.17 ± 0.05	0.18 ± 0.05	0.19 ± 0.05	**0.20 ± 0.05**
DREAM30	0.22 ± 0.03	0.23 ± 0.03	**0.24 ± 0.03**	0.22 ± 0.03

Table 5.3: Average F_1 scores of methods across variants of CausPSL. We show how each CausPSL component contributes to performance.

5.5.1 Datasets

We validate our approach using three datasets: (1) synthetic linear acyclic models with Gaussian noise; (2) simulated gene expression from the DREAM4 challenge [94, 120]; (3) perturbation experiments on protein-signaling pathways [130].

Synthetic data

To generate synthetic observations, as in previous work [63, 93, 62], we randomly generate 100 ground truth DAGs over 15 variables with edge probability of 0.2 using the `pcalg` package. We sample 500 observations from each using a linear Gaussian model. CSD methods typically evaluate on this low-noise synthetic setting which serves as a contrast to the more realistic noisy settings described below.

DREAM4 Challenge

Our second dataset from the DREAM4 challenge consists of a gold-standard yeast transcriptional regulatory network and simulated gene expression measurements [94, 120]. For cross validation, we sample 10 subnetworks of sizes 20 and 30, denoted DREAM20 and DREAM30, with low Jaccard overlap. The real-valued gene expression measurements are simulated from differential equation models of the system at 210 time points. We perform independence tests on the mea-

surements which yield numerous spurious correlations. Additionally, we include domain knowledge of undirected protein-protein interaction (PPI) edges modeled by $\textsc{Anc}(A, B) \wedge \textsc{Local}_{\text{PPI}}(A, B) \rightarrow \textsc{Causes}(A, B)$.

Protein Signaling Pathway in Human T-Cells

Our third dataset comes from a protein-signaling pathway in human T-cells with flow cytometry measurements [130]. The discovered protein signaling network has been biologically validated and used extensively as a benchmark for evaluating CSD algorithms [152, 93, 101, 37, 117]. The variables are abundance levels of 11 molecules, measured across eight experimental conditions with 700 to 900 observations each. The first condition activates the pathway and is considered by previous work as the steady-state observed data. The remaining conditions are interventions on seven out of 11 proteins. Following prior work, we consider statistically significant ($\alpha = 0.05$) post-interventional changes as evidence of an ancestral relation between the intervention target and effected protein [93, 130]. We model this intervention-based local evidence as $\textsc{Local}_{\text{Intervention}}(A, B) \rightarrow \textsc{Ancestor}(A, B)$.

5.5.2 Experimental Setup

To evaluate the result quality across methods and robustness to noise, we compute F_1 scores of predicted causal edges against the ground truth edges from each dataset. To calculate F_1 in DREAM and synthetic settings, rounding thresholds on the continuous outputs of CausPSL and Bootstrapped PC are selected using cross-validation with 10 and 100 folds, respectively. In the Sachs setting where the small network size prevents sampling of subnetworks for cross-validation, a standard 0.5 threshold is used. For independence tests in all settings, we use lin-

ear and partial correlations with Fisher's Z transformation for continuous data. We run both PC variants and MMHC with the `pcalg` and `bnlearn` R packages, respectively. CausPSL uses ADMM inference implemented in PSL [7]. Without a priori preference for rules, we set all CausPSL rule weights to 5.0 except for causal and ancestral orientation rules 2 which are set to 10.0, since they encode strong asymmetry constraints. For both PC variants and CausPSL, we condition on sets up to size two for DREAM20 and up to size one for DREAM30. The MMPC phase of MMHC performs tests on sets up to size $|V| - 2$. For Bootstrapped PC, we follow the bootstrapping procedure used by [93] and randomly sample 50% of the observations to include in 100 iteration of PC and average the predictions across multiple runs. In DREAM and synthetic settings, α thresholds on independence tests for all methods are also selected within the cross-validation framework. Baselines use α to prune undirected edges while CausPSL uses separate α values to categorize association tests and identify Adj. Since α is typically small, we rescale truth values p for CondIndep, Indep by $\sqrt[3]{p}$ to reduce right-skewness of values. For Sachs, we use $\alpha = 0.05$ for all methods, which has been reported to have the best performance in prior work. We rescale p-values of the post-interventional changes with the sigmoid function to prevent overconfident local evidence.

5.5.3 Cross-validation Study of Modeling Components

We first investigate how each type of constraint in CausPSL bolsters performance. CausPSL has three critical modeling components that contribute in differing degrees to improvements in CSD: CausPSL-PC, CausPSL-Joint, and CausPSL-Anc. CausPSL-PC uses only the causal orientation rules and upgrades PC with multiple independence tests and collective inferences. The

CausPSL-Joint model combines CausPSL-PC and basic joint rules for longer-range structural consistency but excludes full ancestral modeling. The CausPSL-Anc model extends CausPSL-Joint with ancestral orientation rules. Finally, we distinguish between CausPSL-Anc and the complete CausPSL model, which includes the novel ACI constraint [93]. To understand the factors affecting result quality, we perform cross-validation across the model variants of CausPSL and compare against both PC variants and MMHC in the DREAM4 and synthetic settings. Table 5.2 shows average F_1 scores across the compared baseline methods. Table 5.3 shows average F_1 scores across all variants of CausPSL.

CausPSL-PC alone outperforms PC in all settings, with significant gains over both PC variants in two. These improvements suggest that collective inference and multiple statistical tests without pruning alone provide robustness benefits, even over bootstrapping the PC algorithm. CausPSL-Joint outperforms CausPSL-PC in two of three settings, suggesting that modeling even transitivity and short-range dependencies between ancestral and causal structures improves performance. CausPSL-Anc and CausPSL further gain over CausPSL-Joint in two of three settings. CausPSL achieves the best performance in DREAM20 with significant gains over MMHC and PC. CausPSL-Anc outperforms all methods in DREAM30 with gains of up to 50% over both PC variants and 20% over MMHC. Our best performing PSL models significantly outperform multiple baselines using a paired t-test on DREAM, showing the benefit of more sophisticated ancestral-causal constraints under noisy experimental conditions, where spurious correlations dominate. On straightforward linear Gaussian data, all modeling variants of CausPSL significantly outperform both PC variants and MMHC with F_1 score improvements of up to 17.5%. However, in this synthetic setting, simpler CausPSL-PC and CausPSL-Joint models suffice for

D	Size	PC	MMHC	PSL;C=1		PSL;C=2	
				CI	Inf	CI	Inf
Synth	10	0.02	0.01	0.07	0.19	0.35	0.23
	20	0.06	0.03	0.93	0.65	19.7	1.11
	30	0.19	0.15	4.94	1.55	684	8.91
	50	0.44	0.48	65.4	6.99	440k	159
DREAM4	10	0.03	0.02	0.06	0.09	0.3	0.19
	20	0.08	0.06	0.73	0.37	14.3	3.12
	30	0.22	0.15	3.76	1.5	433	30.2
	50	0.41	0.49	57.1	9.96	437k	425

Table 5.4: Running times in seconds for obtaining conditional independence tests (CI) and inference (Inf). CausPSL scales to large networks using multiple tests with no pruning.

good performance. The contrasting result highlights the importance of evaluating CSD methods on more realistic settings.

5.5.4 Comparisons in Real-World Sachs Setting

In the real-world Sachs setting, we compare the F_1 scores of causal edge predictions by CausPSL-Anc and CausPSL against those of MMHC, the best performing baseline method. Additionally, we compare our ancestral edge predictions to ACI results reported by [93]. CausPSL-Anc improves over MMHC from 0.307 to 0.32 F_1 while CausPSL performs as well as MMHC. For ancestral inference, ACI achieves a reported F_1 score of 0.38. CausPSL-Anc gains over ACI with an F_1 of 0.43 and CausPSL also improves over ACI with a score of 0.4.

5.5.5 Scalability

Our second evaluation focuses on the scalability of our approach. PC and MMHC scale by iteratively pruning adjacencies with statistical tests, potentially sacrificing result quality despite permitting larger conditioning set sizes. More

flexible SAT-based methods enumerate all statistical tests but cannot scale to large networks. For example, running the SAT approach proposed by [63] with nine variables and a conditioning set size of one required over 40 minutes [93]. In contrast, CausPSL uses all statistical tests without pruning and requires less than a second for 10 variables, overcoming the inference scalability bottleneck. To evaluate running times, we generate synthetic linear Gaussian networks and sample DREAM4 subnetworks of increasing size. Our method computes all possible statistical tests up to conditioning set size denoted by C and the baseline methods prune conditioning sets through independence. In Table 5.4, we present running times for all methods, splitting up our approach into conditional independence testing (CI) and inference (Inf). We show that CausPSL can efficiently infer causal graphs while using more information than competing methods.

The running time depends on the network size n and the maximum conditioning set size C. The results indicate that the dominant factor in the running time of our method is enumerating all statistical tests rather than inference. For the largest networks ($n = 50, C = 1$), computing statistical tests requires approximately a minute, while inference only requires 7 to 10 seconds. Larger conditioning sets impact running time, requiring up to 10 minutes when $n = 30, C = 2$. However, Table 5.3 shows that by enumerating statistical tests, CausPSL outperforms pruning-based methods with only $C = 1$. SAT-based methods also enjoy this benefit [93] but require expensive inference. In contrast, CausPSL completes inference within 10 seconds for 30- and 50-variable networks when $C = 1$. In further study, CausPSL completed inference for a DREAM4 network with 100 variables in 27 minutes, scaling to an order of greater magnitude than SAT-based methods. In future work, statistical tests can be parallelized to admit larger C.

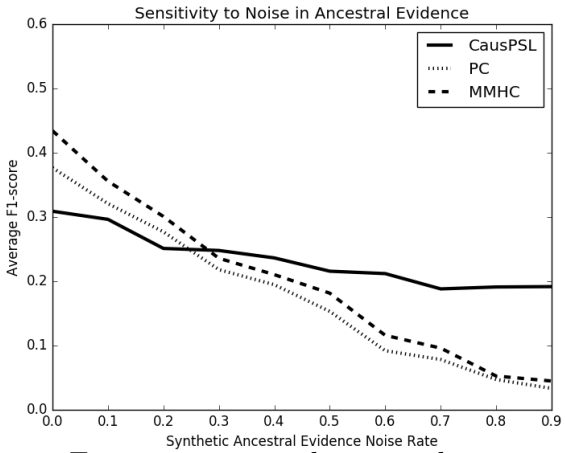


Figure 5.1: Average F_1 score vs. synthetic evidence noise rate on DREAM4 ($n = 30, C = 1$). CausPSL remains robust as noise rate increases.

5.5.6 Robustness to Noisy Evidence

In our final evaluation, we validate the robustness of CausPSL to imperfect evidence. CausPSL incorporates real-valued noisy signals within joint inference, exploiting global structural constraints to smooth local errors. In contrast, MMHC and PC must discretize noisy evidence and incorporate domain knowledge as fixed edges or non-edges.

To evaluate the robustness of CSD methods on DREAM30 subnetworks, we simulate noise with a new local ancestral signal drawn by fixing a Bernoulli error rate and sampling real-valued evidence from its conjugate, a β distribution. We set a Bernoulli error rate of $1-p$. For each pair of vertices, with probability p, true ancestral edges are sampled from $\beta(8, 2)$, and true non-edges are sampled from $\beta(2, 8)$ which are peaked at high-confidence and accurate soft truth values. With probability $1 - p$, incorrect values are sampled from $\beta(2, 5)$ and $\beta(5, 2)$ for edges and non-edges, respectively. For CausPSL, we incorporate this new signal using the local ancestral evidence rule shown in the Sachs setting. For MMHC and PC, synthetic values from this signal of < 0.5 are treated as fixed causal non-

edges, representing the hard version of joint rule J3 in Table 5.1. Synthetic values ≥ 0.5 are intersected with PPI edges to obtain fixed causal edges, simulating the discretized version of the PPI rule given in the DREAM setting.

In Fig. 5.1, we compare average F_1 scores across all modified methods as the Bernoulli error rate of the synthetic signal increases from 0.0 to 0.9. CausPSL remains robust as the error increases beyond 0.3 while PC and MMHC steadily degrade in performance. When the signal is near-perfect with error ≤ 0.2, the baselines receive select correct causal edges while CausPSL fuses the signal with imperfect statistical tests. However, analysis of intervention-based evidence in the Sachs setting shows that real-world local signals are in the ≥ 0.5 noise regime, where CausPSL excels over compared methods.

5.6 Discussion

We propose a probabilistic model for the CSD problem that achieves scalability despite using multiple independence tests and global structural constraints. Our method is flexible, fusing noisy ancestral and causal signals with side information from PPI networks and interventions. Our experimental highlights include: 1) scaling up to networks with hundreds of variables; 2) achieving significant performance gains over constraint- and score-based baselines despite many spurious correlations; and 3) showing robustness to increasingly noisy local signals. In future work, we will extend our approach to support latent variables and perform approximate marginal inference to score possible causal and ancestral edges.

Chapter 6

Estimating Causality in Text

Chapter 5 underscores the importance of discovering causal knowledge, focusing on inferring DAGs that capture networks of cause-and-effect relationships. This chapter presents a complementary viewpoint, studying the estimation of a single causal effect on outcomes of interest. Particularly, we focus on socio-behavioral phenomena where observational data, as outlined in Chapter 1, is obtained from digital sources such as social media or mood/activity logging platforms. In contrast to the previous chapter that focused on conventional forms of observational data such as gene expression measurements, in this chapter, we consider including textual data in causal inference. This chapter is divided into two sections to study socio-behavioral causal inference problems that require fusing varying degrees of textual information. The first section focuses on estimating the effects of exercise on mood from a recently proposed activity and mood tracking application that combines both text and measurements. This first task allows us to understand the ramifications of combining traditional forms of observational data such as variable measurements with text observations. The second section again studies online debate threads to understand the causal impact of reply tone on users' sentiment. This dialogue analysis relies only on text data, requiring new

methodologies for adapting causal inference to textual data.

Section 6.1 explores the behavioral domain of understanding mood through platforms that enable users to track their behaviors. Mood and activity logging applications empower users to monitor their daily well-being and make informed health choices. To provide users with useful feedback that can improve quality of life, a critical task is understanding the causal effects of daily activities on mood and other wellness markers. In this section, we analyze observational data from EmotiCal, a recently developed mood-logging web application, to explore the effects of exercise on mood. Since the causal link between exercise and mood is validated in literature, this study allows us to develop and evaluate approaches for causal inference. To develop a robust methodology for estimating the average treatment effect (ATE), an important estimator, from heterogeneous user data, we outline and investigate three important modeling questions about: 1) filtering or stratifying strategies on users to eliminate implicit confounds or outliers; 2) performing the analysis per-user or per-entry; and 3) including text sources to estimate the ATE. The question of aggregating information at the user level follows from Chapter 3, and we again find that selecting modeling granularity affects our analysis. With these modeling strategies, we tackle causal inference when only textual data is present.

Section 6.5 investigates the causal effect of reply tone on user sentiment change in online debate forums. Going beyond stance and disagreement prediction, this inference question aims to develop a deeper understanding of dialogue strategies and their effectiveness in invoking certain responses from others. In contrast to the previous section, to perform causal inference in this setting, we require extracting variables from text to encode treatment, outcome and confounders. After identifying variables, estimating causal effects with existing methods such as propensity

score matching is not straightforward and poses modeling choices, as above. In this section, we explore: 1) formulating the causal estimation problem in the context of discussions threads; 2) modeling the propensity score from users' posts; and 3) proposing a structured method for directly modeling treatment, outcome and confounders in this relational data graph. We evaluate these three imperatives on the previously studied 4Forums domain and study the effect of multiple styles of replies on the change in users' sentiment. Our empirical findings again highlight the importance of careful consideration of modeling choices, especially in textual data and demonstrate the benefits of our novel structured model for estimating the ATE.

6.1 Causal Effects of Exercise on Mood

Mood and activity logging applications play an important role in the larger, emerging trend of technologies that empower users to monitor and improve their quality of life [84, 110, 88]. Notable platforms include Fitbit or Strava for exercise tracking, and Moodscape or Echo for reflection on emotion and mood [84, 88]. In mood logging applications, users track their activities together with markers of their mental state to promote psychological well-being. To facilitate positive outcomes, these applications must provide actionable feedback on how factors in users' daily life affect their mood. An important step to generating feedback is understanding the causal effects of these factors on mood, estimated by the *average treatment effect* (ATE). Estimating ATE requires several modeling assumptions and careful choices, especially on complex user-behavior data.

Prior approaches to understanding factors that affect mood rely on traditional methods such as surveys and randomized intervention trials [146, 131]. The findings suggest a link of exercise and socializing on mood. In recent years, mood

logging applications such as Echo, Moodscape and iHappy have driven empirical research in user behavior [64, 84, 88, 110]. Some platforms perform direct interventions such as recommending activities to improve users' mood [110], while predictive applications like Moodscape infer mood changes from activity patterns.

Recently, a different line of work focuses on Twitter social media posts to find causal links on outcomes that span mood or emotion to significant life milestones [108, 35]. These approaches extract treatments such as exercising behavior and potential outcomes such as mood from tweets, and perform matching on text to eliminate confounds. Despite using non-conventional forms of observational data, both studies report findings validated in literature, such as the exercise and mood link.

Our work focuses on causal estimation using a unique mood logging application, EmotiCal (*Emoti*onal *Cal*endar), that features both recorded values for daily activities and mood, and text descriptions from users [58, 138]. Motivated by promising results from the complementary studies of social media sites and task-specific mood logging platforms, we study the causal effect of exercise on mood by combining text and observational data. The link between exercise and mood is well-validated in literature, providing a benchmark for our analysis. To develop a robust methodology for estimating ATE from heterogeneous user data, we outline and investigate three important modeling questions in this paper:

1. What filtering or stratifying strategies on users are necessary to eliminate implicit confounds or outliers?

2. Should we perform our analysis per-user or treat all logged user entries as independent units of study?

3. What impact does including text sources in our analysis have on the estimated ATE?

Our findings highlight the importance of each modeling choice, and suggest that per-user analysis and incorporating text provide stronger causal results. We illustrate our empirical results with useful qualitative examples.

6.2 Dataset

We obtain our dataset from EmotiCal (Emotional Calendar), an application created to help people regulate and improve their mood and well-being [58, 138] [1]. EmotiCal users were asked to use the application at least twice a day, logging an entry each time. These entries consist of users' current mood, energy level, and up to 14 trigger activities that users believe have influenced their mood. For example, users can log social interactions (e.g., time spent with a friend or coworker), aspects of physical health (e.g., sleep or exercise), and work activities (e.g. meetings) to track these activities' effects on mood. EmotiCal also prompts users to generate short textual explanations of how and why they think those activities have affected their mood.

Figure 6.1 shows the EmotiCal user interface for logging mood and energy levels (left panel), and activities that affect these factors (right panel). To create a mood entry, users first make a simple mood valence and strength decision, choosing a mood ranging from -3 (very negative) to +3 (very positive). Users also recorded energy levels ranging from -3 (low energy) to +3 (high energy). After selecting mood, users engage in active mood analysis. Users identify which of 14 possible trigger activities influenced their mood and rate that influence on a scale of -2 (negatively impacted mood) to +2 (positively impacted mood).

Users choose as many activities as they deem relevant, although most users

[1]The data collection process has IRB approval. All participants were directly informed of the research uses of their anonymized data and allowed to exclude private items from the final dataset. Strict procedures are in place to secure and protect users' privacy.

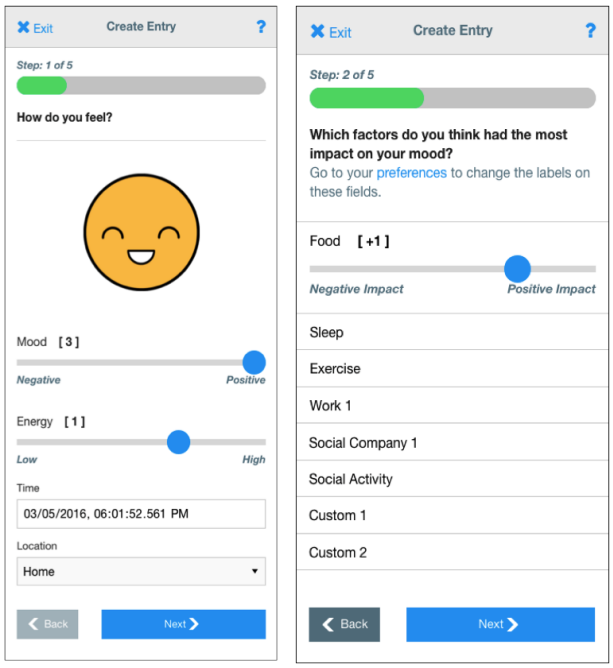


Figure 6.1: EmotiCal System Components. The left screen shows the logging of mood and energy levels. The right screen shows the logging of different activities which affected the user's mood

choose relatively few per entry. Eight of these 14 trigger activities are constant across users: food, sleep, exercise, social activity, work, leisure, mood, and social company; the other 6 categories are customizable, allowing users to record triggers that are unique to their lives. After logging their trigger activities, users write short textual entries about the factors that affected their mood.

In this work, we focus on the eight activities listed above which are consistent across users. In addition, we use the textual entries users wrote to improve the significance of our results. In total, the EmotiCal dataset consists of 6344 entries from 143 unique users. The EmotiCal data enjoys two important advantages over typical user-behavior modeling datasets: 1) participants provide real-time and longitudinal self-labels for all attributes without need for crowd-sourced annotations; and 2) users log their own perception of how activities influence mood and also include textual information, producing a reliable dataset.

6.3 Problem Statement

We consider units $X = \{x_1, \ldots, x_n\}$ where each logged entry $x_i = (v_1, \ldots v_8, t)$ includes measurements for the eight specified variables v_i and text denoted t. The treatment assignment for x_i is 1 if $v_{\text{exercise}} \geq 1$ and 0 otherwise, and is denoted by random variable T_i. For each x_i, we observe only the mood outcome under treatment $Y_1(x_i)$ when $T_i = 1$ or the outcome under control $Y_0(x_i)$ when $T_i = 0$. The goal is to estimate the average treatment effect (ATE), defined as:

$$ATE = \mathbb{E}_{p(x)}\Big[\mathbb{E}_{p(Y_1|x)}[Y_1(x_i)] - \mathbb{E}_{p(Y_0|x)}[Y_0(x_i)]\Big] \tag{6.1}$$

The expectation requires unobservable outcomes $Y_{1-t_i}(x_i)$, called counterfactuals. One well-studied approach to estimating ATE from this incomplete observed data alone is to perform matching [129]. The goal of matching is to pair every unit x_i with another unit x_j that has the opposite treatment assignment, i.e. $T_j = 1 - T_i$. For example, the match for a treated entry where the user records having exercised is a control entry where potentially a different user does not record exercise activity. The selected match $x_j = \arg\min_{x_k \in X} d(x_i, x_k)$ should be the nearest-neighbor of x_i according to a distance measure $d(\cdot)$. Since for every entry, we only observe the outcome under a given treatment assignment, matching estimates the difference in outcomes by comparing against a unit which similar in every other aspect except the treatment assignment. In our problem, for example, we expect to find matched entries that record similar values for the other activities besides exercise such as sleep or food. The matching produces a set of pairs $M = \{(x_i, x_j) | x_i, x_j \in X, T_i \neq T_j\}$ to estimate ATE as:

$$\hat{ATE} = \frac{1}{|M|} \sum_{m \in M} (2T_i - 1)[Y_{T_i}(x_i) - Y_{T_j}(x_j)] \tag{6.2}$$

The summand is shorthand to indicate correct subtraction order, depending on whether $T_i = 1$ or not. In our analysis, we investigate key experimental choices for estimating ATE with user-behavioral data.

6.4 Experimental Results

The goal of our analysis is to investigate three modeling questions to develop a methodology for estimating treatment effects from mood logging data:

- Q1: Which steps to filter users help to control for implicit confounds and eliminate outliers in causal estimation from behavioral data?
- Q2: Should we aggregate the ATE estimate over treatment and control entries matched per-user, treating users as the key units of study, or should we treat entries as the unit of study?
- Q3: What signal does textual data contain to help us control for additional implicit confounding?

To validate key modeling decisions, we focus on estimating the ATE of exercise on mood, a link that has been well-studied in literature and found to have a significant positive effect. Studying the exercise-mood link allows us to interpret differences in ATE across the experimental conditions we use as better or worse performance. Our findings below suggest that filtering users and matching entries per-user are important, and highlight the benefits of incorporating textual data sources.

6.4.1 Experimental Setup

For all experimental questions, we perform matching on treated units to obtain their nearest control unit. For Q1 and Q2, the metric used for matching is Euclidean distance over the eight other measured variables. We introduce a text-based propensity score matching [129, 127] technique when we examine Q3. We also perform a Z-test to compare the mean mood across treatment and control samples to understand the significance of the effect. We introduce experimental conditions to evaluate each question that we investigate. Table 6.1 shows the ATE and hypothesis test p-value results for all conditions, and we provide a detailed discussion of these results below.

Table 6.1: ATE and hypothesis testing results for experimental conditions across evaluation questions Q1 to Q3. The results suggest benefits to including textual data in matching methods.

Condition	ATE	Hypothesis Test P-value
Baseline	0.26	$3 \times 10^{-}5$
Filtered	0.31	$1.5 \times 10^{-}6$
User	0.49	$2 \times 10^{-}12$
Text, C=0.9	0.53	$6.7 \times 10^{-}13$
Text, C=0.01	0.61	0.0

Table 6.2: p-values from T-tests evaluating balance of other measured activities across control and treated groups. We compare the balance between three matching strategies. Text, C=0.9 improves balance over the User matching for three covariates.

Covariate	User	Text, C=0.01	Text, C=0.9
Food	0.058	$1.3 \times 10^{-}4$	0.023
Sleep	0.012	$3.8 \times 10^{-}5$	0.549
Work	0.163	0.017	0.586
Leisure	0.025	$6.1 \times 10^{-}5$	0.005
Social Company	0.072	0.014	0.068
Social Activity	0.437	0.005	0.649

6.4.2 Q1: Filtering Users

For our first question Q1, we investigate whether filtering a subset of the users to consider in our analysis is necessary to mitigate noise in the estimates. Our baseline condition, which we call BASELINE, includes all users who report the effects of exercise at least once. The BASELINE condition retains 114 users out of 143 total users. To compare against BASELINE, we consider whether to target our study to the 73 users of EmotiCal who report having exercised at least 3 times. We refer to this experimental condition as FILTERED. Users that report exercise as a factor that effected mood only once may not consistently value this activity, introducing noise to the causal estimation.

Table 6.1 shows that the ATE estimated through the BASELINE condition is significant. This result validates a previous regression analysis on EmotiCal data that showed significant correlations between exercise and mood [138]. However, the FILTERED condition increases both the ATE and its significance, which we expect to see for the well-validated causal link between exercise and mood. This comparison shows the importance of excluding the users for whom exercise rarely plays a role in affecting their mood. The gains in ATE significance may be due to comparing users that are more similar in their proclivity for exercise.

6.4.3 Q2: User-specific Matching

Following findings from Q1, we adopt the FILTERED condition throughout the analysis. Our next critical question is whether to only match entries of the same user, while still aggregating across all users' matched entries when estimating the ATE. To evaluate this, we introduce condition USER which adds a constraint to the matching algorithm that matched entries must be from the same user.

Formally, U_i is the user of entry x_i and the modified matched pairs are:

$$M_U = \{(x_i, x_j) | x_i, x_j \in X, T_i \neq T_j, U_i = U_j\}$$

The compute ATE given by equation 6.2, we now sum over $m \in M_u$. The USER condition tests the impact of controlling for variations across users that potentially introduce noise and reduce significance of the ATE.

Table 6.1 shows that compared to the previous condition FILTERED, USER increases ATE to 0.49 and makes it more significant with a p-value of $2 \times 10^{-}12$. The number of matched entry pairs used in the USER condition (614) remains comparable with the number of matched entries used to compute significance in the FILTERED case (632). This validates that a decreased power of the significance test when estimating ATE does not account for the increased significance achieved by the USER condition. This finding substantiates approaches that estimate causal effects at the user-level, aggregating over tweets or entries [108, 35]. The goal of mood logging platforms is to personalize feedback or recommendations for each user, and the USER condition better captures this end goal.

6.4.4 Q3: Incorporating Text Data

For our final question about the additional benefits of incorporating textual data, we apply the USER condition and extend it with TEXT, which upgrades the matching strategy to support variables from text. In TEXT, we replace the distance metric $d(x_i, x_j)$ used in matching with the propensity score which includes text variables. Formally, given treatment assignments T_i and user entries x_i, the text-based propensity score is:

$$P(T_i | x_i, C_i)$$

where $C_i = \{c_1, \ldots, c_{|V|}\}$ is the set of variables where c_j counts the appearances of the j-th unigram from a vocabulary V in the text entry t_i corresponding to x_i. The vocabulary V consists of all unigrams in the text entries t of user U_i. Intuitively, the propensity score models the conditional probability of treatment assignments given attributes of units of interest. We model the propensity score with logistic regression. To reduce overfitting and select a sparser model, we include a L_1 penalty term with a cost parameter C that controls the degree of sparsity. Small values of C induce sparser models and might exclude the covariates that represent the other measured activities such as quality of sleep or food among others.

To evaluate this trade-off between sparsity and a propensity score that encourages better balance across measured covariates, we consider two variants of TEXT, with a low value of $C = 0.01$ that induces greater sparsity and a high value of $C = 0.9$. We evaluate the balance across treated and control groups for each of the other measured activities by performing t-tests to assess the difference in means. Table 6.2 shows the p-values from these t-tests; larger values indicate better balancing of covariates between control and treated groups, which is an important criteria for causal inference.

Table 6.1 shows that condition TEXT, $C = 0.01$ gives the highest ATE of 0.61 and greatest significance, as the p-value is effectively 0.0. However, the p-values for TEXT, $C = 0.01$ in Table 6.2 suggests that the stricter L_1 penalty removes several other measured activity variables from the model, resulting in imbalance for these covariates. In contrast, by setting $C = 0.9$, the balance across these covariates remains comparable with that of the USER matching. Interestingly, for covariates sleep, work and social activity, the balance even improves when we include text-based attributes in the propensity score model. Additionally, the trade-off against estimating ATE remains desirable as TEXT, $C = 0.9$ still gains

over USER in both significance of the ATE and its value.

The results from both estimation and balance suggest that users' language might encode other factors and variables that the study could not measure. However, the balance analysis points to the importance of carefully evaluating modeling choices and parameter settings to not violate key causal assumptions. To further understand the text-based approach, an in-depth exploration of the learned propensity score model is critical to indicate which signals were useful. Below, we follow this analysis up with several qualitative results that shed light on the usefulness of text in causal effect estimation.

6.4.5 Qualitative Results

We further study the TEXT (C=0.9 is used for the remainder of the analysis) propensity score model by examining its features and outputs. We first find the unigrams used in TEXT with the overall highest coefficients in logistic regression and identify two trends. First, we find highly weighted unigrams such as "drained" and "tired" which may serve as proxies for unmeasured confounders that affect both whether a person exercises and their mood. Since EmotiCal users do not record metrics of their health, these adjectives may provide text signals about their physical well-being. Second, we see several positive valence unigram features such as: energized, destresses, enjoyed, great, helped, productivity. Since the users were asked to describe the factors which affected their mood, this finding is expected, but captures reasons for why exercise impacted users' mood positively. These reasons can be interpreted as intermediate variables between exercise and mood, indicating that exercising affects stress levels, for example, which then affects mood.

Next, we provide illustrative examples of treated entries that are matched with

Table 6.3: Examples of matched treatment and control pairs that highlight differences between conditions TEXT and USER. TEXT results in more contextually similar pairs.

Treated Entry	**Text-Matched Control**	**User-Matched Control**
"Really enjoyed a bike ride today to Pioneer park, an old timey park area. It was a fun new experience to explore it, it reminded me a bit of main street in Disney world. Before the bike ride I wrote in my diary too, nice. I feel good, but noah seems more distant today so my mood is more subdued and reflective."	"Visited the visitors center in Fairbanks for a few hours which I had never been to before. I chatted with some nice folks and it was fun. Also had dinner with a friend's family which I enjoyed."	"Slept well. Feeling relaxed."
"Got up early to do yoga class outside in morning ramped up my mood and energy"	"Content that I'm learning different things in excel but today's class required lots of focus"	"Going for walk to harvest garden event and shopping for gardening supplies impacted mood positively although energy could be higher but not due to oversleeping and running late and unhealthy breakfast"
"Went on a run which was good stress relief. Spending the day outside and getting sun also upped my mood. Ate something and got sick from it."	"Got to learn something new at work today which made me happy, but then the new tech that replaced me came in and I started feeling jealous/sad that I never had the interaction she has with my old boss. "	"Drinking makes me happy"

different control entries by TEXT and USER. Table 6.3 contrasts the differences in control entries chosen by each matching strategy. The examples suggest that condition TEXT, which models a text-based propensity score, yields matched pairs that are more lexically similar than those produced by the distance matching in User.

In the first treated entry, the user describes exploring a new area and interestingly, TEXT produces a matching control entry that also discusses travel and exploration. On the other hand, the control entry matched using USER is brief and less related, only discussing sleep. The second treated entry example conveys a relaxed, positive tone which is mirrored in the matched control entry chosen by TEXT. In contrast, USER produces a match that initially exudes a negative tone. The final example suggests a common tone of positivity combined with annoyance in the matched pairs by TEXT while the USER-selected control entry is semantically unrelated.

6.5 Causal Effects of Online Debate Styles

Debate and dialogue on social media sites provide rich observational data for both socio-political and linguistic analysis. Online debate forums are already well-studied for collectively inferring users' stances [69, 40, 39, 157, 56, 91], identifying the polarity of interactions between users [1, 99, 149] and even reasons or moral arguments for their chosen stances [57, 100]. In the context of online debates, our work strives to further facilitate understanding of argumentation styles and their effects. In contrast to the previous section on estimating the effects of exercise, when studying online debates, we only observe text and interactions between users. In this dialogue context, we ask and answer the important question of how various styles of reply impact users' subsequent sentiment and wording choices. Our analysis of dialogue patterns can help suggest effective communication strategies and support downstream interventions to mitigate online harassment.

To estimate the effect that a reply has on subsequent dialogue, we need to ask how else a user may have responded had the reply tone been different. This what-if question requires reasoning about a counterfactual dialogue which remains

unobserved. However, several well-studied methods for estimating counterfactual statements have been proposed in the long-standing research area of causal inference [113, 111, 128]. One such technique is matching samples in training data with their closest counterfactual neighbor based on a propensity score which can be effectively modeled from observations [129]. In the previous section, we use propensity score matching to match users' entries based on both measurements and text. In our debate setting, the propensity score should model the similarity between two dialogues in terms of content to identify the closest match where the reply tone is different but the arguments made are similar.

Recently, approaches to estimating outcomes such as the effects of exercise on mood from social media sites such as Twitter have adapted propensity score matching for text data [108, 35]. In this text-based observational setting, these methods must first identify and extract possible outcomes and influencing variables from text before applying matching techniques. However, counterfactual reasoning with propensity scores based on text from online debates remains largely an open problem. Estimating the effects of dialogue styles requires controlling for several latent confounding variables such as facets of arguments and ideological values espoused by users.

In this work, to estimate the effects of reply styles on subsequent dialogue, we extend propensity score matching for threads of discussion on online debate sites. We evaluate several modeling choices to represent dialogue outcomes and control for latent content confounding. We propose a propensity score that uses a distributional representation of dialogue turns based on latent Dirichlet allocation (LDA) and an interpretable outcome representation that captures changes in wording, sentiment, perception and noun usage. Our technical contributions include:

- Formulating the problem of estimating the effects of reply styles on subsequent dialogue within the framework of causal inference and counterfactual reasoning.
- Extending propensity score matching with latent representations of dialogue content.
- Highlighting interpretable findings on how reply styles affect sentiment and wording choices from a comprehensive analysis across several topics from a real-world debate forums site.

We perform extensive counterfactual analysis on debates from 4Forums.com, a forum corpus that includes annotations for multiple styles of replies. Our findings on the effects of reply styles substantiate long-held domain knowledge that replies can significantly change wording patterns and agreeable replies increase positive sentiment.

6.6 Background and Related Work

Prior work on online debate forums primarily focus on using the textual content and interaction context to predict stance, sentiment or reply polarity [141, 99, 1, 157, 56, 91, 134]. Our work instead focuses on the linguistic analysis of debate dialogue. Our hypotheses on the effects of various reply styles are guided by the well-established theory of linguistic accommodation [48, 52] that dialogue participants adopt one another's wording styles. Recently, approaches have proposed to quantify linguistic accommodation on both Twitter and other dialogue sources such as arguments in front of the U.S. Supreme Court [28, 29]. These methods develop probabilistic models and metrics that capture linguistic accommodation, and evaluate the fit on observed data. Here, we instead formulate

an approach based on counterfactual reasoning and estimate multiple linguistic effects from debate forum data.

In the online debate setting, one line of existing work studies the Change My View forum on Reddit.com to find correlations between argumentation styles and their persuasiveness [149, 164]. These approaches focus on a supervised task of identifying correlations between linguistic patterns and their effectiveness in persuading users to reconsider their views. A similar line of work on the effects of wording and stylistic choice on post likability draws on methods from causal inference [148, 67]. While Jaech et al. [67] control for topic and timing of posts, they focus on developing a classifier. In contrast, Tan et al. [148] propose a matching-based approach to control for the inherent popularity of the user and topic Our work further extends matching to control for latent aspects of argumentation and content in dialogue.

6.7 Dataset

For our estimation of the effects of replies on dialogue, we use the 4Forums.com corpus collected and annotated as part of the Internet Argument Corpus [158]. 4Forums.com has been well-studied for predicting users' stances on a variety of topics, disagreements between users, sarcasm use, and summarizing arguments made by users [92, 100, 157, 159, 141].

4Forums.com is a collection of debate discussions where each discussion belongs to a topic such as "evolution" or "climate change." 4Forums.com includes quote-response pair annotations from Amazon Mechanical Turker workers on a subset of these discussions. A quote-response pairs is an interaction between a user and a replying user where the replier quotes a portion of the original user's post and then directly responds to it. The response is annotated by multiple

Topic	N/N	A/D	R/A	F/F
Abortion	317	378	303	285
Evolution	349	410	331	334
Gay Marriage	158	211	137	127
Gun Control	296	316	279	289

Table 6.4: Numbers of annotated quote-response pairs of posts in the four most annotated debate forum topics. N/N: nice/nasty; A/D: agreement/disagreement; R/A: reason/attack; F/F: fact/feeling

annotators along four dimensions which we refer to as **reply types**: nice/nasty, agree/disagree, fact/feeling, reason/attack. The annotation score for each type ranges from -5 to 5, where negative values correspond to the antagonistic polarity such as nasty or disagree and positive values conversely map to agreement or niceness.

We select the four debate topics with the most quote-response annotations. Table 6.4 shows the number of quote-response pairs annotated in each of these topics across the four reply types. We see that aggregating over topics yields thousands of annotations per reply type. We follow prior work and consider the mean score across annotators for each quote-response pair. Additionally, for users that participate in these quote-response annotated pairs, we have labels for their PRO or ANTI stance toward the topics they debate. In the next section, we formalize the use of these reply type annotations to perform counterfactual reasoning and estimate the effect of each reply type on subsequent dialogue.

6.8 Problem Statement

To study effects on online debate dialogue, we first introduce **post triples**. A **post triple** $t_i = \{p_i^1, p_i^2, p_i^3\}$ is an ordered sequence of three posts where each post p_i^j belongs to the i-th triple and appears j-th in the sequence. Based on

the discussion in which the triple appears, the triple t_i has a debate topic $\tau(t_i)$. Henceforth, we commonly refer to p_i^1 as the original post and to p_i^2 as the reply post. The author of post p_i^j is denoted by $a(p_i^j)$. Each author $a(\cdot)$ has a stance $\sigma(a(\cdot))$ towards the topic $\tau(t_i)$ that belongs to {PRO,ANTI}. In this work, we only consider triples where the original and final post have the same author, i.e. $a(p_i^1) = a(p_i^3)$. This constraint allows us to characterize the change in a particular user's wording and sentiment patterns before and after an dialogue exchange with the replying user.

Section 6.7 introduced annotations for reply posts toward their parent post. The triples we consider belong to the set of quote-response post pairs that are annotated. In causal inference terminology, these annotations constitute the treatment in our analysis whose effects on subsequent dialogue we wish to estimate. We build on notation from the potential outcomes framework [128] which considers binary-valued treatment variables. Given a reply type α and a real-valued annotation score (already averaged across annotators) between -5 and 5, we first binarize the values by considering those ≤ -1 as 0 and ≥ 1 as 1. We follow prior work in not considering annotations with a mean score between -1 and 1 [141]. As an example, if we consider the nice/nasty annotation type, then we treat nasty responses as having a value of 0 and nice ones as value 1. With this binarizing strategy, for each triple t_i and reply type α, the annotation of reply post p_i^2 toward p_i^1 gives the treatment assignment $R_i^\alpha \in \{0, 1\}$ for the triple. In nice/nasty example, we say that a triple t_i where the reply p_i^2 is nice towards p_i^1 is treated, i.e. $R_i^\alpha = 1$ and is a control triple otherwise.

The next problem is to quantify the linguistic changes between p_i^3 and p_i^1 after receiving reply p_i^2. We consider a set of K functions $S = \{S_1(p_i^j), \ldots, S_K(p_i^j)\}$ that map post p_i^j to a vector. One possible vector-valued function could return

a n-vector for a vocabulary of n positive sentiment words where the i-th entry indicates whether word w_i appeared in p_i^j or not. In the next section, we make precise several functions that capture sentiment, wording, perception, and other attributes of posts p_i^j.

Finally, we define the estimation of reply type effects on dialogue for a given reply type α. The units of study are the set of all post triples t. The treatment assignment R_α^i partitions the units into treated and control groups based on the reply post polarity towards the original post. Given a function $S_k(\cdot)$ that measures attributes of posts, the potential outcome of a triple t_i given its treatment assignment is $Y_{R_i^\alpha}(t_i) = \Delta_{S_k(p_i^1),S_k(p_i^3)}$. This class of outcomes capture changes between the original post of user $a(p_i^1)$ and the final post p_i^3 which responds to the reply post p_i^2. The average treatment effect (ATE) on the outcome is given by:

$$ATE = \mathbb{E}_{p(t)}\Big[\mathbb{E}_{p(Y_1|t)}[Y_1(t_i)] - \mathbb{E}_{p(Y_0|t)}[Y_0(t_i)]\Big] \tag{6.3}$$

This quantity estimates the mean difference between the outcome if a triple t_i receives the treatment (a positive polarity reply) and the outcome when t_i receives no treatment (a negative polarity reply). In the observed data, the treatments assignments have already been made, and for each triple, we only observed a single outcome. The missing, unobserved outcome is called the counterfactual and must be imputed somehow from our observed data.

A common approximation of the ATE is to match each triple t_i with another triple t_j such that it has the opposite treatment assignment $R_j^\alpha = 1 - R_i^\alpha$ and $t_j = \arg\min_{t_k \in t} d(t_i, t_k)$ is the nearest neighbor of t_i according to $d(\cdot)$, a measure of distance between triples. The matching produces a set of pairs $M =$

$\{(t_i, t_j)|t_i, t_j \in t, T_i \neq T_j\}$ to estimate ATE as:

$$\hat{ATE} = \frac{1}{|M|} \sum_{m \in M} (2R_i^\alpha - 1)[Y_{R_i^\alpha}(t_i) - Y_{R_j^\alpha}(t_j)] \tag{6.4}$$

Intuitively, we replace the counterfactual outcome with the observed outcome from a highly similar other triple to approximate the treatment effect. The important open problems addressed in this work are designing appropriate distance metrics between post triples that take into account the content in the dialogue, and introducing methods for capturing linguistic changes.

6.9 Text-based Propensity Score

In this section, we introduce various propensity scores of triples that provide an appropriate similarity measure when perform nearest-neighbor matching. Propensity score matching is a well-used technique for estimating the ATE especially from text observations [129, 108, 35]. In our dialogue context, given reply type α, the propensity score is defined as:

$$PS = P(R_i^\alpha = 1|f(t_i))$$

the conditional probability of R_i^α, that a triple receives the treatment of a positive polarity reply, given attributes of triple $f(t_i)$. The attributes, or features, should include those that potentially confound the effect of a reply on linguistic changes in the response post. Similar approaches to studying social media effects already recognize that need to control for the topic of a post [148]. In the dialogue setting, we include the thread topic as a potential confounder, but extend the propensity score model features to capture finer-grained reply post content

including latent representations of the text. We propose multiple choices for modeling these confounders from text, ranging from high-dimensional word counts to low-dimensional distributional representations of posts and authors.

6.9.1 Modeling Dialogue Content

As motivated above, a critical confounder in analyzing the effects of a reply is the topic and content of the reply post itself. The debate topic and finer points of arguments being made by the replying user affect both the polarity of the reply and the subsequent changes in wording and sentiment by the original post's user. Also, the propensity score should be close for triples with contextually similar reply posts so that the selected match will have an opposite polarity reply, but similar arguments. To model this content from triples of dialogue, we introduce coarse- to fine-grained strategies that use text, author stances and debate topics.

Topic-only As in prior work [148], restricting the debate topic of a triple $\tau(t_i)$ and its match $\tau(t_j)$ to be the same is a straightforward but coarse-grained way to capture the general content of a dialogue. This approach is equivalent to setting the propensity score of all triples t_i with the same topic to be 1.0. The matching then randomly breaks ties when selecting the nearest other triple. In our evaluation, we compare this simple matching procedure with the more sophisticated text-based approaches proposed below.

Bag-of-words (BOW) In our first text-based approach, we consider a bag-of-words representation of the reply post p_i^2. Given vocabulary of words V, a post p_i^j is represented by vector $c =< c_1, \ldots, c_|V| >$ of length $|V|$ where c_i counts the appearance of the i-th word in V. We compute these vectors for each reply post p_i^2 to use in the features $f(t_i)$ for each triple t_i. We remove standard English stop

words to obtain the vocabulary over all posts. This bag-of-unigrams representation is high-dimensional but can be computed efficiently.

LDA-based In contrast, prior work has shown that in political debates, the main points made by users lie in a lower dimensional space which correspond to argument facets, or frames, which capture broader moral, economic, or religious principles that guide particular ideological viewpoints [70, 100, 65, 17]. Unsupervised approaches have been used to discover word-clusters that correspond to these frames directly from text [65]. As an alternative to the high-dimensional unigram representation above, we use latent Dirichlet allocation (LDA) [16] to discover k topics of words. LDA infers each post's distribution over these k topics and we use this low-dimensional vector representation of posts as features in $f(t_i)$. The choice of k is discussed in our analysis. We combine two granularities of LDA features: latent representations of each post and those of each author obtained by concatenating the posts of authors to train LDA.

LIWC features For the BOW and LDA-based approaches, we follow much prior work in computational linguistics [157] and include an additional vector representation of post p_i^2 in our feature set $f(t_i)$ based on the Linguistic Inquiry and Word Count (LIWC) tool [114]. LIWC is a dictionary which maps an extensive set of English words to categories that capture both lexical and semantic choices. LIWC has been successfully used in several text classification tasks for a more sophisticated, though shallow, representation of text [157, 1, 99]. For each p_i^2 and LIWC categories L, we compute an $|L|$-vector where each entry i captures the normalized counts of the i-th category in L. We include this vector in $f(t_i)$ to further model the content of reply posts.

Author stance constraints In all settings, to further account for potential variances in the linguistic change outcome from content confounding, we include a user-stance constraint. For t_i and its match t_j, we restrict the stance $\sigma(a(p_i^1))$ and $\sigma(a(p_j^1))$ towards topic $\tau(t_i)$ to be the same. Users' stance offers another coarse-grained proxy that captures their ideological views and bases of their arguments. This stance restriction still yields several hundred matched triples for each reply type.

Wording Changes	Positive Sentiment	Negative Sentiment	Nouns	Perception
Total Function Words	Positive Emotion	Swear Words	Family	Tentative
Total Pronouns	Achievement	Negative Emotion	Friends	Inhibition
Personal Pronouns	Assent	Anxiety	Humans	Certainty
First Person Singular	Insight	Anger	Biological Processes	Affective Processes
First Person Plural		Sadness	Body	Cognitive Processes
Second Person			Health	Perceptual Processes
Third Person Singular			Sexual	See
Third Person Plural			Ingestion	Hear
Impersonal Pronouns			Time	Feel
Articles			Work	
Common Verbs			Leisure	
Auxiliary Verbs			Home	
Past Tense			Money	
Present Tense			Religion	
Future Tense			Death	
Adverbs			Motion	
Prepositions			Space	
Conjunctions				
Negations				
Quantifiers				
Number				
Causation				
Discrepancy				
Relativity				
Nonfluencies				
Fillers				
Social Processes				

Figure 6.2: LIWC categories that belong to each vector that captures representations of posts and enable measuring change in wording choices and sentiment.

6.10 Measuring Linguistic Outcomes

To obtain the set of functions S which allow us to measure the change across p_i^1 and p_i^3, we follow other linguistic analyses [28] and again use LIWC. We first combine LIWC categories into groups that measure wording, positive sentiment, negative sentiment, perception and common nouns. Fig. 6.2 lists the LIWC categories in each of these groupings. These groupings allow us to construct vector representations of the original post p_i^1 and final post p_i^3. We then use cosine similarity to compute a difference between posts. For each of these five **category types**, k, consisting of l LIWC categories, we obtain a corresponding $S_k(\cdot)$ that returns a l-vector for a post p_i^j where each entry i captures the counts of the i-th category in category type k. To compute $\Delta_{S_k(p_i^1),S_k(p_i^3)}$, we measure the cosine similarity of the two vectors. This strategy suggests a rich set of possible vector representations of posts including document embeddings [98]. However, in this work, our choice of LIWC vector representation allows us to maintain interpretability of results.

6.11 Empirical Analysis

The goals of our empirical analysis are twofold: 1) comparing the BOW feature representation against the more sophisticated LDA-based distributional features in simply predicting observed reply types; 2) estimating dialogue effects from reply types for each of our matching strategies to contrast these findings against known socio-linguistic theories. We consider three dialogue-based propensity score matching approaches: topic-only, BOW, and LDA-based. We apply these strategies to estimate effects on debates from 4Forums.com and highlight significant dialogue changes from various reply tones. Since gold-standard true linguistic

effects are not available, we contrast our findings with well-established domain knowledge on linguistic accommodation and argumentation behaviors.

6.11.1 Experimental Setup

All matching variants are implemented in `Python`, with BOW and LDA-based propensity scores trained using logistic regression in the `sci-kit learn` library. LDA is also trained using the implementation in `sci-kit learn`. To select the best value for the number of topics k for each debate topic, we hold out a development set when training LDA and choose k that maximizes marginal likelihood of the observed text. We consider $k = \{2, 3, 4, 5, 6\}$. To compute the post representations for both LDA and BOW, we exclude unigrams occurring in more than 60% of the posts to eliminate generic words. For BOW, this document frequency threshold reduces the dimensionality of the representation. We also report the cross-validation F1 score of trained propensity score models for predicting each binarized reply type annotation. We perform five-fold cross-validation, training the propensity score model on four folds of the available reply type annotations, and validating on the remaining fold.

Reply type	**BOW**	**LDA-based**
Nice/Nasty	0.57 ± 0.05	$\mathbf{0.83 \pm 0.03}$
Agree/Disagreement	0.17 ± 0.02	0.17 ± 0.03
Reason/Attack	0.55 ± 0.06	$\mathbf{0.76 \pm 0.04}$
Fact/Feeling	0.45 ± 0.11	$\mathbf{0.70 \pm 0.09}$

Table 6.5: Mean F1 scores from cross-validation averaged also across topics. We compare BOW and LDA-based propensity score models in predicting binary observed reply types (i.e. treatment assignment). We see that the latent LDA representations used as features are significantly more predictive in three out of four reply type settings.

6.11.2 Results and Findings

The first part of our evaluation focuses on predictive performance to validate the use of either dialogue-based propensity score model. The second analysis compares the fine-grained content-based approaches to the topic-only matching strategy to contrast the differences in findings. The results motivate an in-depth look into how sentiment changes differ between treated and control triples across each content-based approach.

Cross-validation performance We first validate BOW and LDA-based propensity score models against predicting the binary reply polarity (i.e. the treatment assignment for each triple) for each reply type by performing cross-validation. Since we train propensity score models per debate topic, we compute an average across topics over mean cross-validation F1 performance. Table 6.5 shows the mean F1 scores for each method, separated by reply type. The results suggest that the distributional representation of posts based on latent groupings of words from LDA are more powerful in predicting the polarity of observed replies. The LDA-based features for propensity score models outperform the BOW features in three out of four reply type settings. We see that agreement/disagreement reply type yields the lowest predictive performance and a closer look shows that replies are highly skewed towards disagreement than agreement. This imbalance in agreement/disagreement replies matches the intuition that debates inherently provoke more disagreement than agreement. However, the skew necessitates future work in developing more sophisticated linguistic features which capture deeper text semantics to overcome the imbalance.

Estimation of treatment effects We apply each of the three matching strategies proposed in this work to our debate triples dataset: 1) topic-only, 2) BOW

Approch	Reply Type	Wording	Common Nouns	Perception	Pos. Sentiment	Neg. Sentiment
Topic-only	N/N	✓				
	A/D	✓	✓	✓		
	R/A	✓				
	F/F		✓			
BOW	N/N	✓	✓	✓	✓	✓
	A/D	✓	✓			✓
	R/A	✓	✓			✓
	F/F	✓			✓	✓
LDA-based	N/N					✓
	A/D	✓				
	R/A	✓		✓	✓	✓
	F/F			✓		✓

Table 6.6: Checkmarks indicate a significant difference (at level $\alpha = 0.1$) in the particular LIWC-vector outcome between treated and control groups for a given reply type. The large number of significant changes in wording found by all matching strategies supports the intuition that the tone of a reply provokes different word usage. However, the topic-based approach finds no changes sentiment while the text-based matching approaches do.

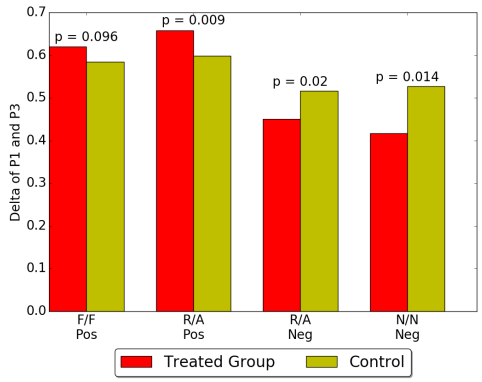
and 3) LDA-based. After triples differing in treatment assignment are matched, we compute the ATE using Equation 6.4, which calculates the mean difference in outcome across matched pairs. We measure the ATE in turn using the $\Delta_{S_k(p_i^1),S_k(p_i^3)}$ representation based on each of the five LIWC category groupings shown in Fig. 6.2. Table 6.6 shows the $\Delta_{S_k(\cdot),S_k(\cdot)}$ change outcomes which vary significantly (at significance level $\alpha = 0.1$) between treated and control groups for each possible LIWC vector representation and each reply type which constitutes a treatment. The checkmarks indicate that a significant average treatment effect occurs.

The first notable insight is that all three matching methods find multiple significant wording changes. BOW estimates that wording significantly changes for all reply types, while LDA-based uncovers that agreement/disagreement and rea-

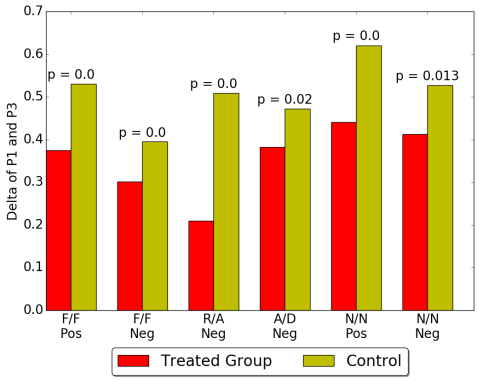
son/attack provoke significant changes in wording. These results support prior work which suggests that dialogue dynamics do affect users' subsequent responses [149, 29]. While our formulation differs from measuring linguistic accommodation, our framework can be naturally extended to analyze such effects. The second notable result is that the topic-only matching strategy does not find any significant changes in sentiment, while the LDA-based and BOW approaches estimate significant negative and positive sentiment changes as the polarity, or tone, of the reply varies, especially for nice/nasty, reason/attack and fact/feeling reply types. The changes in sentiment fit with the intuition that negative replies breed further negativity in a dialogue, and vice versa. The final finding is that overall, BOW estimates the most number of significant treatment effects across all reply types but its poorer performance compared to LDA-based in predicting held-out reply polarity suggests that confounding may still exist in its estimates of effects. Often, better adjustment for confounders reduces the effect of the treatment alone on the outcome, which may explain the reduced number of effects found by the LDA-based approach.

Examination of effects on sentiment The effects of replies on change in sentiment highlighted by Table 6.6 prompt a closer look into the differences between treated and control triples across all reply types. Fig. 6.3 plots the mean $\Delta_{S_k(p_i^1),S_k(p_i^3)}$ values when considering positive and negative sentiment LIWC vectors for control and treated groups. We consider the differences for each reply type and across both LDA-based and BOW strategies. The p-value of these differences is shown, and we abbreviate the reply types as detailed in the figure and indicate the positive or negative LIWC sentiment vector considered below the reply type.

Fig. 6.3a shows that for LDA-based propensity score matching, factual and non-attacking language as captured by the treated groups increases positive sen-



(a) LDA-based propensity score model. N/N: nice/nasty, A/D: agree/disagree, F/F: fact/feeling, R/A: reason/attack.



(b) BOW propensity score model.

Figure 6.3: A closer look at significant positive and negative sentiment changes between treated and control groups across reply types when using each type of propensity score model.

timent. In a similar vein, attacking and nasty replies which correspond to control groups, show significant increases in negative sentiment. Both of these trends support domain knowledge on argumentation, that antagonistic replies can provoke defensive behavior. Fig. 6.3b shows sentiment changes found by the BOW approach. On one hand, the negative sentiment increase found when replies are nasty, attacking, disagreeing or feeling seem to corroborate the LDA-based findings. However, BOW finds that positive sentiment decreases when replies are nice or factual, which contradicts our intuition. This finding further indicates the importance of investigating modeling assumptions since the BOW approach may simply be estimating spurious effects.

6.12 Discussion

This chapter extends causal inference methods to combine text data, ranging from fusing text with traditional measurements to using only observed textual

sources. In Section 6.1, we introduce a propensity score matching method that integrates both measurements and text to estimate the average treatment effect between exercise and mood. We use data from the recently developed EmotiCal mood-logging application. We develop our approach TEXT which incorporates text variables by carefully examining several modeling choices. Our findings highlight the importance of user-specific, stratified analysis when modeling user-behavior domains. Our preliminary results suggest future work in modeling dependencies between multiple causal factors and finding latent representations of confounders from text.

In Section 6.5, we estimate the effect of various reply styles on the change in users' linguistic choices and sentiment. We formulate this problem within the framework of causal inference, using only dialogue threads from online debates. We propose multiple propensity scores that use coarse- to fine-grained representations of dialogue content, ranging from simple debate topic to latent representations of posts and authors. Our extensive analysis of four reply types and multiple outcome representations for measuring the change in users' posts validates domain knowledge and intuition around argumentation and debates. We find that wording changes often differ significantly depending on whether users' receive positive or negative polarity replies, and positive polarity replies typically increase positive sentiments. Our contribution points to a promising avenue of argumentation research, including future work in using this framework to validate socio-linguistic theories around linguistic accommodation, power dynamics, and persuasion.

This chapter concludes the exploration of causality in computational science and contrasts against the causal graph discovery problem of the previous chapter. We thus address the third requirement for extending PSL to meet computational science desiderata by inferring causal knowledge. In the next chapter, we turn to

the fourth and final task of discovering PSL model structure from data.

Chapter 7

Learning Structured Models

Chapter 5 motivates the need for model structure discovery algorithms when domain knowledge to specify a probabilistic model is limited. Chapter 5 introduces the CausPSL modeling framework that infers causal graphs by fusing constraints, statistical tests, and side information. The inferred causal graphs are DAGs that define BN distributions, and as described in Section 2.1, the d-separation criteria for DAGs induces a correspondence between observed independences and graph structure, providing useful structural constraints. Moreover, logical characterizations of these constraints supported the specification of CausPSL. In contrast, this chapter focuses on discovering the model structure for PSL and HL-MRFs, an undirected and logic-templated graphical model. In contrast to directed models, SRL methods such as PSL present new challenges for model structure discovery algorithms.

As discussed in Section 2.2, SRL frameworks such as MLNs and PSL encode model structure with weighted first-order logical clauses. The model discovery problem of learning these clauses from data is referred to as structure learning. As in Chapter 5, structure learning is a critical step in computational science tasks to discover new knowledge. Additionally, structure learning alleviates the

manual cost of specifying models and evaluating modeling decisions, as we propose in Chapter 3. However, these benefit comes with high computational costs; structure learning typically requires an expensive search over the space of clauses which involves repeated optimization of clause weights. This chapter proposes the first two approaches to structure learning for PSL. We introduce a greedy search-based algorithm and a novel optimization method that trade-off scalability and approximations to the structure learning problem in varying ways. The highly scalable optimization method combines data-driven generation of clauses with a piecewise pseudolikelihood (PPLL) objective that learns model structure by optimizing clause weights only once. We compare both methods across several tasks including the familiar drug-drug interaction prediction setting. However, we revisit the task with a complex relational database that describes relationships between drugs, enzymes, transporters and other biological entities. We demonstrate that structure learning discovers complex modeling patterns for this domain that outperform the previously proposed similarity-based method. Finally, we show that PPLL achieves an order of magnitude runtime speedup and AUC gains up to 15% over greedy search.

This chapter thus addresses the final task of extending PSL for computational science domains: discovering PSL model structure from data. The subsequent final chapter provides concluding remarks and highlights the key interplay between these chapters in learning structured and causal probabilistic models for computational science.

7.1 Structure Learning for PSL

Statistical relational learning (SRL) methods combine probabilistic reasoning with knowledge representations that capture the structure in problem domains.

Markov logic networks (MLN) [126] and probabilistic soft logic (PSL) [7] are notable SRL frameworks that define model structure with weighted first-order logic. However, specifying logical clauses for each problem is laborious and requires domain knowledge. The task of discovering these weighted clauses from data is referred to as *structure learning*, and has been well-studied for MLNs [79, 81, 82, 97, 14, 60, 76, 78]. The extensive related work for MLNs underscores the importance of structure learning for SRL.

Structure learning approaches alleviate the cost of model discovery. However, they face several critical computational challenges. First, even when the model space is restricted to be finite, it results in a combinatorial search. Second, heuristic approaches that iteratively refine and grow a set of rules require interleaving of several costly rounds of parameter estimation and scoring. Finally, scoring the model often involves computing the model likelihood which is typically intractable to evaluate exactly.

Structure learning approaches for MLNs vary in the degree to which they address these scalability challenges. An efficient and extensible class of MLN structure learning algorithms adopt a *bottom-up* strategy, mining patterns and motifs from training data to generate informative clauses [97, 81, 82]. The data-driven heuristics reduce the search space to useful clauses but still interleave rounds of parameter estimation and scoring, which is expensive for SRL methods.

Motivated by the success of structure learning for MLNs, in this paper, we formalize the structure learning problem for PSL. We extend the data-driven approach to generating clauses and propose two contrasting PSL structure learning methods that differ in scalability and choice of approximations. We build on *path-constrained* relational random walk methods [85, 49] to generate clauses that capture patterns in the data. To find the best set of clauses, we introduce a

greedy search-based algorithm and an optimization method that uses a piecewise pseudolikelihood (PPLL) objective function. PPLL decomposes the search over clauses into a single optimization over clause weights that is solved with an efficient parallel algorithm. Our proposed PPLL approach addresses the scalability challenges of structure learning and its formulation can be easily extended to other SRL techniques, including MLNs. In this paper, our key technical contributions are to:

- formulate path-constrained clause generation that efficiently finds relational patterns in the data.
- propose greedy search and PPLL methods that select the best path-constrained clauses by trading off scalability and approximations for structure learning.
- validate the predictive performance and runtimes of both methods with real-world tasks in biological paper recommendation, drug interaction prediction and knowledge base completion.

We compare both proposed PSL structure learning methods and show that our novel PPLL method achieves an order of magnitude runtime speedup and AUC improvements of up to 15% over the greedy search method.

7.2 Background

We briefly review of structure learning for statistical relational learning (SRL) before formalizing structure learning in the context of PSL and HL-MRFs.

7.2.1 Structure Learning for SRL

Our work focuses on SRL methods such as MLNs and PSL that encode dependencies with first-order logic. Chapter 2 introduces the notation for these SRL methods. Below, we define the structure learning for these methods.

The problem of *structure learning* finds the model $M_{C,\mathbf{w}}$ which best fits a set of observed assignments $\mathbf{X}$, regularized by model complexity. We denote the set of possible clauses as the language $\mathcal{L}$. Although $\mathcal{L}$ can be infinite, it is standard to impose restrictions that make $\mathcal{L}$ finite for structure learning. Furthermore, we make the commonly used closed-world assumption that all predicates are fully observed. Formally, the structure learning problem finds $C \subseteq \mathcal{L}, \mathbf{w} \in \mathbb{R}^N, N = |C|$ that maximize a regularized log likelihood function $l_{ll}(C, \mathbf{w})$ given observed assignments:

$$\begin{aligned} &\arg\max_{\mathbf{w}\in\mathbb{R}^N,\ C\subseteq\mathcal{L}} l_{ll}(C, \mathbf{w}) \\ &= \arg\max_{\mathbf{w}\in\mathbb{R}^N,\ C\subseteq\mathcal{L}} \log P_{C,\mathbf{w}}(\mathbf{X}) - r(C, \mathbf{w}) \end{aligned} \tag{7.1}$$

where $r(C, \mathbf{w})$ represents priors on the weights and structure. Typical choices for r combine a Gaussian prior on weights and an exponential prior on clause length.

The log likelihood requires an exponential sum to compute Z and the optimization combines a combinatorial search over $\mathcal{L}$ with a maximization of continuous weights $\mathbf{w}$ (called weight learning). Consequently, solving structure learning requires further approximations to search and scoring. Approaches to structure learning broadly interleave two key components: clause generation and model evaluation, or scoring. The clause generation phase produces a candidate language $\mathcal{L}$ over which to search. In practice, $\mathcal{L}$ is a subset of all possible clauses, chosen to restrict the search to useful regions of the space. Model evaluation typically iteratively refines the existing model by learning $\mathbf{w}$ and scoring candidate

clauses in $\mathcal{L}$ using approximations to $l_{ll}(C, \mathbf{w})$.

7.3 Problem Statement

Given target predicates $\mathbb{P}_T$, structure learning for PSL finds a model $\tilde{M_{C,\mathbf{w}}}$ to infer $t_i \in \mathbb{P}_T$. The language space $\mathcal{L}_R$ is restricted to clauses of the form $\wedge_i L_i \rightarrow \vee_i T_i$ and we again constrain $\mathcal{L}_R$ to be finite. In Section 2.3, we review the pseudolikelihood approximation to the likelihood score. For structure learning, as in weight learning, it is conventional to use the pseudolikelihood score to optimize over the space of clauses.

Given target predicates $\mathbb{P}_T$, real-valued variable assignments $\mathbf{Y}$ and $\mathbf{X}$ where each Y_i atom consists of $p \in \mathbb{P}_T$, following the objective in Equation 7.1, structure learning for PSL maximizes log pseudolikelihood $l_{pll}(C, \mathbf{w})$:

$$\arg\max_{C \subseteq \mathcal{L}_R, \mathbf{w} \in \mathbb{R}^+} \sum_{Y_i \in Y} -\log(Z_i) - \mathbf{w}^T \Phi_C(\mathbf{X}, \mathbf{Y}) \tag{7.2}$$

where Φ_C denotes all ground rules that can be instantiated from clauses C. In the next section, we propose two approaches to the structure learning problem for HL-MRFs that rely on an efficient clause generation algorithm.

7.4 Approaches

To formulate PSL structure learning algorithms, we introduce approaches for both key method components: clause generation and model evaluation. We outline an efficient algorithm for data-driven clause generation. For model evaluation over these clauses, we first propose a straightforward greedy local search algorithm (GLS). To improve upon the computationally expensive search-based approach,

we introduce a novel optimization approach, piecewise pseudo-likelihood (PPLL). PPLL unifies the efficient clause generation with a surrogate convex objective that can be optimized exactly and in parallel.

7.4.1 Path-Constrained Clause Generation

The clause generation phase of structure learning outputs the language $\mathcal{L}_R$ of first-order logic clauses over which to search. Driven by relational random walk methods used for information retrieval tasks [86, 49], we formulate a special class of *path-constrained clauses* that capture relational patterns in the data. Path-constrained clause generation is also related to the pre-processing steps in bottom-up structure learning methods [97, 81, 82]. Bottom-up methods typically use relational paths as heuristics to cluster predicates into templates and enumerate all clauses that contain predicate literals from the same template. The structure learning algorithm greedily selects from these clauses. Path-constrained clause generation also produces $\mathcal{L}_R$ prior to structure learning. Here, we use a breadth-first traversal algorithm which directly generates informative path-constrained clauses by variablizing relational paths in the data.

The inputs to path-constrained clause generation are the ground atoms of a domain, the set of all predicates $\mathbb{P}$ and target predicate $\mathbb{P}_T$. In this work, we consider predicates with arity of two but our approach will be extended to support predicates with arity three and higher. We begin with a running example that illustrates the definitions below.

Example 1. Consider a ground atom set with CITES(Paper1, Paper2), MENTIONS(Paper2, Gene), MENTIONS(Paper1, Gene) and $\mathbb{P}_T = \{\text{MENTIONS}\}$. In this simple example, all ground atoms have an assignment of 1. In general, real-valued assignments to atoms must be rounded to 0 or 1 during path-constrained

clause generation.

Definition 6. A **target relational path** for $t_i \in \mathbb{P}_T$ denoted $\pi_j^{t_i}$ is defined by an ordered list of ground atoms $[p_1(e_1, e_2), p_2(e_2, e_3) \ldots, p_s(e_s, e_{s+1}), t_i(e_1, e_{s+1})]$ such that each $p_i(e_i, e_{i+1}) = 1$, its last argument e_{i+1} is the first argument of $p_{i+1}(e_{i+1})$, and $t_i(e_1, e_{s+1}) \in \{0, 1\}$ is a target atom.

Definition 7. Given a target relational path $\pi_j^{t_i}$, the corresponding *first-order* **path-constrained clause** $c_{\pi_j}^{t_i}$ has the form $p_1(E_1, E_2) \wedge \ldots \wedge p_s(E_s, E_{s+1}) \rightarrow t_i(E_1, E_{s+1})$ where each E_i is a logical variable and the j-th literal in the clause variablizes the j-th atom in $\pi_j^{t_i}$. The **negation** of $c_{\pi_j}^{t_i}$ is the clause with $\neg t_i(E_1, E_{s+1})$, the target predicate literal negated.

For Example 1, given target relational path [CITES(Paper1, Paper2), MENTIONS(Paper2, Gene), MENTIONS(Paper1, Gene)], we obtain the first-order path-constrained clause:

$$\text{CITES}(E_1, E_2) \wedge \text{MENTIONS}(E_2, E_3) \rightarrow \text{MENTIONS}(E_1, E_3)$$

We generate the set of all possible path-constrained clauses C_Π up to length s, by performing breadth-first search (BFS) of up to depth s from the first argument e_j of each target atom $t_i(e_j, e_k)$.

Definition 8. A **connected BFS search tree** b_{jk}^i for training example $t_i(e_j, e_k)$ is rooted at e_j and one of its leaf nodes *must* be e_k. Every non-leaf constant e_u in b_{jk}^i has child entities e_v connected by ground atoms $p_i(e_u, e_v) = 1$.

For Example 1, the connected BFS search tree of depth 2 for target atom MENTIONS(Paper1, Gene) is:

$$\text{Paper1} \xrightarrow{\text{CITES}} \text{Paper2} \xrightarrow{\text{MENTIONS}} \text{Gene}$$

Given a tree b^i_{jk}, each path from its root e_j to leaf node e_k is a target relational path $\pi^{t_i}_j$. For target predicate t_i, $B^i = \{b_1 \dots b_n\}$ is the set of connected BFS search trees corresponding to all n target atoms. For all $t_i \in \mathbb{P}_T$, we enumerate all such $\pi^{t_i}_i$ from each $b \in B^i$ and obtain the unique set of these paths Π. For each $\pi_i \in \Pi$, we form the corresponding path-constrained clause and its negation to obtain all such clauses C_Π. Moreover, we can further restrict C_Π to those clauses that connect $\geq t$ target atoms, preferring clauses that cover, or explain, at least training t examples. The language defined by C_Π guides the search over models that capture informative relational patterns in the data. Although C_Π produces only Horn clauses and is thus a subset of the language $\mathcal{L}_R$ [**?**], it has been successfully used in several relational learning tasks [85, 49]. While our path-constrained clause generation performs well in the tasks we study, where needed, we will explore more expressive strategies.

7.4.2 Greedy Local Search

Given N path-constrained clauses, exactly maximizing the pseudolikelihood objective given by Equation 2.9 requires evaluating 2^N subsets of clauses, which is already infeasible with only 100 clauses. Instead, we propose an approximate greedy search algorithm that selects locally optimal clauses in each iteration to maximize pseudolikelihood.

Algorithm 1 gives the pseudocode for **greedy local search** (GLS) which approximately maximizes the pseudolikelihood score $l_{pll}(\cdot)$. GLS iteratively picks the $c^* \in C_\Pi$ that maximizes $l_{pll}(\cdot)$ and adds it to the model M until the score has only improved by $\leq \epsilon$ or a maximum number of iterations l has been reached. While GLS is straightforward to implement, it requires $O(Nl)$ rounds of weight learning and evaluating $l_{pll}(\cdot)$ where N denotes the size of C_Π. As N grows, the

Algorithm 1 Greedy Local Search (GLS)

Input: C_Π: path-constrained clauses; ϵ: tolerance; l: max iterations
Output: $C^*, \mathbf{w}$: optimal clauses and weights

```
S ← C_Π
C* ← ∅
current, prev, i ← 0
while current − prev ≥ ε or i ≤ l do
    current ← prev
    for s ∈ S do
        C* ← C* ∪ s
        score ← max_w l_pll(C*, w)
        if score > current then
            current ← score
            c* ← s
        C* ← C* \ s
    C* ← C* ∪ c*
    S ← S \ c*
    i ← i + 1
```

GLS becomes prohibitively expensive unless we sacrifice performance by increasing ϵ or decreasing l. To overcome the scalability pitfalls of GLS and search-based methods at large, we introduce a new structure learning objective that can be optimized efficiently and exactly.

7.4.3 Piecewise Pseudolikelihood

The partition function Z_i in pseudo-likelihood involves an integration that couples all model clauses. Optimizing pseudo-likelihood requires evaluating all subsets of the language $\mathcal{L}_R$, necessitating greedy approximations to the combinatorial problem. To overcome this computational bottleneck, we propose a new, efficient-to-optimize objective function called **piecewise pseudolikelihood** (PPLL). Below, we derive two key results which have significant consequences for scalability of structure learning: 1) with PPLL, structure learning is solved by performing weight learning once; and 2) the factorization used by PPLL admits

an inherently parallelizable gradient-based algorithm for optimization.

PPLL was first proposed for weight learning in conditional random fields (CRF) [147]. For HL-MRFs, PPLL factorizes the joint conditional distribution along both random variables and clauses and is defined as:

$$
\begin{aligned}
&P^*_{\tilde{M_{C,\mathbf{w}}}}(\mathbf{Y}|\mathbf{X}) = \prod_{c \in C} \prod_{Y_i \in \mathbf{Y}} \frac{\exp(-f_i^c(Y_i, \mathbf{Y}, \mathbf{X}))}{Z_i^c(\mathbf{Y}, \mathbf{X})} \\
&\text{where} \\
&Z_i^c(\mathbf{Y}, \mathbf{X}) = \int_{Y_i} \exp(-f_i^c(Y_i, \mathbf{Y}, \mathbf{X})) \\
&f_i^c(Y_i, \mathbf{Y}, \mathbf{X}) = \sum_{j: Y_i \in G_c} w_j \phi_j(Y_i, \mathbf{Y}, \mathbf{X})
\end{aligned}
\tag{7.3}
$$

The key advantage of PPLL over pseudo-likelihood arises from the factorization of Z_i into Z_i^c, which requires only clause c and variable Y_i for its computation.

Following standard convention for structure learning, we optimize the log of PPLL denoted $l_{ppll}(C, \mathbf{w})$. We highlight a connection between PPLL and pseudo-likelihood that is useful in deriving the two key scalability results of PPLL. The product of terms in PPLL corresponding to clause c is the log pseudo-likelihood of the model containing only clause c. We denote this $l_{pll}^c(w_c)$:

$$
l_{pll}^c(w_c) = \sum_{Y_i \in \mathbf{Y}} -\log(Z_i^c(\mathbf{Y}, \mathbf{X})) - f_i^c(Y_i, \mathbf{Y}, \mathbf{X}) \tag{7.4}
$$

We now show that for the log PPLL objective function, performing weight learning on the model containing all clauses in $\mathcal{L}_R$ is equivalent to optimizing the

objective function over the space of all models. Formally:

$$\begin{aligned} &\arg\max_{C \subseteq \mathcal{L}_R, \mathbf{w} \in \mathbb{R}^+} l_{ppll}(C, \mathbf{w}) \\ &\equiv \\ &\arg\max_{\mathbf{w} \in \mathbb{R}^+} l_{ppll}(\mathcal{L}_R, \mathbf{w}) \end{aligned} \tag{7.5}$$

Lemma 1. *Optimizing $l_{ppll}(C, \boldsymbol{w})$ over the set of weights $\boldsymbol{w}$ is equivalent to optimizing over each w_c separately.*

Proof Each $l^c_{pll}(w_c)$ is a function of only w_c. By definition of $l_{ppll}(C, \mathbf{w})$, we have

$$\begin{aligned} \arg\max_{\mathbf{w} \in \mathbb{R}^+} l_{ppll}(C, \mathbf{w}) &= \arg\max_{\mathbf{w} \in \mathbb{R}^+} \sum_{c \in C} l^c_{pll}(w_c) \\ &= \sum_{c \in C} \arg\max_{w_c \in \mathbb{R}^+} l^r_{pll}(w_c) \end{aligned}$$

■

Theorem 1. *For PPLL, maximizing the weights* $\mathbf{w}$ *of the model containing all clauses in $\mathcal{L}_R$ is equivalent to optimizing the structure learning objective.*

Proof

$$\arg\max_{C\subseteq\mathcal{L}_R,\mathbf{w}\in\mathbb{R}^+} l_{ppll}(C,\mathbf{w})$$

$$= \arg\max_{C\subseteq\mathcal{L}_R} \sum_{c\in C} \arg\max_{w_c\in\mathbb{R}^+} l^c_{pll}(w_c) \text{ [Lemma 1]}$$

By setting $w_c = 0$, we get $l^r_{pll}(w_c) = 0$.

Therefore, the maxima must be non-negative, i.e.:

$$\arg\max_{w_c\in\mathbb{R}^+} l^c_{pll}(w_c) \geq 0.$$

This implies that:

$$\arg\max_{C\subseteq\mathcal{L}_R} \sum_{c\in C} \arg\max_{w_c\in\mathbb{R}^+} l^c_{pll}(w_c)$$

$$= \sum_{c\in\mathcal{L}_R} \arg\max_{w_c\in\mathbb{R}^+} l^c_{pll}(w_c)$$

$$= \arg\max_{\mathbf{w}\in\mathbb{R}^+} l_{ppll}(\mathcal{L}_R,\mathbf{w})$$

■

As a result of Theorem 1, instead of combinatorial search, we perform a simpler continuous optimization over weights that can be solved efficiently. Since the objective is convex, and the weights are non-negative, we optimize the above objective using projected gradient descent.

The projected gradient descent algorithm for optimizing the objective function is shown in Algorithm 2. The partial derivative of $l_{ppll}(C,\mathbf{w})$ for a given weight w_c is of the form:

$$\nabla_{w_c} = \Phi_c(Y_i,\mathbf{Y},\mathbf{X}) - \mathbb{E}_{ppll}[\Phi_c(Y_i,\mathbf{Y},\mathbf{X})]$$

where (7.6)

$$\Phi_c(Y_i,\mathbf{Y},\mathbf{X}) = \sum_{Y_i\in\mathbf{Y}} \sum_{j:Y_i\in G_c} \phi_c(Y_i,\mathbf{Y},\mathbf{X})$$

The gradient for any weight w_c is the difference between observed and expected penalties summed over corresponding ground clauses G_c. For both pseudo-likelihood and PPLL, we can compute observed penalties once and cache their values but the repeated expected value computations, even for a one-dimensional integral, remain costly. However, unlike the gradients for pseudo-likelihood, each expectation term in the PPLL gradient considers a single clause. Thus, when evaluating gradients for weight updates in Algorithm 2, we use multi-threading to compute the expectation terms in parallel. The dual advantages of parallelizing and requiring weight learning only once makes PPLL highly scalable. After convergence of the gradient descent procedure, we return the set of clauses with non-zero weights as the final model.

Algorithm 2 Piecewise Pseudolikelihood (PPLL)

Input: C_Π: path-constrained clauses; ϵ: tolerance; l: max iterations; α: step size
Output: $C^*, \mathbf{w}$: optimal clauses and weights

```
for c ∈ C_Π do
    C* ← c
i ← 0
score_prev ← −∞
score_curr ← l_ppll
while score_curr − score_prev > ϵ or i < l do
    i ← i + 1
    for c ∈ C* do
        w_c ← w_c + α∇_{w_c}
        if w_c < 0 then
            w_c = 0
    score_prev ← score_curr
    score_curr ← l_ppll
for c ∈ C* do
    if w_c = 0 then
        C* ← C* \ c
```

Finally, although it is beyond the scope of this work to formalize PPLL for MLNs, our proposed formulation can be extended to MLNs by considering a

discrete variant of pseudolikelihood [126, 78, 79]. The computations required by PPLL for MLNs will involve summing instead of integration, and require counting of satisfied Boolean clauses instead of simply evaluating the value of continuous hinge-loss satisfaction as we do for PSL. Interestingly, since MLNs typically describe joint distributions instead of conditionals for particular target variables as PSL does, we can expect to see further scalability gains from applying fully factorized PPLL to MLNs.

7.5 Experimental Results

The PPLL optimization method uses a fully factorized approximation for scalability while GLS greedily maximizes the less decoupled pseudolikelihood at the expense of speed. We explore the trade-offs made by these two methods by evaluating predictive performance and scalability. We investigate these experimental questions with five prediction tasks and compare PPLL against GLS after generating path-constrained clauses. The evaluation tasks include paper recommendation in biological citation networks, drug interaction prediction and knowledge base completion.

7.5.1 Datasets

For our datasets, we obtain citation networks for biological publications, drug-drug interaction pharmacological networks and knowledge graphs.

Biological Citation Networks Our first dataset consists of biology-related papers and entities such as authors, venues, words, genes, proteins and chemical compounds [86]. The dataset includes relations over these entity types for two domains, "Fly" and "Yeast", resulting in two citation networks. The prediction

target is the GENE relation between genes and papers that mention them. To enforce training only on papers from the past, we partition papers into periods of time, using those from 2006 as observations, training on papers from 2007 and evaluating on papers from 2008. We randomly subsample targets to obtain 1500 train and test links, and generate five such random splits for cross-validation.

Drug-drug interaction The second dataset we use includes chemical interactions between drug pairs, called drug-drug interactions (DDI) across 196 drug compounds obtained from the DrugBank database. This dataset also contains a directed graph of relations from Drugbank between these drugs and gene targets, enzymes, and transporters. Our target for prediction is the INTERACTS relation between drugs. We subsample the tens of thousands of labeled interaction and shuffle the remaining labeled DDI links into five folds for cross-validation. Each fold contains almost 2000 labeled DDI targets. We alternate using one fold of DDI edges as observations, one for training and one for held-out evaluation.

Freebase Our third dataset comes from the Freebase knowledge graph and is well-used in validating knowledge base (KB) completion tasks [50]. We study KB completion for two relations: links from films to their ratings (FILMRATING$(\cdot)$) and links from authors to books written (BOOKAUTHOR$(\cdot)$). The remaining relations in the KB are observed. For both target relations, we subsample edges and split the resultant edges into five folds for cross-validation, yielding 1000 labeled edges per fold.

7.5.2 Experimental Setup

Our first experimental question evaluates predictive performance using area under the ROC curve (AUC) on held-out data with five-fold cross-validation across

the five tasks described above. Our second question validates scalability by comparing running-times for both methods as the number of clauses grows. For both methods, we use ADMM inference implemented in the probabilistic soft logic (PSL) framework [7]. For GLS, we use the pseudo-likelihood learning algorithm in PSL and implement its corresponding scoring function within in PSL [1]. For PPLL, we implement the parallelized learning algorithm in PSL. For all tasks, we enumerate target relational paths using the BFS utility in the Path Ranking Algorithm (PRA) [2] [85, 49, 50] and generate path-constrained clauses from these paths. PRA generates and includes the inverses of all atoms when performing BFS. To form clause literals from these inverses, we use the original predicate and reverse the order of its variablized arguments.

As the number of generated clauses grows, GLS becomes prohibitive as we show in our scalability results and necessitates a clause-pruning strategy. We prune the set of clauses by retaining those that connect at least 10 target atoms and select the top 50 clauses by number of targets connected. For each target predicate t_i in the prediction tasks detailed above, we also add a negative prior clause $\neg t_i(\cdot)$ to the candidate clauses. For link prediction tasks, the negative prior captures the intuition that true positive links are rare and most links do not form. We refer the reader to [7] for detailed discussion on the importance of negative priors. For the biological citation networks and Freebase settings, we subsample negative examples of the targets to mitigate the imbalance in labeled training data. We perform 150 iterations of gradient descent for PPLL and 15 for GLS since it requires several rounds of weight learning.

[1] `psl.linqs.org`
[2] `github.com/matt-gardner/pra`

Figure 7.1: Running times (in seconds) in log scale on Freebase tasks. PPLL consistently scales more effectively than GLS.

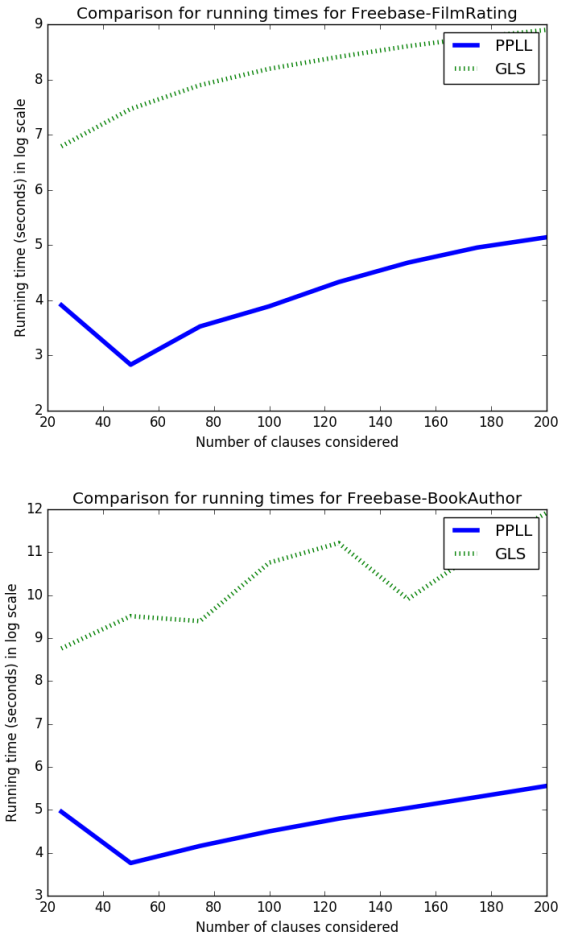


Table 7.1: Average AUC of methods across five prediction tasks. Bolded numbers are statistically significant at $\alpha = 0.05$. We show that PPLL training improves over GLS in three out of five settings.

Setting	GLS	PPLL
Fly-GENE	0.95 ± 0.01	**0.97 ± 0.002**
Yeast-GENE	0.86 ± 0.02	**0.90 ± 0.003**
DDI-INTERACTS	0.66 ± 0.06	**0.76 ± 0.01**
Freebase-FILMRATING	0.65 ± 0.04	0.65 ± 0.05
Freebase-BOOKAUTHOR	0.67 ± 0.03	0.65 ± 0.04

7.5.3 Predictive Performance

Our first experimental question investigates the ramifications of approximations made by each method on predictive performance. We contrast PPLL, which decouples the optimization over both clauses and target variables against GLS, which greedily maximizes the pseudolikelihood approximation that only factorizes across target variables. We first generate path-constrained clauses as input to both methods and evaluate their performance on held-out data. Table 7.1 compares both methods using AUC for all five prediction tasks averaged across multiple folds and splits.

Table 7.1 shows that PPLL gains significantly in AUC over GLS in three out of five settings. These results suggest that the fully factorized PPLL objective maintains and even improves predictive performance over greedy optimization of pseudolikelihood. For the GENE link prediction task in the Yeast and Fly biological citation networks, PPLL also yields lower variance given the same rules. In the DDI setting where we predict INTERACTS links between drugs, PPLL enjoys a 15% AUC gain over GLS from 0.66 to 0.76. In the Freebase setting, for both prediction tasks, FILMRATING and BOOKAUTHOR, both methods achieve comparable performance, with GLS seeing a 0.02 AUC gain in predicting FILMRATING.

Table 7.2: Average AUC of similarity-based approaches to DDI trained with different weight learning methods. We see that the DDI model learned with PPLL significantly improves over all configurations of the similarity-based models.

Learning Method	With relational path similarity	Without relational path similarity
Maximum likelihood	0.59 ± 0.01	0.69 ± 0.01
Pseudolikelihood	0.58 ± 0.01	0.69 ± 0.01
PPLL	0.56 ± 0.03	0.58 ± 0.02

7.5.4 Comparisons against DDI Similarity-based Models

To further validate the advantages of learning complex relational dependencies directly from data, we compare the cross-validation performance of clauses learned with PPLL to the state-of-the-art similarity-based PSL models proposed in Chapter 4. We extend the current Drugbank DDI dataset with five drug-drug similarity matrices obtained from a publicly available source [3]. Four out of the five similarities measure chemical similarity between drugs using different hashing functions and features to represent drugs, as described in detail in Section 4.2. The fifth similarity captures a flattened representation of the relational paths over the full data graph found by PRA, which the structure learning methods use to generate candidate clause. The similarity measure instead computes the reachability between two drugs based on the number of relational paths that connect them.

In our evaluation, we consider two variants of the similarity-based DDI model: with and without the relational path-based similarity measure. We also train these models with three different learning objectives: PPLL, pseudolikelihood, and maximum likelihood estimation (MLE). Table 7.2 shows the average AUC results across the same folds used to evaluate PPLL structure learning for both modeling variants and all training methods. Direct comparison of these results

[3]https://starling.utdallas.edu/datasets/ddi/

against the performance pf PPLL structure learning in Table 7.1 on DDI reveals that the learned clauses significantly outperform all configurations of the similarity-based approaches. Interestingly, a closer look at Table 7.2 shows that the flattened relational path similarity worsens performance regardless of the chosen learning algorithm, and the chemical similarities used on their own give the best performance with an AUC of 0.69 when trained with pseudolikelihood or MLE. The results of this comparison against a well-validated, state-of-the-art PSL model substantiate the benefits of learning complex relational structure from data. The findings suggest that even transforming the relational paths into a similarity metric remain weaker than directly learning weighted clauses.

7.5.5 Scalability Study

Our second experimental question focuses on the scalability trade-offs made by GLS and PPLL. PPLL requires only weight learning over clauses, made faster with parallelized updates while GLS requires iterative rounds of weight learning and model evaluation. We select the two Freebase tasks, BookAuthor and FilmRating where path-constrained clause generation initially yielded several hundred rules. We plot the running time for both methods as the size of the candidate clause set increases from 25 to 200.

Figure 7.1 shows the running times (in seconds) for both methods plotted in log scale across the two Freebase tasks as the number of clauses to evaluate increases. The results show that while PPLL remains computationally feasible as the number of clauses increases, GLS quickly becomes intractable as the clause set grows. Indeed, for BookAuthor, GLS requires almost two days to learn a model with 200 candidate clauses. In contrast, PPLL completes in four minutes using 200 clauses in the same setting. PPLL overcomes the requirement of interleaving

weight learning and scoring while also admitting parallel weight learning updates, boosting scalability. The results suggest that PPLL can explore a larger space of models in significantly less time.

7.6 Related Work

Finally, we review related work on structure learning approaches for MLNs and Markov random fields. We also provide an overview of work in relational information retrieval which motivates our path-constrained clause generation. There is extensive work on learning logical clauses [32, 33, 102] and structure learning for other SRL methods such as relational dependency networks [105, 106] or ProbLog [11] which we do not review here.

For general Markov random fields (MRF) and their conditional variants, structure learning typically induces feature functions represented as propositional logical clauses of boolean attributes [95, 30]. An approximate model score is optimized with a greedy search that iteratively picks clausal feature functions to include while refining candidate features by adding, removing or negating literals to single-literal clauses. MRF structure learning is also viewed as a feature selection problem solved by performing L1-regularized optimization over candidate features, admitting fast gradient descent and online algorithms [115, 172].

Although structure learning has not been studied in PSL, many algorithms have been proposed to learn MLNs. The initial approach to MLN structure learning performs greedy beam search to grow the set of model clauses starting from single-literal clauses. The clause generation performs all possible negations and additions to an existing set of clauses while the search procedure iteratively selects clauses to refine. To efficiently guide the search towards useful models, bottom-up approaches generate informative clauses by using relational paths to capture pat-

terns and motifs in the data [97, 81, 82]. This relational path mining in bottom-up approaches is related to the path ranking algorithm (PRA) for relational information retrieval [85]. PRA performs random walks or breadth-first traversal on relational data to find useful path-based features for retrieval tasks [85, 49, 50]. Wang et al. [161] similarly use PRA to learn logical clauses as relational features for inferring new facts in knowledge bases. Finding patterns in relational has also been applied in exploiting symmetries to speed up loopy belief propagation, which significantly speeds up the training of relational models [3].

Most recently, MLN structure learning has been viewed from the perspectives of moralizing learned Bayesian networks [76] and functional gradient boosting [77, 78]. These methods improve scalability while maintaining predictive performance. Moreover, Khot et al. [78] propose an EM-style structure learning method to overcome the closed-world assumption, and the show the benefits of learning from partially observed relational data. Recently, Van Haaren et al. [154] propose lifted structure learning for MLNs, leveraging symmetry in logical clauses to speed up learning.

Alternately, approaches have been proposed to learn MLNs for target variables specific to a task of interest as we do for PSL. Structure learning methods for particular tasks use inductive logic programming [102] to generate clauses which are pruned with L1-regularized learning [60, 61] or perform iterative local search [14] to refine rules with the operations described above.

7.7 Discussion

In this work, we formalize the structure learning problem for PSL and introduce an efficient-to-optimize and convex surrogate objective function, PPLL. We unify scalable optimization with data-driven path-constrained clause genera-

tion. Compared to the straightforward but inefficient greedy local search method, PPLL remains scalable as the space of candidate rules grows and demonstrates good predictive performance across five real-world tasks. Although we focus on PSL in this work, our PPLL method can be generalized for MLNs and other SRL frameworks. An important line of future work for PSL structure learning is extending L1-regularized feature selection and functional gradient boosting approaches which have been applied successfully to MRFs and MLNs. These methods have been shown to scale while maintaining good predictive performance.

Chapter 8

Conclusion and Future Work

In this thesis, I have introduced new probabilistic frameworks for computational science. I focus on problems in both social and biological sciences throughout my work. I contrast the observational data such as social media interactions which supports socio-behavioral questions with experimental data such as gene expression measurements which facilitate biological inferences. I formulate three desiderata for applying probabilistic models to these types of data: 1) handling interdependencies in the domain; 2) fusing multiple sources of information; 3) supporting the discovery of both causal knowledge and complex, long-range patterns that inform modeling. I show how both types of data in computational science tasks can be cast as relational data graphs, useful and well-studied abstractions which motivate the use of PSL, a structured probabilistic framework which admits exact and efficient MAP inference. As the key contributions of my thesis, I build on PSL to develop a unified framework for computational science with four necessary developments:

1. Useful structural patterns that extend across multiple domains, developed by comprehensively evaluating several modeling choices.

2. Modeling patterns for fusing multiple sources of information with collective reasoning.

3. Methods that support causal inference and discovery, incorporating textual modalities of data.

4. Algorithms for learning PSL model structure directly from data.

These foundational contributions are validated on socio-behavioral and biological tasks to demonstrate the empirical advantages of my work for computational science. In Chapter 3, I show that joint author-level modeling combined with a learning algorithm that mitigates label imbalance achieves best gains in predicting users' stance in online debates. In Chapter 4, my proposed similarity-fusion method for drug-drug interaction outperforms state-of-the-art non-collective method and single-similarity collective variants. In Chapter 5, I introduce the CausPSL causal structure discovery approach which enjoys scalability and robustness benefits in inferring gene tegulatory networks over competing methods. I demonstrate complementary approaches for causal inference in Chapter 6 which incorporate text data to better understand the effects of exercise on user mood and debate styles on user sentiment. In Chapter 7, I propose scalable structure learning approaches for PSL learn complex drug-interaction models that outperform similarity-fusion approach.

8.1 Open Challenges

While the contributions in this thesis lay the groundwork for addressing computational science tasks, I highlight important limitations of this work which motivate several open research problems. Broadly, these limitations span challenges in modeling latent variables to learning from incomplete or partially observed

data. I formalize these problems below and identify future work which builds on my contributions to develop more sophisticated models.

Incomplete Data Settings. In Chapter 3 and Chapter 4, I present structured modeling templates for capturing patterns such as homophily and disagreement while fusing multiple local signals such as text or evidence of similarity between drugs. However, in both settings, to learn the relative reliabilities of various similarity measures or to learn in the presence of imbalanced debate forum data, I assume that the training data are fully observed and belong to two classes. In predicting drug interactions, I make a closed-world assumption that in the training data, unobserved interactions are negative examples. In modeling debate, I treat stance as belonging to PRO and ANTI, and require training labels for both user stances and disagreement links. In both cases, these assumptions can be limiting and prohibitive. In the drug interaction setting, unobserved interactions may actually indicate that a particular drug-drug combination has simply not been tested yet and treating this link as a non-edge can hurt the model's performance on unseen data. In inferring user stance, obtaining labels can be costly, especially as new topics emerge. Learning from weaker and cheap-to-obtain signals of stance, or bootstrapping from a small set of reliable labels will enable modeling on a wider scale and range of debate forum sites.

Latent Representations. In a similar vein to learning from incomplete or partially observed data, another open problem only briefly addressed by my work lies in explicitly learning representations of latent variables in a domain. Chapter 5 and Chapter 6 present methods for inferring both graphs of causal relationships and single cause-and-effect outcomes. When learning causal graphs from observational data, I follow several other methods in assuming that no latent

confounders are present in the data. When estimating causal effects on a single outcome, I typically model confounders from fully observed text using shallow and high-dimensional representations such as word counts. In Section 6.5, I show the benefits of latent distributional representations of posts when modeling changes in dialogue from reply styles, but do not learn hierarchical or deeper representations on confounders. In both settings, latent confounding can bias or even nullify the findings of the constraints used by CausPSL or propensity score matching. For improved causal discovery and inference, it is important to learn causal graphs that explicitly model latent confounding, and find latent representations of confounders from text data when estimating effects.

Similar to learning causal structure, Chapter 7 deals with discovering the clausal structure of PSL models. As in Chapter 4, the structure learning approaches proposed in this chapter make a closed-world assumption that all unobserved relations are negative examples. Moreover, I only learning from the relations, or predicates, present in the observational data. An open challenge in the structure learning setting is to learn latent representations of unobserved relations in the data, resulting in more compact and accurate models.

8.1.1 Future Work

Motivated by the open challenges I describe above, I outline three areas of future research that overcome the limiting assumptions of this thesis and support richer probabilistic modeling. I briefly review current research advances in each of these tasks, and identify novel technical directions that build on my work. These open areas span semi-supervised approaches to relational representation learning.

Semi-supervision in Computational Science. In socio-behavioral domains such as modeling user stance from social media or detecting textual sentiment,

advances have been made in learning from a limited corpus of labeled data or from weak signals that provide model supervision. Johnson and Goldwasser [69], Ebrahimi et al. [40] propose approaches based on relational bootstrapping, where training data is iteratively labeled by a relational classifier, for stance classification on new topics. Qadir and Riloff [121] apply bootstrapping to model emotion from Twitter data. In both biological and social science tasks based on text, substantial work has studied learning from positive and unlabeled data, where unseen links are not assumed to be negative examples [42, 166, 87]. This challenge is referred to as positive-unlabeled (PU) learning. All of these methods broadly fall under semi-supervised learning, which combines limited training data with strategies to learn from weak signals or completely unlabeled data.

Building on these existing approaches, I envision an open research agenda in extending PU learning and weakly supervised learning approaches for structured PSL models of computational science domains. Johnson and Goldwasser [69], Ebrahimi et al. [40] already propose relational bootstrapping using PSL, but there remain open problems in combining latent variables such as ideology to learn stances on unseen topics based on existing training data. Although Elkan and Noto [42], Li and Liu [87] make connections between PU learning and expectation-maximization (EM) algorithms for hidden data, proposing a PU learning algorithm for relational PSL models is a novel problem. Each of the above directions of future work in semi-supervised learning approaches improve the models I propose for drug-drug interaction and debate stance prediction.

Learning Representations of Latent Confounders. In causal inference and discovery, two separate classes of approaches support the representation and inference over latent confounding variables. For discovering causal structure, several methods use a coarser representation of causality called ancestral graphs which

allow edges that represent latent confounding [93, 170, 24]. Magliacane et al. [93] and Claassen and Heskes [24] exploit logical constraints which characterize these graphs to orient ancestral edges. In a different vein, for causal inference, Louizos et al. [90] recently propose a deep neural network approach to learn representations of latent confounders from observational data. Wang and Blei [162] and Ranganath and Perotte [124] estimate the effects of multiple causes simultaneously by learning latent factor models of confounding from data.

These advances are promising for make progress in estimating both causal graphs and single cause-effect links from observational data alone, especially with modalities such as text. However, much current work is applied to observed measurements of variables instead of combining or using text data only. Another important direction of future research lies in addressing the limitations of Chapter 6 by learning latent confounders from text data. In the case of mood modeling, a latent variable model might discover categories of words in users' text entries that correspond unmeasured factors such as health, family relationships, or intrinsic mental well-being. In the dialogue analysis setting, text from debates may yield latent groupings of words that represent ideologies or the key facets of a topic. This representation learning from text can be fused with causal graph discovery from statistical tests to improve the understanding of confounding edges which the ancestral graph represents. The interpretability benefits of latent confounder representation learning can be extended to both causal inference in the social sciences and causal discovery in biological settings.

Relational Representation Learning. In learning relational models such as Markov logic networks, many approaches have addressed the representation learning challenge of discovering latent relations or predicates from the observed relational data graph. Kok and Domingos [80], Popescul and Ungar [119] cluster

relations to discover representations of new predicates, a task called statistical predicate invention. Recently, Dumančić and Blockeel [36] formalize this problem as unsupervised relational representation learning and show the advantages of using unseen predicates within structure learning. The final area of fruitful future work I envision is extending relational representation learning for PSL, and going beyond clustering of relations. An important research direction is to build upon the functional gradient boosting approach of Khot et al. [78] to learn neural representations of PSL feature functions. The advantages of this approach would allow interpretable logical clauses to be combined with non-linear functions that can capture more complex combinations of observed relations to represent unseen predicates. This novel contribution would support the learning of more sophisticated models of social science and biology.

Bibliography

[1] Rob Abbott, Marilyn Walker, Jean E. Fox Tree, Pranav Anand, Robeson Bowmani, and Joseph King. How can you say such things?!?: Recognizing disagreement in informal political argument. In *ACL Workshop on Language and Social Media*, 2011.

[2] Amjad Abu-Jbara and Dragomir R Radev. Identifying opinion subgroups in Arabic online discussions. In *ACL*, 2013.

[3] Babak Ahmadi, Kristian Kersting, Martin Mladenov, and Sriraam Natarajan. Exploiting symmetries for scaling loopy belief propagation and relational training. *Machine learning*, 92(1):91–132, 2013.

[4] Nir Atias and Roded Sharan. An algorithmic framework for predicting side effects of drugs. *J. Comput. Biol.*, 18:207–218, 2011.

[5] Stephen Bach. *Unifying MAX SAT, Local Consistency Relaxations, and Soft Logic with Hinge-Loss Markov Random Fields*. PhD thesis, University of British Columbia, 2015.

[6] Stephen H. Bach, Bert Huang, Ben London, and Lise Getoor. Hinge-loss Markov random fields: Convex inference for structured prediction. In *Uncertainty in Artificial Intelligence (UAI)*, 2013.

[7] Stephen H Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. Hinge-loss Markov random fields and probabilistic soft logic. *Journal of Machine Learning Research*, 18(109):1–67, 2017.

[8] Alexandra Balahur, Zornitsa Kozareva, and Andres Montoyo. Determining the polarity and source of opinions expressed in political debates. *Computational Linguistics and Intelligent Text Processing*, 2009.

[9] Mohit Bansal, Claire Cardie, and Lillian Lee. The power of negative thinking: Exploiting label disagreement in the min-cut classification framework. *COLING*, 2008.

[10] Mark Bartlett and James Cussens. Integer linear programming for the Bayesian network structure learning problem. *Artificial Intelligence*, 244: 258–271, 2017.

[11] Elena Bellodi and Fabrizio Riguzzi. Structure learning of probabilistic logic programs by searching the clause space. *Theory and Practice of Logic Programming*, 15(2):169–212, 2015.

[12] Islam Beltagy, Katrin Erk, and Raymond J Mooney. Probabilistic soft logic for semantic textual similarity. In *ACL*, 2014.

[13] Julian Besag. Statistical analysis of non-lattice data. *The statistician*, pages 179–195, 1975.

[14] Marenglen Biba, Stefano Ferilli, and Floriana Esposito. Discriminative structure learning of Markov logic networks. In *ILP*, 2008.

[15] Kevin Bleakley and Yoshihiro Yamanishi. Supervised prediction of drug–target interactions using bipartite local models. *Bioinformatics*, 25:2397–2403, 2009.

[16] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[17] Amber E Boydstun, Justin H Gross, Philip Resnik, and Noah A Smith. Identifying media frames and frame dynamics within and across policy issues. In *New Directions in Analyzing Text as Data Workshop, London*, 2013.

[18] Clinton Burfoot, Steven Bird, and Timothy Baldwin. Collective classification of congressional floor-debate transcripts. In *ACL*, 2011.

[19] D-S Cao, N Xiao, Y-J Li, W-B Zeng, Y-Z Liang, A-P Lu, Q-S Xu, and AF Chen. Integrating multiple evidence sources to predict adverse drug reactions based on a systems pharmacology model. *CPT: Pharmacometrics Syst. Pharmacol.*, 4:498–506, 2015.

[20] Feixiong Cheng and Zhongming Zhao. Machine learning-based prediction of drug–drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *Journal of the American Medical Informatics Association*, 21:e278–e286, 2014.

[21] Feixiong Cheng, Chuang Liu, Jing Jiang, Weiqiang Lu, Weihua Li, Guixia Liu, Weixing Zhou, Jin Huang, and Yun Tang. Prediction of drug-target interactions and drug repositioning via network-based inference. *PLoS Comput Biol*, 8:e1002503, 2012.

[22] David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2003.

[23] Ting-Chao Chou. Drug combination studies and their synergy quantification using the chou-talalay method. *Cancer research*, 70:440–446, 2010.

[24] Tom Claassen and Tom Heskes. A logical characterization of constraint-based causal discovery. In *UAI*, 2011.

[25] Diego Colombo and Marloes H Maathuis. Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15: 3741–3782, 2014.

[26] N Renee Crowther, Anne M Holbrook, Robert Kenwright, and Margaret Kenwright. Drug interactions among commonly used medications. chart simplifies data from critical literature review. *Canadian Family Physician*, 43:1972, 1997.

[27] James Cussens. Bayesian network learning with cutting planes. In *UAI*, pages 153–160, 2011.

[28] Cristian Danescu-Niculescu-Mizil, Michael Gamon, and Susan Dumais. Mark my words!: linguistic style accommodation in social media. In *Proceedings of the 20th international conference on World wide web*, pages 745–754. ACM, 2011.

[29] Cristian Danescu-Niculescu-Mizil, Lillian Lee, Bo Pang, and Jon Kleinberg. Echoes of power: Language effects and power differences in social interaction. In *Proceedings of the 21st international conference on World Wide Web*, pages 699–708. ACM, 2012.

[30] Jesse Davis and Pedro Domingos. Bottom-up learning of Markov network structure. In *ICML*, 2010.

[31] Cassio P De Campos and Qiang Ji. Efficient structure learning of Bayesian networks using constraints. *Journal of Machine Learning Research*, 12:663–689, 2011.

[32] Luc De Raedt and Kristian Kersting. Probabilistic inductive logic programming. In *Probabilistic Inductive Logic Programming*, pages 1–27. Springer, 2008.

[33] Luc De Raedt and Ingo Thon. Probabilistic rule learning. In *International Conference on Inductive Logic Programming*, pages 47–58. Springer, 2010.

[34] Rui Dong, Yizhou Sun, Lu Wang, Yupeng Gu, and Yuan Zhong. Weakly-guided user stance prediction via joint modeling of content and social interaction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1249–1258. ACM, 2017.

[35] Virgile Landeiro Dos Reis and Aron Culotta. Using matched samples to estimate the effects of exercise on mental health from twitter. In *AAAI Conference on Artificial Intelligence*, 2015.

[36] Sebastijan Dumančić and Hendrik Blockeel. Demystifying relational latent representations. In *International Conference on Inductive Logic Programming*, pages 63–77. Springer, 2017.

[37] Daniel Eaton and Kevin P Murphy. Exact Bayesian structure learning from uncertain interventions. In *AISTATS*, 2007.

[38] Imme Ebert-Uphoff and Yi Deng. Causal discovery for climate research using graphical models. *Journal of Climate*, 25(17):5648–5665, 2012.

[39] Javid Ebrahimi, Dejing Dou, and Daniel Lowd. A joint sentiment-target-stance model for stance classification in tweets. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2656–2665, 2016.

[40] Javid Ebrahimi, Dejing Dou, and Daniel Lowd. Weakly supervised tweet stance classification by relational bootstrapping. In *EMNLP*, 2016.

[41] Sean Ekins and Steven A Wrighton. Application of in silico approaches to predicting drug–drug interactions. *J. Pharmacol. Toxicol. Methods*, 45: 65–69, 2001.

[42] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 213–220. ACM, 2008.

[43] Varun Embar, Dhanya Sridhar, Golnoosh Farnadi, and Lise Getoor. Scalable structure learning for probabilistic soft logic. In *Workshop on Statistical Relational AI*, 06/2018 2018.

[44] Shobeir Fakhraei, Louiqa Raschid, and Lise Getoor. Drug-target interaction prediction for drug repurposing with probabilistic similarity logic. In *ACM SIGKDD 12th International Workshop on Data Mining in Bioinformatics (BIOKDD)*, 2013.

[45] Shobeir Fakhraei, Bo Huang, Louiqa Raschid, and Lise Getoor. Network-based drug-target interaction prediction with probabilistic soft logic. *Comput. Biol. Bioinf.*, 11:775–787, 2014.

[46] Shobeir Fakhraei, Eberechukwu Onukwugha, and Lise Getoor. Data analytics for pharmaceutical discoveries. In *Healthcare Data Analytics*, pages 599–623. CRC Press, 2015.

[47] Nir Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799–805, 2004.

[48] Cindy Gallois and Howard Giles. Communication accommodation theory. *The international encyclopedia of language and social interaction*, pages 1–18, 2015.

[49] Matt Gardner, Partha Pratim Talukdar, Bryan Kisiel, and Tom Mitchell. Improving learning and inference in a large knowledge-base using latent syntactic cues. 2013.

[50] Matt Gardner, Partha Pratim Talukdar, Jayant Krishnamurthy, and Tom Mitchell. Incorporating vector space similarity in random walk inference over knowledge bases. In *EMNLP*, 2014.

[51] Lise Getoor and Ben Taskar. *Introduction to statistical relational learning*. MIT press, 2007.

[52] Howard Giles and Susan C Baker. Communication accommodation theory. *The international encyclopedia of communication*, 2008.

[53] Kevin Gimpel and Noah A Smith. Softmax-margin crfs: Training log-linear models with cost functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 733–736. Association for Computational Linguistics, 2010.

[54] Mehmet Gönen. Predicting drug–target interactions from chemical and genomic kernels using bayesian matrix factorization. *Bioinformatics*, 28:2304–2310, 2012.

[55] Assaf Gottlieb, Gideon Y Stein, Yoram Oron, Eytan Ruppin, and Roded Sharan. Indi: a computational framework for inferring drug interactions and their associated recommendations. *Mol. Syst. Biol.*, 8:592, 2012.

[56] Kazi Saidul Hasan and Vincent Ng. Stance classification of ideological debates: Data, models, features, and constraints. *International Joint Conference on Natural Language Processing*, 2013.

[57] Kazi Saidul Hasan and Vincent Ng. Why are you taking this stance? Identifying and classifying reasons in ideological debates. In *EMNLP*, 2014.

[58] Victoria Hollis, Artie Konrad, Aaron Springer, Matthew Antoun, Christopher Antoun, Rob Martin, and Steve Whittaker. What does all this data mean for my future mood? actionable analytics and targeted reflection for emotional well-being. *Human–Computer Interaction*, 32(5-6):208–267, 2017.

[59] Jialiang Huang, Chaoqun Niu, Christopher D Green, Lun Yang, Hongkang Mei, and JD Han. Systematic prediction of pharmacodynamic drug-drug interactions through protein-protein-interaction network. *PLoS Comput Biol*, 9:e1002998, 2013.

[60] Tuyen N. Huynh and Raymond J. Mooney. Discriminative structure and parameter learning for Markov logic networks. In *ICML*, 2008.

[61] Tuyen N Huynh and Raymond J Mooney. Online structure learning for markov logic networks. In *ECML-PKDD*, 2011.

[62] Antti Hyttinen, Patrik O Hoyer, Frederick Eberhardt, and Matti Jarvisalo. Discovering cyclic causal models with latent variables: A general sat-based procedure. In *UAI*, 2013.

[63] Antti Hyttinen, Frederick Eberhardt, and Matti Järvisalo. Constraint-based causal discovery: Conflict resolution with answer set programming. In *UAI*, 2014.

[64] Ellen Isaacs, Artie Konrad, Alan Walendowski, Thomas Lennig, Victoria Hollis, and Steve Whittaker. Echoes from the past: how technology mediated reflection improves well-being. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1071–1080. ACM, 2013.

[65] Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. Political ideology detection using recursive neural networks. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1113–1122, 2014.

[66] Tommi Jaakkola, David Sontag, Amir Globerson, and Marina Meila. Learning Bayesian network structure using LP relaxations. In *AISTATS*, 2010.

[67] Aaron Jaech, Vicky Zayats, Hao Fang, Mari Ostendorf, and Hannaneh Hajishirzi. Talking to the crowd: What do people react to in online discussions? *POLITICS*, 7:23–7.

[68] Guangxu Jin, Hong Zhao, Xiaobo Zhou, and Stephen TC Wong. An enhanced petri-net model to predict synergistic effects of pairwise drug combinations from gene microarray data. *Bioinformatics*, 27:i310–i316, 2011.

[69] Kristen Johnson and Dan Goldwasser. "All I know about politics is what I read in Twitter": Weakly supervised models for extracting politicians' stances from Twitter. In *COLING*, 2016.

[70] Kristen Johnson and Dan Goldwasser. Classification of moral foundations in microblog political discourse. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 720–730, 2018.

[71] Daniël M Jonker, Sandra AG Visser, Piet H van der Graaf, Rob A Voskuyl, and Meindert Danhof. Towards a mechanism-based analysis of pharmacodynamic drug–drug interactions in vivo. *Pharmacol. Ther.*, 106:1–18, 2005.

[72] TS Verma Judea Pearl. Equivalence and synthesis of causal models. In *UAI*, 1991.

[73] Seyed Mehran Kazemi, David Buchman, Kristian Kersting, Sriraam Natarajan, and David Poole. Relational logistic regression. In *KR*. Vienna, 2014.

[74] Michael J Keiser, Vincent Setola, John J Irwin, Christian Laggner, Atheir I Abbas, Sandra J Hufeisen, Niels H Jensen, Michael B Kuijer, Roberto C Matos, Thuy B Tran, et al. Predicting new molecular targets for known drugs. *Nature*, 462:175–181, 2009.

[75] Kristian Kersting and Luc De Raedt. Basic principles of learning bayesian logic programs. In *Probabilistic Inductive Logic Programming*, pages 189–221. Springer, 2008.

[76] Hassan Khosravi, Oliver Schulte, Tong Man, Xiaoyuan Xu, and Bahareh Bina. Structure learning for Markov logic networks with many descriptive attributes. In *AAAI*, 2010.

[77] Tushar Khot, Sriraam Natarajan, Kristian Kersting, and Jude Shavlik. Learning Markov logic networks via functional gradient boosting. In *ICDM*, 2011.

[78] Tushar Khot, Sriraam Natarajan, Kristian Kersting, and Jude Shavlik. Gradient-based boosting for statistical relational learning: the markov logic network and missing data cases. *Machine Learning*, 100(1):75–100, 2015.

[79] Stanley Kok and Pedro Domingos. Learning the structure of Markov logic networks. In *ICML*, 2005.

[80] Stanley Kok and Pedro Domingos. Statistical predicate invention. In *Proceedings of the 24th international conference on Machine learning*, pages 433–440. ACM, 2007.

[81] Stanley Kok and Pedro Domingos. Learning Markov logic network structure via hypergraph lifting. In *ICML*, 2009.

[82] Stanley Kok and Pedro Domingos. Learning Markov logic networks using structural motifs. In *ICML*, 2010.

[83] Daphne Koller. Probabilistic relational models. In *Inductive logic programming*, pages 3–13. Springer, 1999.

[84] Artie Konrad, Simon Tucker, John Crane, and Steve Whittaker. Technology and reflection: Mood and memory mechanisms for well-being. *Psychology of well-being*, 6(1):5, 2016.

[85] Ni Lao and William W Cohen. Relational retrieval using a combination of path-constrained random walks. *Machine learning*, 81(1):53–67, 2010.

[86] Ni Lao, Tom Mitchell, and William W Cohen. Random walk inference and learning in a large scale knowledge base. In *EMNLP*, 2011.

[87] Xiaoli Li and Bing Liu. Learning to classify texts using positive and unlabeled data. In *IJCAI*, volume 3, pages 587–592, 2003.

[88] Robert LiKamWa, Yunxin Liu, Nicholas D Lane, and Lin Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 389–402. ACM, 2013.

[89] Fei Liu, Shao-Wu Zhang, Wei-Feng Guo, Ze-Gang Wei, and Luonan Chen. Inference of gene regulatory network based on local Bayesian networks. *PLoS computational biology*, 12(8):e1005024, 2016.

[90] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Advances in Neural Information Processing Systems*, pages 6446–6456, 2017.

[91] Y. Lu, H. Wang, C. Zhai, and D. Roth. Unsupervised discovery of opposing opinion networks from forum discussions. In *CIKM*, 2012.

[92] Stephanie Lukin and Marilyn Walker. Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. *NAACL 2013*, page 30, 2013.

[93] Sara Magliacane, Tom Claassen, and Joris M Mooij. Ancestral causal inference. In *NIPS*, 2016.

[94] Daniel Marbach, Robert J Prill, Thomas Schaffter, Claudio Mattiussi, Dario Floreano, and Gustavo Stolovitzky. Revealing strengths and weaknesses of methods for gene network inference. *Proceedings of the national academy of sciences*, 107:6286–6291, 2010.

[95] Andrew McCallum. Efficiently inducing features of conditional random fields. In *UAI*, 2002.

[96] Jian-Ping Mei, Chee-Keong Kwoh, Peng Yang, Xiao-Li Li, and Jie Zheng. Drug–target interaction prediction by learning from local information and neighbors. *Bioinformatics*, 29:238–245, 2013.

[97] Lilyana Mihalkova and Raymond J Mooney. Bottom-up learning of Markov logic network structure. In *ICML*, 2007.

[98] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[99] Amita Misra and Marilyn A Walker. Topic independent identification of agreement and disagreement in social media dialogue. In *Conference of the Special Interest Group on Discourse and Dialogue*, 2013.

[100] Amita Misra, Pranav Anand, Jean E Fox Tree, and Marilyn Walker. Using summarization to discover argument facets in online idealogical dialog. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 430–440, 2015.

[101] Joris Mooij and Tom Heskes. Cyclic causal discovery from continuous equilibrium data. In *UAI*, 2013.

[102] Stephen Muggleton. Inductive logic programming. *New generation computing*, 8(4):295–318, 1991.

[103] Akiko Murakami and Rudy Raymond. Support or Oppose? Classifying positions in online debates from reply activities and opinion expressions. In *ACL*, 2010.

[104] Rita Nahta, Mien-Chie Hung, and Francisco J Esteva. The her-2-targeting antibodies trastuzumab and pertuzumab synergistically inhibit the survival of breast cancer cells. *Cancer research*, 64:2343–2346, 2004.

[105] Sriraam Natarajan, Tushar Khot, Kristian Kersting, Bernd Gutmann, and Jude Shavlik. Boosting relational dependency networks. In *Online Proceedings of the International Conference on Inductive Logic Programming 2010*, pages 1–8, 2010.

[106] Sriraam Natarajan, Tushar Khot, Kristian Kersting, Bernd Gutmann, and Jude Shavlik. Gradient-based boosting for statistical relational learning: The relational dependency network case. *Machine Learning*, 86(1):25–56, 2012.

[107] Jennifer Neville and David Jensen. Relational dependency networks. *Journal of Machine Learning Research*, 8(Mar):653–692, 2007.

[108] Alexandra Olteanu, Onur Varol, and Emre Kiciman. Distilling the outcomes of personal experiences: A propensity-scored analysis of social media. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 370–386. ACM, 2017.

[109] Kyunghyun Park, Docyong Kim, Suhyun Ha, and Doheon Lee. Predicting pharmacodynamic drug-drug interactions through signaling propagation interference on protein-protein interaction networks. *PloS one*, 10:e0140816, 2015.

[110] Acacia C Parks, Matthew D Della Porta, Russell S Pierce, Ran Zilca, and Sonja Lyubomirsky. Pursuing happiness in everyday life: The characteristics and behaviors of online happiness seekers. *Emotion*, 12(6):1222, 2012.

[111] Judea Pearl. *Causality*. Cambridge university press, 2009.

[112] Judea Pearl and Thomas S Verma. A theory of inferred causation. *Studies in Logic and the Foundations of Mathematics*, 134:789–811, 1995.

[113] Judea Pearl et al. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

[114] James W Pennebaker, Roger J Booth, and Martha E Francis. Liwc2007: Linguistic inquiry and word count. *Austin, Texas: liwc. net*, 2007.

[115] Simon Perkins, Kevin Lacker, and James Theiler. Grafting: Fast, incremental feature selection by gradient descent in function space. *Journal of Machine Learning Research*, 3(Mar):1333–1356, 2003.

[116] Liat Perlman, Assaf Gottlieb, Nir Atias, Eytan Ruppin, and Roded Sharan. Combining drug and gene similarity measures for drug-target elucidation. *J. Comput. Biol.*, 18:133–145, 2011.

[117] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society*, 78:947–1012, 2016.

[118] Hoifung Poon and Pedro Domingos. Joint inference in information extraction. In *AAAI*, volume 7, pages 913–918, 2007.

[119] Alexandrin Popescul and Lyle H Ungar. Cluster-based concept invention for statistical relational learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 665–670. ACM, 2004.

[120] Robert J Prill, Daniel Marbach, Julio Saez-Rodriguez, Peter K Sorger, Leonidas G Alexopoulos, Xiaowei Xue, Neil D Clarke, Gregoire Altan-Bonnet, and Gustavo Stolovitzky. Towards a rigorous assessment of systems biology models: the DREAM3 challenges. *PloS one*, 5:e9202, 2010.

[121] Ashequl Qadir and Ellen Riloff. Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1203–1209, 2014.

[122] Joseph Ramsey. Bootstrapping the PC and CPC algorithms to improve search accuracy. 2010.

[123] Joseph Ramsey, Jiji Zhang, and Peter L Spirtes. Adjacency-faithfulness and conservative causal inference. In *UAI*, 2006.

[124] Rajesh Ranganath and Adler Perotte. Multiple causal inference with latent confounding. *arXiv preprint arXiv:1805.08273*, 2018.

[125] Philip Resnik et al. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.(JAIR)*, 11:95–130, 1999.

[126] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62(1-2):107–136, 2006.

[127] Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

[128] Donald B Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469): 322–331, 2005.

[129] Donald B Rubin. *Matched sampling for causal effects*. Cambridge University Press, 2006.

[130] Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308:523–529, 2005.

[131] Martin EP Seligman, Tracy A Steen, Nansook Park, and Christopher Peterson. Positive psychology progress: empirical validation of interventions. *American psychologist*, 60(5):410, 2005.

[132] A Skrbo, B Begović, and S Skrbo. [classification of drugs using the atc system (anatomic, therapeutic, chemical classification) and the latest changes]. *Med. Arh.*, 58:138–141, 2003.

[133] Swapna Somasundaran and Janyce Wiebe. Recognizing stances in online debates. In *ACL and AFNLP*, 2009.

[134] Swapna Somasundaran and Janyce Wiebe. Recognizing stances in ideological on-line debates. In *NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, 2010.

[135] Peter Spirtes. An anytime algorithm for causal inference. In *AISTATS*, 2001.

[136] Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9:62–72, 1991.

[137] Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal inference in the presence of latent variables and selection bias. In *UAI*, 1995.

[138] Aaron Springer, Victoria Hollis, and Steve Whittaker. Mood modeling: accuracy depends on active logging and reflection. *Personal and Ubiquitous Computing*, pages 1–15, 2018.

[139] Dhanya Sridhar and Lise Getoor. Joint probabilistic inference of causal structure. In *22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining Workshop on Causal Discovery*, 2016.

[140] Dhanya Sridhar, Lise Getoor, and Marilyn Walker. Collective stance classification of posts in online debate forums. In *ACL Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, 2014.

[141] Dhanya Sridhar, James Foulds, Bert Huang, Lise Getoor, and Marilyn Walker. Joint models of disagreement and stance in online debate. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2015.

[142] Dhanya Sridhar, Shobeir Fakhraei, and Lise Getoor. A probabilistic approach for collective similarity-based drug–drug interaction prediction. *Bioinformatics*, 32(20):3175–3182, 2016.

[143] Dhanya Sridhar, Jay Pujara, and Lise Getoor. Scalable probabilistic causal structure discovery. 2018.

[144] Dhanya Sridhar, Aaron Springer, Victoria Hollis, Steve Whittaker, and Lise Getoor. Estimating causal effects of exercise from mood logging data. 2018.

[145] Christoph Steinbeck, Christian Hoppe, Stefan Kuhn, Matteo Floris, Rajarshi Guha, and Egon L Willighagen. Recent developments of the chemistry development kit (cdk)-an open-source java library for chemo-and bioinformatics. *Curr. Pharm. Des.*, 12:2111–2120, 2006.

[146] Arthur A Stone, Joseph E Schwartz, David Schkade, Norbert Schwarz, Alan Krueger, and Daniel Kahneman. A population approach to the study of emotion: diurnal rhythms of a working day examined with the day reconstruction method. *Emotion*, 6(1):139, 2006.

[147] Charles Sutton and Andrew McCallum. Piecewise pseudolikelihood for efficient training of conditional random fields. In *ICML*, 2007.

[148] Chenhao Tan, Lillian Lee, and Bo Pang. The effect of wording on message propagation: Topic-and author-controlled natural experiments on twitter.

[149] Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th international conference on world wide web*, pages 613–624. International World Wide Web Conferences Steering Committee, 2016.

[150] Ben Taskar, Pieter Abbeel, and Daphne Koller. Discriminative probabilistic models for relational data. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 485–492. Morgan Kaufmann Publishers Inc., 2002.

[151] Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *EMNLP*, 2006.

[152] Sofia Triantafillou and Ioannis Tsamardinos. Constraint-based causal discovery from multiple interventions over overlapping variable sets. *Journal of Machine Learning Research*, 16:2147–2205, 2015.

[153] Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65:31–78, 2006.

[154] Jan Van Haaren, Guy Van den Broeck, Wannes Meert, and Jesse Davis. Lifted generative learning of markov logic networks. *Machine Learning*, 103 (1):27–55, 2016.

[155] Santiago Vilar, Eugenio Uriarte, Lourdes Santana, Nicholas P Tatonetti, and Carol Friedman. Detection of drug-drug interactions by modeling interaction profile fingerprints. *PloS one*, 8(3):e58321, 2013.

[156] Santiago Vilar, Eugenio Uriarte, Lourdes Santana, Tal Lorberbaum, George Hripcsak, Carol Friedman, and Nicholas P Tatonetti. Similarity-based modeling in large-scale prediction of drug-drug interactions. *Nature protocols*, 9 (9):2147–2163, 2014.

[157] Marilyn Walker, Pranav Anand, Rob Abbott, Jean E. Fox Tree, Craig Martell, and Joseph King. That's your evidence?: Classifying stance in online political debate. *Decision Support Sciences*, 2012.

[158] Marilyn Walker, Pranav Anand, Robert Abbott, and Jean E. Fox Tree. A corpus for research on deliberation and debate. In *LREC*, 2012.

[159] Marilyn Walker, Pranav Anand, Robert Abbott, and Richard Grant. Stance classification using dialogic properties of persuasion. In *NAACL*, 2012.

[160] Lu Wang and Claire Cardie. A piece of my mind: A sentiment analysis approach for online dispute detection. In *ACL*, 2014.

[161] William Yang Wang, Kathryn Mazaitis, and William W Cohen. Structure learning via parameter learning. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1199–1208. ACM, 2014.

[162] Yixin Wang and David M Blei. The blessings of multiple causes. *arXiv preprint arXiv:1805.06826*, 2018.

[163] Yuhao Wang and Jianyang Zeng. Predicting drug-target interactions using restricted boltzmann machines. *Bioinformatics*, 29(13):i126–i134, 2013.

[164] Zhongyu Wei, Yang Liu, and Yi Li. Is this post persuasive? ranking argumentative comments in online forum. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 195–200, 2016.

[165] David S Wishart, Craig Knox, An Chi Guo, Savita Shrivastava, Murtaza Hassanali, Paul Stothard, Zhan Chang, and Jennifer Woolsey. Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Res.*, 34(suppl 1):D668–D672, 2006.

[166] Peng Yang, Xiao-Li Li, Jian-Ping Mei, Chee-Keong Kwoh, and See-Kiong Ng. Positive-unlabeled learning for disease gene identification. *Bioinformatics*, 28(20):2640–2647, 2012.

[167] Shuo Yang, Tushar Khot, Kristian Kersting, Gautam Kunapuli, Kris Hauser, and Sriraam Natarajan. Learning from imbalanced data in relational domains: A soft margin approach. In *Data Mining (ICDM), 2014 IEEE International Conference on*, pages 1085–1090. IEEE, 2014.

[168] Shuo Yang, Mohammed Korayem, Khalifeh AlJadda, Trey Grainger, and Sriraam Natarajan. Combining content-based and collaborative filtering for job recommendation system: A cost-sensitive statistical relational learning approach. *Knowledge-Based Systems*, 136:37–45, 2017.

[169] Changhe Yuan, Brandon M Malone, et al. Learning optimal Bayesian networks: A shortest path perspective. *Journal of Artificial Intelligence Research*, 48:23–65, 2013.

[170] Jiji Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172:1873–1896, 2008.

[171] Lei Zhang, Yuanchao Derek Zhang, Ping Zhao, and Shiew-Mei Huang. Predicting drug–drug interactions: an fda perspective. *The AAPS journal*, 11 (2):300–306, 2009.

[172] Jun Zhu, Ni Lao, and Eric P Xing. Grafting-light: fast, incremental feature selection and structure learning of markov random fields. In *KDD*, 2010.