Shapley Values and Meta-Explanations for Probabilistic Graphical Model Inference

Yifei Liu[†], Chao Chen[*], Yazheng Liu[†], Xi Zhang[†], Sihong Xie[*]
{liuyifei,LiuYZ,zhangx}@bupt.edu.cn,chc517@lehigh.edu,xiesihong1@gmail.com
[†]Key Laboratory of Trustworthy Distributed Computing and Service Ministry of Education, BUPT, Beijing, China
[*]Computer Science and Engineering, Lehigh University, Bethlehem, PA, USA

ABSTRACT

Probabilistic graphical models, such as Markov random fields (MRF), exploit dependencies among random variables to model a rich family of joint probability distributions. Inference algorithms, such as belief propagation (BP), can effectively compute the marginal posteriors for decision making. Nonetheless, inferences involve sophisticated probability calculations and are difficult for humans to interpret. Among all existing explanation methods for MRFs, no method is designed for fair attributions of an inference outcome to elements on the MRF where the inference takes place. Shapley values provide rigorous attributions but so far have not been studied on MRFs. We thus define Shapley values for MRFs to capture both probabilistic and topological contributions of the variables on MRFs. We theoretically characterize the new definition regarding independence, equal contribution, additivity, and submodularity. As brute-force computation of the Shapley values is challenging, we propose GraphShapley, an approximation algorithm that exploits the decomposability of Shapley values, the structure of MRFs, and the iterative nature of BP inference to speed up the computation. In practice, we propose meta-explanations to explain the Shapley values and make them more accessible and trustworthy to human users. On four synthetic and nine real-world MRFs, we demonstrate that GraphShapley generates sensible and practical explanations.

CCS CONCEPTS

• **Computing methodologies** → **Learning in probabilistic graphical models**.

KEYWORDS

Graphical models; explainability

ACM Reference Format:
Yifei Liu[†], Chao Chen[*], Yazheng Liu[†], Xi Zhang[†], Sihong Xie[*]. 2020. Shapley Values and Meta-Explanations for Probabilistic Graphical Model Inference. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3340531.3411881

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3411881

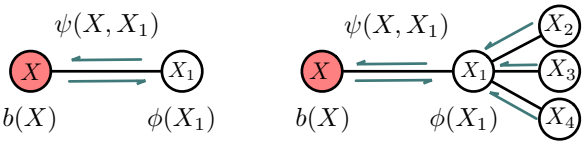

Figure 1: ***Left*: the *probabilistic* contribution of the explaining variable X_1 to the target X's belief $b(X)$ depends on the prior distribution $\phi(X_1)$ and their dependency encoded by the compatibility matrix $\psi(X, X_1)$. *Right*: the contribution of X_1 to $b(X)$ is made up of two parts: its own prior $\phi(X_1)$ and the messages from X_2, X_3, and X_4 to X_1. The Shapley value of X_1 should reflect both parts of the contribution of X_1.**

1 INTRODUCTION

Probabilistic graphical models (PGMs) play an important role in many real-world applications where dependencies between entities are essential to describe uncertain and complex interactions and dynamics. For example, fraudulent activities in review and auction networks [27, 31] and personal interests on social networks [24] can be modeled by PGMs and detected by various inference algorithms. However, the lack of explanations of the models and inferences limits the practical utility. For example, compared with "the detection" of fraudulent accounts, it is equally, if not more, important to explain "why" the detected accounts are suspicious so the users can decide if the detection is trustworthy, and if not, how to rectify the data and model for more accurate detection [33].

We focus on explaining the results produced by belief propagation (BP) on Markov Random Fields (MRF). As shown in Figure 1, an MRF is a graph that describes the dependencies among random variables (shown as circles). Each variable is assigned a prior distribution (e.g., $\phi(X)$), representing prior knowledge of the variable X without the dependencies. The dependencies among the variables are represented by the edges, each of which has compatibility parameters (e.g., $\psi(X, X_1)$ in the figure) that describe how two variables (e.g., X and X_1) interact with each other. BP computes the beliefs (posteriors) $b(X)$ of each random variable X by passing messages between variables until the messages converge. When the computed beliefs are used to aid decision making, we aim to find the elements of the MRF that contribute most to the beliefs as explanations.

Different from the explainable AI techniques that focus on independent variables, explainable MRF needs to handle the probabilistic dependencies. On the one hand, the dependencies may capture the interactions between the variables more accurately in certain situations (e.g., in spam detection, an account is more suspicious if it posts suspicious reviews to dishonest businesses). On the other hand, the dependencies compound with other elements on the MRF so that the inferred beliefs are computed using complex probability calculations and cannot be easily interpreted by human decision-makers. For

example, an Amazon reviewer account has a high belief of being a spammer since it posted reviews to some dishonest businesses, whose beliefs can further depend on thousands of reviews.

Existing explainable AI methods are insufficient in handling explainable MRF (see the surveys [8, 13]). The methods in [32, 35] assign an importance score to each feature without taking into account the dependencies between the features. On graphs, the authors of [46] explain arbitrary graph neural networks using explainer models but neither consider PGMs nor topological importance. The methods proposed in [4, 5, 7] are more relevant to PGM inference explanations: they use gradients or greedy search to find salient elements to explain beliefs. Unlike Shapley values, these methods lack a rigorous characterization of the attribution. Visualization of a graph (a knowledge graph or a graph generated by DeepWalk) via embedding [16] help users understand the semantics of a node in the graph but do not address the probabilistic dependencies between variables. Interactive graphic user interfaces for PGMs, especially Bayesian networks [18], have been developed to provide a more intuitive understanding of the models but do not compute Shapley values that are designed for attributions. Attention weights learned through an attention mechanism have been used as explanations [15], but probabilistic inference on MRFs does not involve attention weights.

On explainable MRF inferences, we argue that a more principled explanation is necessary to complement the prior work on explaining MRF inferences [4, 5, 7]. We adopt Shapley's framework, which originally attempts to provide a fair attribution of a total gain to the players in a cooperative game [34]. When used to explain a machine learning model, the explaining variables are the players, who cooperate to generate the model outputs and the attributions are considered as explanations of the outputs. Since an MRF represents a joint distribution of a set of random variables using a graph, the desired Shapley values should reflect the probabilistic and topological importance of the variables. See Figure 1 for examples. In [35], multiple explanation methods for classification are unified as Shapley value computation. In [36], they deal with Shapley and Myerson values for graph-restricted games rather than PGMs. In [6], only linear chains and grids are considered while we deal with a general graph topology. None of these Shapley value definitions quantifies both probabilistic and topological importance.

We define the Shapley value $\mathrm{SV}(X_i; X, G)$ that measures the topological and probabilistic contribution of any explaining variable X_i to the belief $b(X)$ of a target random variable X on a given MRF G. To compute $\mathrm{SV}(X_i; X, G)$, we enumerate all possible subgraphs of G where X_i and X are connected. On each subgraph, we compute the contribution of X_i to $b(X)$, measured by the change in $b(X)$ before and after X_i is omitted from the subgraph. For example, in the right panel of Figure 1, by removing X_1, the variables X_2, X_3, X_4 can not contribute to $b(X)$, which will change dramatically, and we have measured the topological importance of X_1 on this subgraph. This topological contribution cannot be measured by alternating X_1's prior only [6], as X_1 and its other connections remain in the subgraph and still play a role in message passing. These subgraphs also help measure the probabilistic importance of X_i since the prior $\phi(X_i)$ will be removed as X_i is omitted (try removing X_1 from the subgraph in the left of Figure 1). **Theoretically**, we prove several properties of the Shapley values to deliver deeper insights (Section 5). This definition can be generalized to measure the contribution of a subgraph $G_s \subset G$ as a single explaining unit. For example, the Shapley value $\mathrm{SV}(G_s; X, G)$ of the combination $G_s = (X_1, X_2)$ explains the belief $b(X)$. Note that each enumerated subgraph, though containing the target variable X, is not expected to approximate the belief of X but to measure the contribution of an explaining variable (see Eq. (6)). This is different from the prior work [5, 6] where subgraphs are used to approximate the inference on the original graphical models.

Computationally, enumerating all subgraphs can be expensive and we propose to approximate $\mathrm{SV}(X_i; X, G)$ on a small neighborhood of the target X that contains most of the salient explaining variables. We design a depth-first subgraph search algorithm called GraphShapley (Algorithm 1) that avoids duplicated enumeration while including most relevant subgraphs. By exploiting the iterative computations in BP, we retrieve the cached results from processed subgraphs to incrementally compute the messages on larger subgraphs during the DFS search. **Empirically**, on four synthetic MRFs, we provide explanations of the Shapley value explanations, termed as "meta-explanations" (Section 4) to confirm the validity of the exact and approximated Shapley values. On nine real-world MRFs, we confirm that GraphShapley is computationally efficient and can better identify influential explaining variables, compare to the gradient-based and other explanation methods (Section 6).

2 PROBLEM FORMULATION

An MRF is an undirected graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of random variables. As commonly found in multi-class classification problems [31], we assume that each random variable X_i is discrete, taking values $X_i = x_i$ from c classes $[c] = \{1, \ldots, c\}$. Each random variable $X_i \in \mathcal{V}$ has a prior distribution $\phi(x_i) \in \mathbb{R}^c$, and each edge $(X_i, X_j) \in \mathbb{R}^{c \times c}$ has the compatibility matrix $\psi(x_i, x_j)$ to encode the likelihood of X_i and X_j taking value (x_i, x_j) jointly. The graph G factorizes the joint distribution $P(\mathcal{V})$ as

$$P(\mathcal{V}) = \frac{1}{Z} \sum_{X_i \in \mathcal{V}} \phi(X_i) \prod_{X_j \in \mathcal{N}(X_i)} \psi(X_i, X_j), \tag{1}$$

where Z normalizes the product to a probability distribution, and $\mathcal{N}(X_i) = \{X_j | (X_j, X_i) \in \mathcal{E}\}$ is the neighbors of X_i.

Belief propagation (BP) is a general algorithm to compute the marginal distributions $b(X)$ by message passing. Specifically, $m_{j \to i}(x_i)$ is the message from X_j to X_i, recursively defined as:

$$m_{j \to i}(x_i) = \frac{1}{Z_i} \sum_{x_j \in [c]} \left[\psi(x_i, x_j) \phi(x_j) \prod_{k \in \mathcal{N}(X_j) \setminus \{i\}} m_{k \to j}(x_j) \right], \tag{2}$$

where Z_i is the normalization factor. The belief (marginal posterior) of X_i, denoted by $b(X_i)$, can be inferred using:

$$b(X_i) \propto \phi(X_i) \prod_{X_j \in \mathcal{N}(X_i)} m_{j \to i}(X_i). \tag{3}$$

It can be seen that $b(X_i)$ is not only related to $\phi(X_i)$, but also related to the incoming messages that recursively depend on other priors, edge potentials, and messages. This makes graph inference less transparent to a human end-users and calls for intuitive (and not necessarily exact) explanations. Prior MRF explanation methods focused on the sensitivity of the MRF parameters rather than the structures [4], or simulating the inference process using simpler

models [5]. We propose another form of explanations that answers the question of "what makes important *probabilistic* and *topological* contributions to the inference outcomes on the original MRF", rather than providing a simplified mechanics for simulating the BP inference ("simulatability" [23]).

Consider the generation of beliefs using BP as a game and all variables from $\mathcal{V}$ as players in a coalition that collectively contributes to the belief $b(X)$ of a variable $X \in \mathcal{V}$. We propose to compute Shapley values of the players to fairly attribute $b(X)$ to the variables in the coalition $\mathcal{V}$. The top few variables receiving the most attributions can be regarded as a succinct explanation of the belief $b(X)$.

The Shapley value $\text{SV}(X_i; X, G)$ of $X_i \in \mathcal{V}$ when contributing to $b(X)$ on G is the average of X_i's contributions to $b(X)$ in all possible subgraphs $S \subset G$ where X and X_i are connected. Each subgraph is a new MRF where BP can compute a new belief $\tilde{b}(X)$ of X. We define the characteristic function $v : \mathcal{S} \to \mathbb{R}$ to evaluate the quality of a coalition $S \in \mathcal{S}$ in approximating $b(X)$. As KL-divergence is a well-established measurement of approximating a probability distribution [5, 39], the symmetric KL-divergence between the two beliefs $b(X)$ and $\tilde{b}(X)$, $\text{KL}(b(X)||\tilde{b}(X)) + \text{KL}(\tilde{b}(X)||b(X))$, will measure the distance between $b(X)$ and $\tilde{b}(X)$. In particular,

$$\begin{aligned} v(S; X) &= -(\text{KL}(b(X)||\tilde{b}(X)) + \text{KL}(\tilde{b}(X)||b(X))) \\ &= -\sum_x b(x) \log[b(x)/\tilde{b}(x)] - \sum_x \tilde{b}(x) \log[\tilde{b}(x)/b(x)] \\ &= H(b) + H(\tilde{b}) - \text{NLL}(b, \tilde{b}) - \text{NLL}(\tilde{b}, b), \end{aligned} \tag{4}$$

where $H(b) = -\sum_x b(x) \log b(x)$ is the entropy of the distribution $b(X)$ and $\text{NLL}(b, \tilde{b}) = -\sum_x b(x) \log \tilde{b}(x)$ is the negative log-likelihood loss when using $\tilde{b}(X)$ to predict $b(X)$, and likewise for $H(\tilde{b})$ and $\text{NLL}(\tilde{b}, b)$. A higher v indicates that $\tilde{b}(X)$ can approximate $b(X)$ better without over-committing to a particular class, similar to the maximum entropy classifier [20]. The characteristic function in [6] is just $-\text{NLL}(b, \tilde{b})$, which is not symmetric and is a lower-bound of $-\text{KL}(b(X)||\tilde{b}(X))$. We directly evaluate the approximation quality of S without further resorting to the lower-bound.

Note: each coalition S does not aim to approximate the original belief $b(X)$ well. The role of the coalition/subgraph S is different from that in prior work [5, 6] where a subgraph works as an "explanation" of the larger MRF G and needs to approximate the MRF well. $v(S; X)$ is used to find the contribution to X. We define the *marginal contribution* of X_i to $b(X)$ when X_i works within the coalition S, as the difference in the approximation quality with and without X_i:

$$\mu(X_i; X, S) = v(S, X) - v(S \setminus \{X_i\}, X). \tag{5}$$

$\mu(X_i; X, S)$ can be positive or negative, and if the magnitude of $\mu(X_i; X, S)$ is large, then X_i makes a big difference in approximating $b(X)$ when it presents in the coalition S and when it does not. The Shapley value of X_i when contributing to $b(X)$ on G is then obtained by averaging the marginal contributions over all coalitions in $\mathcal{S}(X_i; X, G)$:

$$\text{SV}(X_i; X, G) := \frac{1}{|\mathcal{S}(X_i; X, G)|} \sum_{S \in \mathcal{S}(X_i; X, G)} \mu(X_i; X, S). \tag{6}$$

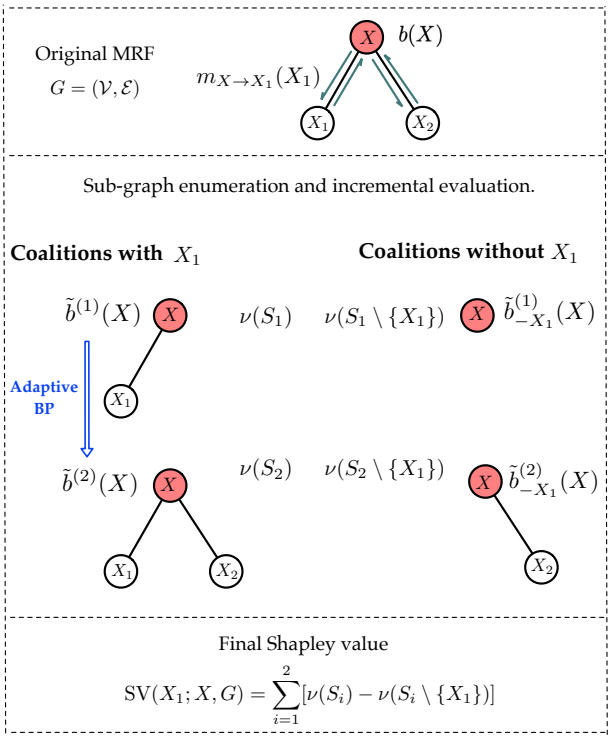

Figure 2: An example of calculating Shapley value $\text{SV}(X_1; X, G)$ on G with subgraph enumeration and incremental evaluation.

Note that Eq. (6) is not an approximation but an exact definition of Shapley values. The steps for evaluating $\text{SV}(X_1; X, G)$ is demonstrated in Figure 2 over a simple MRF.

The definition of $\text{SV}(X_i; X, G)$ can be extended to include the Shapley value of a subgraph G_s so that $\text{SV}(X_i; X, G)$ is just a special case of $\text{SV}(G_s; X, G)$, where G_s is a subgraph of G and does not contain variable X. This definition is useful to evaluate the contribution of subgraphs. For example, a user may want to know how important the connection between a reviewer and a review is to the classification of the reviewer as a spamming account [31]. We let $\text{SV}(\emptyset; X, G) = 0$ for an empty subgraph.

Computation challenges. On larger MRFs in real-world applications, computing $\text{SV}(X_i; X, G)$ is challenging since: 1) the whole set $\mathcal{S}(X_i; X, G)$ is exponentially large, 2) the enumeration all coalitions needs a carefully designed search algorithm, and 3) the evaluation of the characteristic function v on each coalition require running BP to estimate $\tilde{b}(X)$. Prior work [6, 36] computes Eq. (6) in a combinatoric manner by enumerating all possible subsets of the random variables. Variable subset enumeration does not apply to computing Eq. (6) since the same set of variables can be connected in multiple ways, depending on what edges are included in the subgraph to connect the variables. To compute $\text{SV}(X_1; X, G)$, we need a connected subgraph enumeration algorithm.

3 METHOD

To address the above challenges, we propose GraphShapley (Algorithm 1), for efficient computation of $\text{SV}(X_i; X, G)$. GraphShapley restricts the maximum search distance to approximate Shapley value

Algorithm 1: GraphShapley

Input : Graph $G = (\mathcal{V}, \mathcal{E})$, a target variable $X \in \mathcal{V}$ to be explained with its belief $b(X)$, maximum search distance D, maximum subgraph complexity C

Output : Shapley Value $\mathrm{SV}(X_i; X, G)$ for each explaining variable X_i for target variable X

1 **Function** `GraphShapley()`**:**
2 Sort $\mathcal{N}(X_i)$ by degree for each $X_i \in G$
3 $S(X, G)$ = DFSEnumerate$(G, \{X\}, X, \emptyset)$
4 **foreach** $S \in S(X, G)$ **do**
5 **foreach** $X_i \in S$ **do**
6 Evaluate $\mu(X_i; X, S) = v(S, X) - v(\setminus\{X_i\}, X)$.
7 **end**
8 **end**
9 $\mathrm{SV}(X_i; X, G) = \frac{1}{|S(X_i;X,G)|} \sum_{S \in \mathcal{S}(X_i;X,G)} \mu(X_i; X, S)$
10 **End Function**
11 **Function** `DFSEnumerate(G, Sub, v, Forbidden)`**:**
12 // *Sub*: current subgraph; v: current node to explore.
13 **foreach** $u \in \mathcal{N}(v)$ **do**
14 **if** $(v, u) \notin Fobidden \wedge u \notin Sub \wedge len(Sub) < C \wedge d(u, X) < D$ **then**
15 Record $\tilde{b}(X)$ =AdaptiveBP$(Sub, (v, u), X)$
16 DFSEnumerate$(G, Sub \cup \{(v, u)\}, u, Forbidden)$
17 $Forbidden \longleftarrow Forbidden \cup \{(v, u)\}$
18 **end**
19 **foreach** $m \in Sub \setminus \{v\}$ **do**
20 // *expand from other variables.*
DFSEnumerate$(G, Sub, m, Forbidden)$
21 **end**
22 **Return** *Sub*
23 **End Function**
24 **Function** `AdaptiveBP(S, (v, u), X)`**:**
25 // S: current MRF with converged messages.
26 // v: variable in S; u: a new variable to be added to S.
27 // Refer to [9] for more details.
28 $S \leftarrow S \cup (v, u)$
29 Use Adaptive BP to compute the new belief $\tilde{b}(X)$.
30 **Return** $\tilde{b}(X)$
31 **End Function**

Eq. (6) and only taking into account the variables that contribute more significantly. The algorithm has a systematic subgraph search to avoid duplicated enumeration and to ensure complete enumeration of the *desired* subgraphs. To further reduce the cost, GraphShapley re-uses existing BP inference results on the processed subgraphs for incremental BP inference on a larger subgraph.

DFS Subgraph Enumeration. Depth-first search makes use of the topology of graphical models to explore the desired subgraphs recursively. Due to the high cost of enumerating all subgraphs on a large graph, we consider a maximum search distance D from the target variable, beyond which the variables will not be included. Besides the distance restriction, a maximum subgraph complexity C is used to limit the size of the enumerated subgraph. This limitation on subgraph size concerns about enumeration cost rather than the simplicity in the explanations, since we don't use subgraphs as explanations as in [5]. The length of the shortest path between two nodes X and A is computed using a breadth-first search and is denoted as $d(X, A)$ in Algorithm 1. The approximation errors due to the limitations by D and C will be studied in the experiments (Figure 7).

Given an MRF G, a target variable X on G, and search parameters D and C, a divide-and-conquer technique is applied to enumerate connected acyclic subgraphs (trees). The enumeration has two parts. First, starting from the variable X, we explore the subgraphs containing further edges through depth-first search (line 13-18). Second, we expand a subgraph from other variables on the subgraph recursively (line 19-21). Forbidden edges are those that DFS has completed and will be flagged so that the DFS will not visit them in the future search. A newly explored edge will not be added to the subgraph if: 1) the edge has been flagged as forbidden, or 2) its addition will lead to a cycle in the subgraph, or 3) its addition will make the subgraph larger than the capacity C, or 4) the node to be added is D hops away from X on G. The first rule is to ensure the completeness and avoid redundancy in the enumeration, and the last two rules aim to reduce the enumeration cost. We do not process another edge until the previous edge is fully-processed, and the enumeration will be completed when all edges have been processed, The whole enumeration subgraphs $S(X, G)$ of X on G within the limits of the parameters D and C will be obtained for further computation.

A canonical ordering of the edges is determined by a breadth-first-search (BFS) on the MRF graph before running our algorithm and there are no duplicated subgraphs enumerated. The BFS also canonically numbers the nodes to avoid isomorphism test during graph lookup: the same graph will be represented by a unique array of edges with canonical node numbering. A hash table is used to record BP inference results of enumerated existing MRFs (line 15), while the new belief based on the retrieved subgraph is computed by Adaptive BP (line 24).

Figure 2 demonstrates the calculation of $\mathrm{SV}(X_1; X, G)$. Starting from the subgraph containing only the target variable $\{X\}$, S_1 is obtained through adding X_1. Exploration stops at X_1, so the algorithm returns from the recursive call at line 16 and adds (X, X_1) to forbidden edges (line 17). When the enumeration of all subgraphs is complete, the $\mu(X_1; X, S_i)$ concerning every subgraph S_i containing X_1 and Shapley value $\mathrm{SV}(X_1; X, G)$ can be computed based on Eq. (5) and Eq. (6), respectively. The algorithm finds a subgraph containing X and X_2, which is not shown as X_1 is not in the subgraph and is not relevant to $\mathrm{SV}(X_1; X, G)$.

Adaptive Belief Propagation. The contribution $\mu(X_i; X; S)$ from the enumerated subgraph S needs to find $v(S; X)$ and $v(S \setminus \{X_i\}; X)$, which require running BP on two subgraphs S and $S \setminus \{X_i\}$. Since the subgraph enumeration builds larger subgraphs upon smaller subgraphs, the constituent smaller subgraphs may have their BP inference done. We adopt *adaptive belief propagation* [9] to recycle existing inference results to speed up BP on a larger subgraph (line 28-30). For example, considering two MRFs S_1 and S_2 in Figure 2, where S_2 is enumerated after S_1. The belief $b^{(2)}(X)$ on S_2 can be computed on S_2, starting from the messages converged on the edge (X, X_1) in S_1. Experiments show that the adaptive BP reduces running time significantly (Figure 9, *left*).

Shapley Values Calculation. Rather than calling GraphShapley for each explaining variable, the enumeration is done only once for a single target X to find the Shapley values of *all* explaining variables. When the subgraph enumeration on G is completed, $\mu(X_i; X, S)$

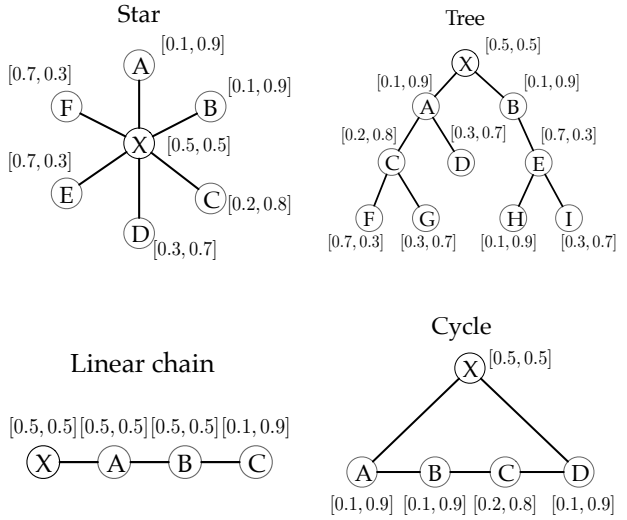

Figure 3: In the four synthetic MRFs, the variable X is the target with a uniform prior. The compatibility matrices are two-by-two, which have 0.9 on the diagonal, and have 0.1 elsewhere. The belief $b(X)$ needs to be explained by GraphShapley. The remaining variables contribute to $b(X)$ through their pre-set priors and/or their specific locations in the graphs. The Shapley values are displayed in Table 1, along with the meta-explanations.

for each explaining variable X_i will be calculated. Shapley value $\mathrm{SV}(X_i; X, G)$ will be obtained by averaging all $\mu(X_i; X, S)$ over $S \in \mathcal{S}(X_i; X, G)$ (line 9 in Algorithm 1). For example, $\mathrm{SV}(X_1; X, G)$ is the average marginal contribution of X_1 to X over the two enumerated subgraphs in Figure 2.

Shapley values of general subgraphs. When computing the Shapley values $\mathrm{SV}(G_s; X, G)$ of a subgraph G_s containing more than one variable, Algorithm 1 remains the same but the participating subgraphs in the summation of Eq. (6) will be changed to include those subgraphs where G_s and X are connected.

4 META-EXPLANATION: EXPLAINING GRAPHSHAPLEY

We view explanations of a machine learning model as communications between the model and its human users. Like other effective communication, the explanations must be transparent and trustworthy. First, the computation of Shapley values must be clear to the users to understand the meaning and the composition of the computed values. Second, the computed Shapley values must reflect what it intends to mean, rather than something else.

4.1 Why questions for meta-explanations

To convince human users to trust and adopt the proposed Shapley values for decision making, it is important to explain the Shapley values. We term such explanations of explanations as "*meta-explanations*." For this quest, we rephrase some of the *why* questions in [42] and aim at answering the questions as meta-explanations:

Table 1: Shapley values and the meta-explanations for the variables in the MRFs in Figure 3. The settings of priors, compatibility matrices are mentioned in Figure 3.

MRF	Var	SV	Meta-Explanations
Star ($b(X) = [0.2, 0.8]$)	A	0.236	$\phi(A)$ has a more positive influence over $b(X)$ than $\phi(D)$.
	D	0.124	$\phi(D)$ has a positive influence over $b(X)$.
	F	-0.14	$\phi(F)$ has a negative influence over $b(X)$.
Tree ($b(X) = [0.21, 0.79]$)	A	0.226	$\phi(A)$ has more a positive influence over $b(X)$ than $\phi(B)$ since A has children with more influence over $b(X)$.
	B	0.189	$\phi(B)$ has a positive influence over $b(X)$.
	E	-0.01	$\phi(E)$ has a negative influence over $b(X)$.
Chain ($b(X) = [0.48, 0.52]$)	A	0.007	The contribution of A to $b(X)$ is to introduce C.
	B	0.01	The contribution of B to $b(X)$ is to introduce C.
	C	0.02	$\phi(C)$ has a positive influence over $b(X)$.
Cycle ($b(X) = [0.2, 0.8]$)	A	0.174	$\phi(A)$ has the most positive influence over $b(X)$.
	B	0.02	$\phi(B)$ has a less positive influence over $b(X)$ than A due to the longer distance from X .

- Plain Fact: "*Why does variable X have this Shapley value.*" Answering this question helps the users know how a Shapley value is produced and be assured about the correctness of the value.
- P-Contrast: "*Why does variable X have a [positive/negative/zero] rather than [non-positive/non-negative/non-zero] Shapley value.*" This is a counterfactual or *what-if* question and can be answered in a similar way as the Plain Fact question.
- O-Contrast: "*Why does variable A have a [positive/negative/zero] Shapley value while variable B have a [non-positive/non-negative/non-zero] Shapley value.*" By accentuating the difference in the Shapley values of two distinct explaining variables, a user can more clearly see what contributes most to the two Shapley values.
- T-Contrast: "*Why variable A had that Shapley value and has this Shapley value now.*" Answering this question will help users understand how the change of the MRF results in the change in the Shapley values and evaluate the robustness and sensitivity of the computed Shapley values.

4.2 Answers to the why questions: a case study

There is no agreement in how the why questions can be answered, as each can be answered in many different ways [22]. We demonstrate meta-explanations using the following small synthetic MRFs, shown in Figure 3. In all models, X is the target variable whose belief $b(X)$ is to be explained. We assume there are only two classes $\{0, 1\}$ on all nodes so that all distributions are vectors of length two. The target node X has prior $\phi(X = x) = 0.5$ for $x \in \{0, 1\}$, and the priors of the explaining nodes are annotated in Figure 3.

- Star: all explaining variables $\{A, \ldots, F\}$ directly connect to the target variable X and there is no incoming message into these explaining variables except that from X, so the contribution to $b(X)$ should be just the explaining variables' priors.
- Tree: the target variable X is a root of the tree, and variables A and B directly connect to X so that A and B contribute their priors $\phi(A)$ and $\phi(B)$ to $b(X)$ by Eq. (2). The other explaining variables

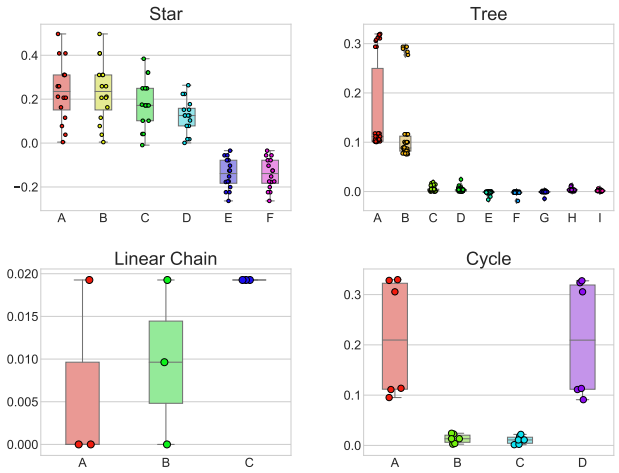

Figure 4: Meta-explanation for the Plain Fact, P-Contrast, and O-Contrast questions using the compositions of the Shapley values of all explaining variables in the MRFs in Figure 3. y-axes represent Eq. (5).

$\{C, \ldots, I\}$ are at least 2-hop away from X and messages pointing from these variables towards X will eventually be part of $b(X)$, with A and B passing messages $m_{C\to A}$, $m_{D\to A}$, and $m_{E\to B}$ to X.

- Chain: variables A and B have uniform prior distribution so their contributions to $b(X)$ are only from passing the message $m_{C\to B}$ to $b(X)$. This case differs from the Star case, where all the explaining variables contribute to their priors *only*.
- Cycles: variables $\{A, \ldots, D\}$ are in a cycle with X so that messages can enter X following two paths. BP on a cycle is called "Loopy BP" and may not be accurate [19]. Nonetheless, the Shapley values can still accurately reflect the contributions to the inaccurate $b(X)$.

4.2.1 Answering the Plain Fact, P-Contrast, and O-Contrast questions. We decompose a Shapley value into the terms that add up to the Shapley value. A visualization of the decomposition of the Shapley values for the four small MRFs is shown in four plots in Figure 4. Each variable occupies a position on an x-axis and has the contribution to its Shapley value as circles along the y-axes. A user can easily find the answer to the question "*Why A has this Shapley value*" by averaging the values along the y-axis for variable A, or by tracking the means of the circles represented by the middle bars in the box plots. The P-Contrast question "*Why does variable E have a negative rather than positive Shapley value in the star MRF*" can be answered using the same plots. The O-contrast question "*Why does variable A have a positive Shapley value while variable D has the same Shapley value in the cycle MRF*" can be answered by the subfigure at the lower right: A and D have the same terms adding up to their respective Shapley values.

Explaining the Shapley Values using "ground truth". The answers may seem over-simplified but the meta-explanation details need to be considered against information overloading [29], which can lead to less user adoption of the explanations ("*algorithm aversion*" [45]). In Table 1, we provide more detailed meta-explanations for the more technical readers. For example, on the star MRF, the question "*Why variable A has a positive Shapley value*" can be answered as: because A's prior distribution is closer to the belief of X inferred on the full MRF, and regardless of what other variables

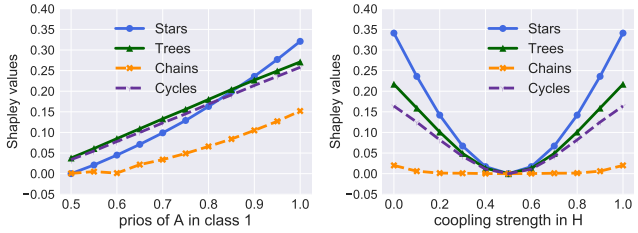

Figure 5: Meta-explanation for T-Contrast Questions on the four synthetic MRFs. *Left*: by varying the prior $\phi(A)$ towards the target variable's ground truth class, the Shapley values of A ($\mathbf{SV}(A; X, G)$) steadily go up. *Right*: by varying the values of the diagonal elements of the compatibility matrix $\psi(X, A)$, we can see the Shapley value of A ($\mathbf{SV}(A; X, G)$) are maximized on the two extremes (highest anti-homophily or homophily dependency) and minimized at zero (no dependency between A and X).

are present, the belief of X will be much more different from the original belief (measured by KL-div), when A does not present and when it does present.

4.2.2 Answering the T-Contrast questions. For questions like "*Why does variable A had a zero Shapley value then and now is having a negative Shapley value in the star MRF?*", we vary the priors of variable A in all four MRFs over time and track the changes in $\mathrm{SV}(A; X, G_i)$. A user can attribute the changes in the Shapley values to the changes in their prior distributions (Figure 5, *left*). Similarly, we vary the compatibility matrices $\psi(A, X)$ in the MRFs and one can attribute the changes in $\mathrm{SV}(A; X, G_i)$ to the varying $\psi(A, X)$ (Figure 5, *right*).

Robustness of GraphShapley with respect to input perturbations. The above answer to the T-Contrast questions can be used to study the robustness of Shapley values when the input MRF is (adversarially) perturbed, a situation that has been studied for other explanations in [1, 10, 30, 44, 48]. From Figure 5, a user can be assured that Shapley values will vary smoothly along with small perturbations (possibly out of adversarial purpose) in the input MRF parameters, such as variable priors and compatibility matrices, and the salient contributing variables can still be identified. See the caption of the figure for more details.

5 THEORETICAL ANALYSIS

We first confirm that GraphShapley conforms with the independence between the target and explaining variables.

THEOREM 5.1. (**Independence**) *If X and X_i are disconnected in G so that $X \perp\!\!\!\perp X_i$, then $SV(X_i; X, G) = 0$. Further, if X_i is connected to X but blocked by the Markov Blanket $B \subset \mathcal{V}$ of X so that $X \perp\!\!\!\perp X_i | B$, then $SV(X_i; X, G) = 0$.*

The first statement in the theorem is obvious based on the definition of $\mathrm{SV}(X_i; X, G)$: since X and X_i are disconnected in all subgraph S, so whether X_i presents or not in S, the belief of X remains the same. The second statement can be proved using the definition of $\mathrm{SV}(X_i; X, G)$ and the definition of Markov Blanket [19]. More specifically, the Markov Blanket of X ($\mathrm{MB}(X)$) is the set of immediate neighbors of X on G. Any nodes in $G - \{X\} - \mathrm{MB}(X)$ must be connected to X through some nodes in $\mathrm{MB}(X)$. Since the nodes in $\mathrm{MB}(X)$ is considered as observed (rather than a random variable),

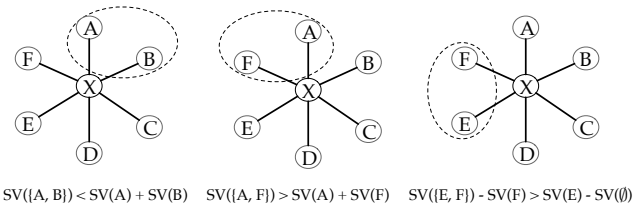

Figure 6: This star MRF is taken from Figure 3. The right-most subfigure shows a counterexample of the submodularity of Shapley values: $\mathrm{SV}(\{EF\}) - \mathrm{SV}(E) \geq \mathrm{SV}(E) - \mathrm{SV}(\emptyset)$. That is, adding the same variable E to the existing explaining variable $\{F\}$ will have a larger increase in Shapley value than adding the same E to $\emptyset \subset \{F\}$. We simplify the notation $\mathbf{SV}(A; X, G)$ to $\mathbf{SV}(A)$ to avoid clutter.

any messages from any nodes in $G - \{X\} - \mathrm{MB}(X)$ to X will be blocked by the node in $\mathrm{MB}(X)$ and have no contribution to the belief of X in any subgraph. As an example, in Figure 1 (II), if X_1 is observed, X_2 will have zero contribution of $b(X)$. On the other hand, in the MRF G in Figure 2, even if X_1 is in the Markov blanket of X, X_2 can still contribute to $b(X)$ through the path (X, X_2) (that is, X_2 is not blocked by $\mathrm{MB}(X)$).

THEOREM 5.2. (**Equal contribution**) *Given any two variables X_i and X_j, if $v(S \cup \{X_i\}; X) = v(S \cup \{X_j\}; X)$ for any coalition S, then* $\mathrm{SV}(X_i; X, G) = \mathrm{SV}(X_j; X, G)$.

This theorem is easy to prove using the definition of $\mathrm{SV}(X_i; X, G)$. A few examples are given in Figures 3 and 4.

THEOREM 5.3. (**Dummy contributors**) *If $v(S \cup \{X_i\}; X) = v(S; X)$ for any coalition S, then* $\mathrm{SV}(X_i; X; G) = 0$.

This can be proved mechanically, but its implications are more far-reaching on PGM. What kind of variables are dummy contributors? A random variable with non-uniform prior distribution is not a dummy contributor: according to Eq. (3), the prior will modulate the distribution in an out-going message emitting from that variable. A random variable with uniform prior distribution may not be a dummy contributor since it can pass messages from farther away variables to X so $v(S \cup \{X_i\}; X) \neq v(S; X)$ for some coalition S.

THEOREM 5.4. (**No Additivity**) *There exists an MRF $G = (\mathcal{V}, \mathcal{E})$ and a random variable $X \in \mathcal{V}$, such that $\sum_i \mathrm{SV}(X_i; X, G) \neq v(\mathcal{V}) - v(\emptyset)$*

The lack of additivity is due to the nonlinearity of the contribution of X_i to X, where the nonlinearity comes from the nonlinearity of the inference algorithm and the discreteness of the graph topology. In the experiment, we will demonstrate such nonlinearity. Additivity is satisfied by the Shapley values in [6], as they don't alter the MRF topology. Additivity is also satisfied by the axiomatic definition of Shapley values on a simple linear model [38], due to the lack of interactions among the variables/players. Additivity of the Shapley value in [37] is postulated as an axiom: while providing a rigorous theoretical framework, the definition limits the applicability of Shapley values to situations where the variables/players are not independently contributing to the outcome of the game [38]. It is future work to show whether additivity and topological importance measurement are compatible.

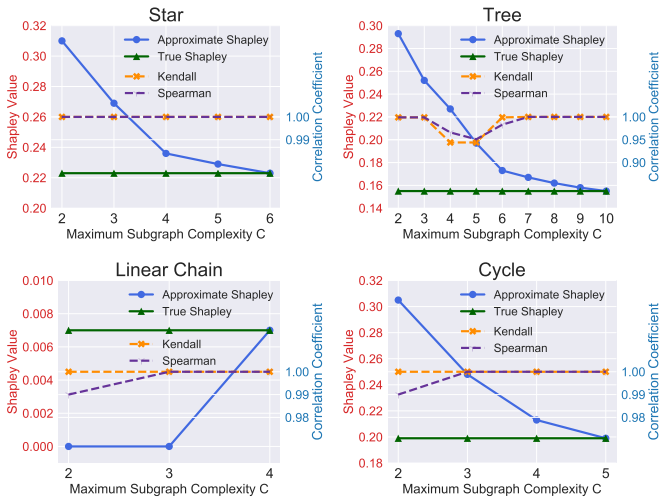

Figure 7: The influence of subgraph complexity on the approximation of Shapley values of the explaining variables A in the four small MRFs in Figure 3. Also plotted are the correlation coefficients of the two rankings of explaining variables ordered by the approximated and exact Shapley values. The correlation coefficients are almost 1, indicating maximal agreement between the two rankings.

THEOREM 5.5. (**No Submodularity**) *There exists an MRF $G = (\mathcal{V}, \mathcal{E})$, a target variable $X \in \mathcal{V}$, and variables $E, F \in \mathcal{V}$, such that* $\mathrm{SV}(\{EF\}; X, G) - \mathrm{SV}(F; X, G) \geq \mathrm{SV}(E; X, G) - \mathrm{SV}(\emptyset; X, G)$.

An example of the lack of submodularity is given in the right-most subfigure of Figure 6, where we found $\mathrm{SV}(\{EF\}; X, G) - \mathrm{SV}(F; X, G) \geq \mathrm{SV}(E; X, G) - \mathrm{SV}(\emptyset; X, G)$. The lack of submodularity indicates that it is computationally difficult to find a subgraph G'_s so that $\mathrm{SV}(G'_s; X, G)$ well-approximates the maximal $\mathrm{SV}(G_s; X, G)$ [25].

6 EXPERIMENTS

We evaluate GraphShapley in terms of approximation quality to the *exact* Shapley value and the accuracy in finding important contributors to BP inference outcomes.

6.1 Evaluation on synthetic datasets

6.1.1 Datasets and MRF setups. We evaluate GraphShapley on the small synthetic MRFs in Figure 3. These MRFs are providing "ground truth" of the variable importance, in the sense that the true Shapley values as defined in Eq. (6) can be evaluated exactly.

6.1.2 Results. In Figure 7, we compare the approximated and exact Shapley values with different subgraph sizes (C). The approximated values can be quite different from the exact values, and whether they agree with each other depends on the diameter of the MRF (the tree needs a larger C and the linear chain needs the smallest C).

The above data seem worrisome. We further compute Kendall-τ and Spearman correlation coefficients of the two rankings of explaining variables ordered by the approximated and the exact Shapley values. From the same figure, we can see that the correlation coefficients are mostly close to 1, confirming that the two rankings

Table 2: *Left*: Statistic of the networks. *Right*: Overall symmetric KL performances (∘ indicates the runner-up methods and • indicates the best method certified by statistically significant t-tests). LIME does not apply to the last three networks since there are no node features.

Datasets	Network Statistics				Performances						
	Classes	Nodes	Edges	edge/node	Random	Embedding	PageRank	Sensitivity	LIME	MC-sampling	GraphShapley
Cora	7	2,708	10,556	3.90	0.831	0.344	0.891	0.729	1.401	0.173 ± 0.03 ∘	**0.104** •
Citeseer	6	3,321	9,196	2.78	0.495	0.301	0.589	0.512	0.921	0.179 ± 0.03 ∘	**0.124** •
PubMed	3	19,717	44,324	2.25	1.043	0.706	1.118	0.941	1.519	0.431 ± 0.14 ∘	**0.092** •
YelpChi	2	105,659	269,580	2.55	0.296	0.058	0.035	0.011 ∘	0.691	0.038 ± 0.01	**0.001** •
YelpNYC	2	520,200	1,436,208	2.76	0.297	0.058	0.043	0.018 ∘	0.692	0.042 ± 0.01	**0.001** •
YelpZip	2	873,919	2,434,392	2.79	0.204	0.084	0.031	0.012 ∘	0.693	0.027 ± 0.01	**0.001** •
Blogcatalog	39	10,312	333,983	32.39	6.673	6.285	3.903	5.944	-	3.323 ± 1.17 ∘	**2.212** •
Flickr	195	80,513	5,899,882	73.28	3.695	3.082	2.789	2.650 ∘	-	2.833 ± 0.31	**1.432** •
Youtube	47	31,703	96,361	3.04	0.077	0.061	0.074	0.070	-	0.044 ± 0.01 ∘	**0.031** •

are mostly the same and the approximated values are still useful for identifying salient explaining variables.

6.2 Evaluation on real-world datasets

6.2.1 Datasets and MRF setups. We drew real-word datasets from three applications of MRF. The statistics of the datasets are shown in Table 2. First, in collective classification, the goal is to classify a paper in a citation network into one of the many research areas. We construct an MRF for each of the three citation networks (Citeseer, Cora, PubMed), with the research area of a paper as a random variable (node) X_i. An undirected edge (X_i, X_j) is added if X_i cites paper X_j. We assume a homophily relationship over the edges, so that two papers are likely to be in the same area if they are connected [24]. The compatibility matrix $\psi(X_i, X_j)$ has value 0.9 on the diagonal, and has $\frac{0.1}{c-1}$ elsewhere (c is the number of classes). We assign probability 0.9 to the true class and $\frac{0.1}{c-1}$ to the other classes to the prior $\phi(X)$ of each labeled node X (80% of the total nodes). The uniform distribution is used as the priors of the unlabeled nodes (20% of the total nodes). The explanations of the inferred beliefs are computed on the unlabeled nodes. Second, we adopt the Yelp review networks for spam detection. We represent reviewers, reviews, and products and their relationships by an MRF and set node priors and compatibility matrix following the state-of-the-art MRF spam detector [31]. Lastly, we represent users as nodes and behaviors including subscription and tagging as edges on three networks (Blogcatalog, Flickr and Youtube) [41]. The MRF setups are the same as for the citation networks and BP infers the preferences of users.

6.2.2 Baselines.

Random generates an importance score of each explaining variable for each target variable randomly, ignoring the BP inference results.

Embedding uses DeepWalk [28] to embed the variables on as MRF graph, and calculates the importance of an explaining variable to a target node based on the similarity between the two nodes.

PageRank [26] is a global ranking of the importance of the variables, regardless of the target node to be explained. It also fails to consider the inference outcomes made by BP.

Sensitivity Analysis [4, 40] is a family of gradient-based approaches that measure how the output changes w.r.t. input changes. The most related work is [4], which derived a closed-form solution for the sensitivity analysis for graphical models. We approximate the gradient of a target belief with respect to the prior of an explaining variable by comparing the beliefs before and after setting the priors to the uniform distribution. Sensitivity analysis relies on gradients and thus can't consider the topological changes which are not continuous. GraphShapley handles the discontinuity by subgraph enumeration.

LIME [32] uses node features to fit a logistic regression model for each target node to predict the beliefs of the target node (see [5] for more details).

MC-sampling [3, 38] uses Monte Carlo simulation to approximate the Shapley values defined in Eq. (6). The calculation is similar to GraphShapley, except that the DFS subgraph enumeration is replaced by a random sampling of spanning trees rooted at a target node.

We do not compare GraphShapley with GraphEXP [5], since the form of explanation is different (a ranking of explaining variables vs. a subgraph for each target). GraphShapley is not comparable to [14, 46] either, as they focused on explaining graph neural networks rather than MRF with BP inference.

6.2.3 Quantitative Results. Unlike classification tasks, due to the lack of ground truth variable contributions, explanations are hard to evaluate quantitatively. Indeed, there exist multiple explanations of the same phenomenon that are valid along different dimensions [33]. We demonstrate that GraphShapley can identify explaining variables that are influential on the target node beliefs. For each target node, after obtaining the Shapley values of the explaining nodes (can be labeled or unlabeled), the priors of all nodes on the MRF are set to uniform and the priors of the top important explaining variables up to 25% are reinstated. Then the belief of the target node $\tilde{b}(X)$ on the reinstated MRF is calculated, whose quality is measured by the symmetric KL-divergence $\mathrm{KL}(b(X)||\tilde{b}(X)) + \mathrm{KL}(\tilde{b}(X)||b(X))$. If the ranking makes sense, then the top-ranked explaining nodes should well approximate the original MRF, as reflected by the symmetric KL-divergence. All baselines except LIME can rank the explaining variables and can be evaluated similarly. LIME is trained to approximate $b(X)$ and we report the symmetric KL-divergence between $b(X)$ and the belief approximated by LIME. The mean symmetric KL-divergences are shown in Table 2, with t-tests conducted between GraphShapley and the runner-ups.

We can conclude that 1) GraphShapley performs best overall, due to the consideration of probabilistic and topological contributions; 2) MC-sampling is frequently the runner-up in the means but with higher variance due to random sampling. 3) LIME has the worst

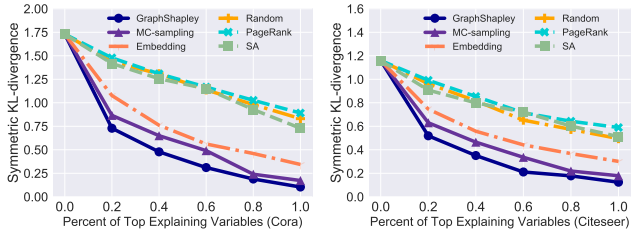

Figure 8: Symmetric KL-divergence (the smaller the better) on Cora (*left*) and Citeseer (*right*) as more and more explaining variables' priors are reinstated.

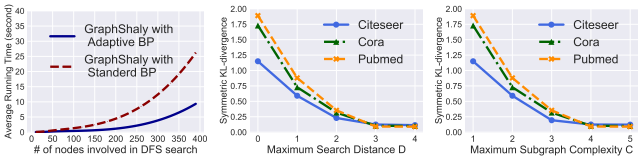

Figure 9: *Left*: Speed-up by Adaptive BP when DFS reaches different numbers of explaining nodes on Cora. *Center*: Parameters sensitivity of maximum search distance D. *Right*: Parameters sensitivity of maximum subgraph complexity C.

performance, as it is not designed for graphs and cannot take into account the many pieces of information on graphical models and the BP inference outcomes.

We also reinstate one variable's prior at a time in the ranking order. Figure 8 shows that as more and more priors are reinstated, the symmetric KL-divergences of all methods go down, with GraphShapley having the sharpest decrease. This indicates that GraphShapley ranks the more salient explaining variables before the less relevant ones.

Speed-up by Adaptive BP. In Figure 9 (left panel), we compare the average running time without and with adaptive BP as the number of explaining nodes visited by the DFS search varies on the Cora dataset. It can be clearly seen that the running time increases exponentially without Adaptive BP and only near-linearly with Adaptive BP.

Parameters Sensitivity. On the Citeseer, Cora, and PubMed datasets, Figure 9 (center and right) shows the symmetric KL-divergences against different D (the maximum distance of the explaining variables from the target variable) and C (the maximum complexity of the enumerated subgraph) values, respectively. The two parameters can affect the quality of the Shapley values, when these two parameters are too small (≤ 2). Larger D and C values seem beneficial but can increase the subgraph search time complexity. Fortunately, when these parameters are ≥ 3, the approximation quality is not too sensitive to these parameters.

6.2.4 Qualitative Results. In Figure 10, we extract two subgraphs from the MRF for the Citeseer network and show the Shapley values on the nodes. For the details, see the caption. The general conclusion is that GraphShapley can capture both the probabilistic contributions due to variable prior distribution and the topological contributions due to connectivities. The left panel also demonstrates the **Equal contribution** property (Theorem 5.3).

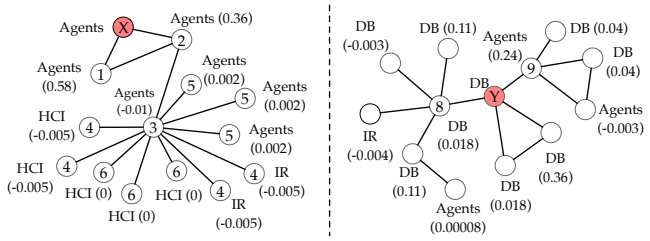

Figure 10: Blank nodes are explaining variables and the red solid nodes (X and Y) are target nodes. The ground truth labels and the computed Shapley values are attached to each explaining node. The ground truth classes of the target nodes are also annotated. *Left*: node 1 and node 2 have the same labels as node X and are the direct neighbor of X, thus they have positive higher Shapley values. Node 3 has the same label but negative Shapley value, since many of its connected neighbors are of classes "*IR*" and "*HCI*." Four nodes labeled as 4 have equal Shapley values, and the two nodes labeled as 5 have equal Shapley values, showcasing the Equal Contribution proof. Nodes 4 and 6 have different contributions due to their different priors (Nodes 6 has a uniform prior and makes no contribution). *Right*: node 9 has a positive contribution to the target node, though being from a different class. The reason is that the node serves as a bridge transporting the "*DB*" probabilities from distance to the target node.

7 RELATED WORK

Interpretability and Explanation of Models. [32] explains the predictions of a classifier by approximating it locally with an interpretable model. [46] explains arbitrary graph neural networks using explainer models but do not consider PGMs. [47] uses graph neural networks to learn the message-passing process in belief propagation while explanations for target nodes are not provided. The more relevant methods are proposed in [4, 5, 7], which use gradients or greedy subgraph search to find salient variables or subgraphs to explain the inference. The prominent feature of GraphShapley is that it can quantify both probabilistic and topological contributions.

Shapley Values as explanations. Shapley values have been applied to the interpretability of machine learning models. In [35], they provide a prediction explanation framework based on Shapley values which encompasses LIME as a special case. Efficient calculation methods of Shapley value have been studied, such as in [17, 21]. Two algorithms with linear complexity for feature importance scoring are developed in [6]. In [11] and [2], they approximate Shapley values for deep networks via sampling.

Explanation Explainability and Robustness. Explanations generated by an algorithm can be as difficult to understand as machine learning predictions. Explaining explanations [12, 23] has been studied for other models but not for Shapley value explanations. The robustness of explanations is studied in [1, 10, 30, 44, 48], but none of them is for Shapley values on MRF.

Subgraph Enumeration. Subgraph enumeration algorithms have been researched for multiple decades [43]. The most relevant one is proposed in [36] where they essentially enumerate all subset of vertices that constitute a connected subgraph for computing Shapley values for a graph-restricted game rather than for explaining BP. However, in an MRF, the same subset of nodes can be connected in different ways and our enumeration algorithm can address the enumeration of different ways of connection.

8 CONCLUSION

We propose GraphShapley to provide a novel form of explanations for graphical model inference. The probabilistic and topological contributions of explaining variables can be measured by GraphShapley. Theoretically, we prove theorems to characterize the Shapley values defined for BP inference on MRFs. Explanation of the Shapley explanations is provided to make the Shapley values more accessible to human users. In terms of explanation faithfulness and speed, we empirically show the superior performance of the GraphShapley over other baselines, such as the gradient-based explanations.

Acknowledgement

Yifei Liu, Yazheng Liu, and Xi Zhang was supported by the National Key Research and Development Program of China (No.2017YFB0803301) and Natural Science Foundation of China (No.61976026, No.U1836215) and 111 Project (B18008). Chao Chen and Sihong Xie are supported by Lehigh young faculty startup and NSF grants CNS-1931042 and IIS-2008155.

REFERENCES

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2018. Sanity Checks for Saliency Maps. In *NeurIPS*.

[2] Marco Ancona, Cengiz Oztireli, and Markus Gross. 2019. Explaining Deep Neural Networks with a Polynomial Time Algorithm for Shapley Values Approximation. In *ICML*.

[3] Javier Castro, Daniel Gómez, and Juan Tejada. 2009. Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research* 36, 5 (2009), 1726–1730.

[4] Hei Chan and Adnan Darwiche. 2005. Sensitivity analysis in Markov networks. In *IJCAI*.

[5] Chao Chen, Yifei Liu, Xi Zhang, and Sihong Xie. 2019. Scalable Explanation of Inferences on Large Graphs. In *ICDM*.

[6] Jianbo Chen, Le Song, Martin J. Wainwrightand, and Michael I. Jordan. 2019. L-shapley and c-shapley: Efficient model interpretation for structured data. In *ICLR*.

[7] Adnan Darwiche. 2003. A differential approach to inference in Bayesian networks. *Journal of the ACM (JACM)* (2003), 280–305.

[8] Mengnan Du, Ninghao Liu, and Xia Hu. 2019. Techniques for Interpretable Machine Learning. *Commun. ACM* 63, 1 (2019), 68–77.

[9] Papachristoudis Georgios and Fisher III John. 2015. Adaptive Belief Propagation. In *ICML*.

[10] Amirata Ghorbani, Abubakar Abid, and James Y Zou. 2017. Interpretation of Neural Networks is Fragile. In *AAAI*.

[11] Amirata Ghorbani and James Zou. 2019. Data Shapley: Equitable Valuation of Data for Machine Learning. In *ICML*.

[12] L H Gilpin, D Bau, B Z Yuan, A Bajwa, M Specter, and L Kagal. 2018. Explaining Explanations: An Overview of Interpretability of Machine Learning. In *DSAA*. 80–89.

[13] Riccardo Guidotti, Anna Monreale, Franco Turini, Dino Pedreschi, and Fosca Giannotti. 2018. A Survey of Methods for Explaining Black Box Models. *ACM Comput. Surv.* 51 (2018), 93:1–93:42.

[14] Qiang Huang, Makoto Yamada, Yuan Tian, Dinesh Singh, Dawei Yin, and Yi Chang. 2020. GraphLIME: Local Interpretable Model Explanations for Graph Neural Networks. (2020).

[15] Sarthak Jain and Byron C Wallace. 2019. {A}ttention is not {E}xplanation. In *NAACL*.

[16] K. Jha, Y. Wang, G. Xun, and A. Zhang. 2018. Interpretable Word Embeddings for Medical Domain. In *ICDM*.

[17] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gurel, Bo Li, Ce Zhang, Dawn Song, and Costas Spanos. 2019. Towards Efficient Data Valuation Based on the Shapley Value. In *AISTATS*. 1167–1176.

[18] Murphy Kevin. 2001 (accessed April 25, 2020). *List of Bayesian Network Software*. https://www.cs.ubc.ca/~murphyk/Bayes/old.bnsoft.html

[19] Daphne Koller and Nir Friedman. 2009. *Probabilistic graphical model: principles and techniques*. MIT Press.

[20] Andrew McCallum, Dayne Freitag, and Fernando C. N. Pereira. 2000. Maximum Entropy Markov Models for Information Extraction and Segmentation. In *ICML*.

[21] Tomasz P. Michalak, Karthik .V. Aadithya, Piotr L. Szczepański, Balaraman Ravindran, and Nicholas R. Jennings. 2013. Effcient Computation of the Shapley Value for Game-Theoretic Network Centrality. *JAIR* 46 (2013), 607–650.

[22] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019), 1–38.

[23] Brent Mittelstadt, Chris Russell, and Sandra Wachter. 2019. Explaining Explanations in AI. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*.

[24] Galileo Mark Namata, Ben London, and Lise Getoor. 2016. Collective graph identification. *ACM TKDD* 10, 3 (2016), 25.

[25] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming* 14, 1 (Dec 1978), 265–294.

[26] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank citation ranking: Bringing order to the web*. Technical Report. Stanford InfoLab.

[27] Shashank Pandit, Duen Horng Chau, Samuel Wang, and Christos Faloutsos. 2007. Netprobe: A Fast and Scalable System for Fraud Detection in Online Auction Networks. In *WWW*.

[28] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *SIGKDD*.

[29] Forough Poursabzi-Sangdeh, Dan Goldstein, Jake Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. 2018. Manipulating and Measuring Model Interpretability.

[30] Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary Chase Lipton. 2019. Learning to Deceive with Attention-Based Explanations. *ArXiv* abs/1909.0 (2019).

[31] Shebuti Rayana and Leman Akoglu. 2015. Collective opinion spam detection: Bridging review networks and metadata. In *SIGKDD*.

[32] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *SIGKDD*.

[33] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. 2017. Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations. In *IJCAI*.

[34] L Shapley. 1953. A Value for n-Person Games. *Contributions to the Theory of Games* (1953), 31–40.

[35] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features Through Propagating Activation Differences. In *ICML*.

[36] Oskar Skibski, Talal Rahwan, Tomasz P. Michalak, and Michael Wooldridge. 2019. Enumerating Connected Subgraphs and Computing the Myerson and Shapley Values in Graph-Restricted Games. *ACM TIST* 10, 2 (2019), 15.

[37] Erik Strumbelj and Igor Kononenko. 2010. An Efficient Explanation of Individual Classifications using Game Theory. *JMLR* (2010).

[38] Erik Štrumbelj and Igor Kononenko. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems* 41, 3 (2014), 647–665.

[39] Henri Jacques Suermondt. 1992. *Explanation in Bayesian Belief Networks*. Ph.D. Dissertation.

[40] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. In *ICML*.

[41] Lei Tang and Huan Liu. 2009. Relational learning via latent social dimensions. In *SIGKDD*.

[42] Jeroen Van Bouwel and Erik Weber. 2002. Remote Causes, Bad Explanations? *Journal for the Theory of Social Behaviour* 32, 4 (2002), 437–449.

[43] Xifeng Yan and Jiawei Han. 2002. gspan: Graph-based substructure pattern mining. In *ICDM*.

[44] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David W Inouye, and Pradeep D Ravikumar. 2019. On the (In)fidelity and Sensitivity of Explanations. In *NeurIPS*.

[45] Ming Yin, Jennifer Wortman Vaughan, and Hanna Wallach. 2019. Understanding the Effect of Accuracy on Trust in Machine Learning Models *(CHI)*.

[46] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. 2019. GNN Explainer: A Tool for Post-hoc Explanation of Graph Neural Networks. In *NeurIPS*.

[47] KiJung Yoon, Renjie Liao, Yuwen Xiong, Lisa Zhang, Ethan Fetaya, Raquel Urtasun, Richard Zemel, and Xaq Pitkow. 2018. Inference in probabilistic graphical models by graph neural networks. In *ICLR workshop*.

[48] Xinyang Zhang, Ningfei Wang, Shouling Ji, Hua Shen, and Ting Wang. 2018. Interpretable Deep Learning under Fire. *ArXiv* abs/1812.0 (2018).