Deep Learning based Scalable Inference of Uncertain Opinions

Xujiang Zhao, Feng Chen
Computer Science Department
University at Albany – SUNY, Albany, NY, USA
xzhao8@albany.edu, fchen5@albany.edu

Jin-Hee Cho *
Department of Computer Science
Virginia Tech, Falls Church, VA, USA
jicho@vt.edu

***Abstract*—Subjective Logic (SL) is one of well-known belief models that can explicitly deal with uncertain opinions and infer unknown opinions based on a rich set of operators of fusing multiple opinions. Due to high simplicity and applicability, SL has been popularly applied in a variety of decision making in the area of cybersecurity, opinion models, and/or trust or social network analysis. However, SL has an issue of scalability to deal with a large-scale network data. In addition, SL has shown a bounded prediction accuracy due to its inherent parametric nature by treating heterogeneous data and network structure homogeneously based on the assumption of a Bayesian network. In this work, we take one step further to deal with uncertain opinions for unknown opinion inference. We propose a deep learning (DL)-based opinion inference model while node-level opinions are still formalized based on SL. The proposed DL-based opinion inference model handles node-level opinions explicitly in a large-scale network using *graph convoluational network* (GCN) and *variational autoencoder* (VAE) techniques. We adopted the GCN and VAE due to their powerful learning capabilities in dealing with a large-scale network data without parametric fusion operators and/or Bayesian network assumption. This work is the first that leverages the merits of both DL (i.e., GCN and VAE) and a belief model (i.e., SL) where each node level opinion is modeled by the formalism of SL while GCN and VAE are used to achieve non-parametric learning with low complexity. By mapping the node-level opinions modeled by the GCN to their equivalent Beta PDFs (probability density functions), we develop a network-driven VAE to maximize prediction accuracy of unknown opinions while significantly reducing algorithmic complexity. We validate our proposed DL-based algorithm using real-world datasets via extensive simulation experiments for comparative performance analysis.**

I. Introduction

In the decision making domain, including the fields of evidence and belief theories, reasoning or managing uncertainty has been studied since 1960s. The examples include Fuzzy Logic, DST, Transferable Belief Model (TBM), and Dezert-Smarandache Theory (DSmT). Most of them deal with uncertainty implicitly [6]. In 1990's, as a variant of DST, Subjective Logic (SL) [14] is proposed to explicitly deal with the dimension of uncertainty in subjective opinions. SL defines a binomial opinion (e.g., pro vs. con or agree vs. disagree) with three dimensions, including belief, disbelief, and uncertainty. SL provides a set of various operators to fuse multiple, different opinions that allow deriving structural relations between opinions (i.e., random variables) in a network. Although SL has offered a rich set of operators that allow us to fuse multiple opinions, its inherent parametric way of combining opinions has been shown as a hurdle to limit its scalability and led to a bounded prediction accuracy in deriving unknown opinions. To handle these issues, the variants of SL have been proposed to resolve the issue of scalability in SL, such as subjective networks based on Bayesian networks [13] and collective subjective logic based on Markov Random Fields (MRFs) [6]. However, due to the inherent parametric opinion derivation (e.g., fusion operators) and the distribution assumption (e.g., Bayesian networks), we have observed the bounded performance of SL and its variants [6, 13].

In this paper, we propose a deep learning (DL)-based opinion inference model that addresses the key challenges of high scalability and handling heterogeneous opinions (i.e., node-level opinions) in large-scale network data. We adopt two state-of-the-art techniques called the *graphical convolutional network* (GCN) and *variational autoencoder* (VAE). In this work, a node-level opinion is formulated as SL-based binomial opinion, consisting of belief, disbelief, and uncertainty masses where the sum of three values is 1. Based on this node-level opinion formulation, we develop a GCN to directly consider heterogeneous structural dependencies between node-level opinions which include beliefs and uncertainties in a given network data. And then, we map their combined forms (opinions) to their equivalent Beta PDFs and develop a network-driven VAE that provides powerful learning capability to achieve highly accurate opinion predictions with low complexity. We summarize the **key contributions** made in this work:

1) This work is the first that combines non-parametric deep learning (DL) algorithm with an opinion formalism of SL in order to achieve powerful learning without being tied to parametric fusion operators in SL while keeping the explicit consideration of the uncertainty dimension in a subjective opinion. In particular, this work does not assume that observations (i.e., evidence) are available to derive an SL-based opinion based on the mapping rule with Beta distribution; instead, given a set of known 'subjective opinions' in SL, which is a second order

* This work is done when Jin-Hee Cho was with US Army Research Laboratory.

978-1-5386-9159-5/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDM.2018.00096

probability, unknown opinions can be inferred by using the proposed DL-based inference method.

2) This work is the first that proposes a DL-based opinion inference algorithm dealing with uncertain opinions characterized by a set of heterogeneous belief and uncertainty in a large-scale network data. In particular, we adopted the DL techniques called the GCN and VAE which provide a powerful capability to obtain non-parametric learning with low complexity and high prediction accuracy. To be specific, the GCN can model heterogeneous structural dependencies among node-level beliefs and uncertainties in a given network data while the proposed network-driven VAE can model the inherent dependencies between beliefs and uncertainties based on the mapping of their SL-based opinions to equivalent PDFs for high opinion prediction with low complexity.
3) The proposed DL-based opinion inference algorithm is validated through extensive experiments using real-world datasets. We conducted comparative performance analysis by comparing the performance of the proposed DL-based algorithms with those of the original SL and the state-of-the-art counterpart (i.e., collective subjective logic, or CSL [6]). The implementation of our proposed methods and the testeddata sets are available at github[1].

The rest of this paper is organized as follows. Section II provides the overview of related work in the area of probabilistic models, belief models, and DL-based inference models. Section III gives the overview of SL and GCN as the basis of the proposed DL-based opinion inference model. Section IV shows an example scenario and addresses a problem statement. Section V describes the proposed DL-based opinion inference model. Section VI demonstrates the experimental results and discusses their overall trends and interpretation. Finally, Section VII concludes this work and suggests the future work directions.

II. Related Work

A. Probabilistic Models

Extensive efforts have been made to model uncertainty caused by *a lack of information* or *knowledge* in network data as a joint probability distribution over a set of variables, in which each variable relates a node in the network. Two typical probabilistic models include MRFs [4] and Gaussian Processes (GPs) [18]. The former models the joint distribution based on potential functions of the cliques to capture the relational structure. The latter models the joint distribution using a multivariate Gaussian distribution and uses the covariance matrix to characterize the structural relations between the variables in the network.

Probabilistic models have shown limited capabilities in considering uncertainty caused by ignorance (i.e., a lack of evidence about the truth of states) and other causes, such as vagueness (i.e., failing in discerning a single state) and ambiguity (i.e., failing in observing consensus due to conflicting evidence). For example, if somebody wants to express ignorance about the state x as "*I don't know*," this would be impossible with a simple probability value. A probability $P(x) = 0.5$ would mean that x and $\bar{x}$ are equally likely, which is quite informative in deed, unlike ignorance.

B. Belief Models

Belief models were designed to manage uncertainty introduced by various root causes, such as ignorance, vagueness, and ambiguity, during the process of decision making. Well-known belief models include Fuzzy Logic, DST, TBM, SL, and DSmT. However, they have not explicitly addressed the dimension of uncertainty in subjective opinions [6].

SL has been proposed to define an opinion that explicitly deals with uncertainty. In addition, SL offers a variety of operators to fuse multiple opinions [14]. New extensions of SL have been proposed to make SL scalable to large-scale networks, such as subjective Bayesian networks [13] and collective subjective logic, as a hybrid approach, by combining SL, probabilistic soft logic, and MRFs [4, 6]. However, all the preceding belief models are designed based on predefined operators or distribution assumptions (e.g., Bayesian networks) that may not effectively deal with heterogeneous uncertain opinions in a given network data. In this work, we proposed a DL-based opinion inference model based on GCN and VAE which provides high prediction accuracy and low complexity for opinion inference in a large-scale network data while heterogeneous node-level opinions are formulated based SL.

C. DL-based Inference Models

In early days of machine learning (or deep learning), recursive neural networks (RNNs) are used to deal with data representations in directed acyclic graphs [8]; later, Graph Neural Networks (GNNs) [9] are developed as a generalization of RNNs to process general directed and undirected graphs. After then, convoluational neural networks (CNNs) is developed to deal with data representations from a spatial domain to a graph domain, which has received significant attention. The methods developed in this direction are called graph convoluational networks (GCNs) and fall into two main categories: spectral approaches and non-spectral approaches. GCNs have demonstrated the state-of-the-art performance in a number of challenging mining tasks (e.g., semi-supervised node classification and link prediction) [10, 15].

Spectral approaches for GCNs explore convolutions based on a spectral representation of the graphs. Bruna et al. [5] implemented the convolution operator as a spectral filter in the Fourier domain by calculating the eigen-decomposition of the graph Laplacian, which however is computationally expensive and leads to non-spatially localized filters. Henaff et al. [12] proposed a parameterization of the spectral filters to make them spatially localized. Defferrard et al. [7] and Kipf and Welling [15] introduced approximations of the filters based on a Chebyshev expansion of the Graph Laplacian that do not need eigen-decomposition of the graph Laplacian, scale

[1]https://github.com/zxj32/GCN-VAE-opinion

linearly in the number of graph edges, and are concurrently spatially localized. *Non-spectral approaches* for GCNs conduct convoluations directly on neighborhoods (i.e., groups of spatially close neighbors) in the graph. As a graph is irregular in general, the receptive fields required for convolutions are irregular as well. Different strategies have been proposed to deal with irregular receptive fields [10, 16, 22].

III. Background

For this present work to be self-contained, here we provide the overview of SL and GCN as the basis of the proposed DL-based opinion inference model.

A. Subjective Logic (SL)

In SL, a binomial opinion is defined in terms of belief, disbelief, and uncertainty towards a given proposition x. For simplicity, we omit x in the following notations [14]. To formally put, an opinion w is represented by:

$$w = (b, d, u, a) \tag{1}$$

where b is belief (e.g., true), d is disbelief (e.g., false), and u is uncertainty (i.e., ignorance or a lack of evidence). a represents a base rate, a prior knowledge upon no commitment (e.g., neither true nor false), where $b + d + u = 1$ for $(b, d, u, a) \in [0, 1]^4$. We denote an opinion by w, which can be *projected* onto a single probability distribution by removing the uncertainty mass.

A binomial opinion follows a Beta pdf (probability density function), denoted by Beta$(p|\alpha, \beta)$, where α represents the amount of positive evidence and β is the amount of negative evidence [14]. In SL, uncertainty u decreases as more evidence, α and β, is received over time. An opinion w can be obtained based on α and β as $w = (\alpha, \beta)$. This can be translated to $w = (b, d, u, a)$ using the mapping rule in SL.

SL offers an operator, $\otimes$, to discount trust when an entity does not have any direct experience with another entity. That is, transitive trust based on structural relations is used to derive trust between two entities who have not interacted before. Trust from i to j, denoted by $w_j^i = (b_j^i, d_j^i, u_j^i, a_j^i)$, and trust from j to k, $w_k^j = (b_k^j, d_k^j, u_k^j, a_k^j)$, are used to derive trust from i to k, $w_k^i := (b_k^i, d_k^i, u_k^i, a_k^i) = w_j^i \otimes w_k^j$. It is obtained by:

$$\begin{gathered} b_k^i = b_j^i \otimes b_k^j = b_j^i b_k^j,\ d_k^i = d_j^i \otimes d_k^j = b_j^i d_k^j \\ u_k^i = u_j^i \otimes u_k^j = d_j^i + u_j^i + b_j^i u_k^j,\ a_k^i = a_j^i \otimes a_k^j = a_k^j. \end{gathered} \tag{2}$$

SL also provides a consensus operator, $\oplus$, to find a consensus between two opinions [14] where two entities observe a same entity. An opinion after i exchanges opinions with k is given by $w_k^i \oplus w_k^j$, where:

$$\begin{gathered} b_k^i \oplus b_k^j = \frac{b_k^i u_k^j + b_k^j u_k^i}{\zeta},\ d_k^i \oplus d_k^j = \frac{d_k^i u_k^j + d_k^j u_k^i}{\zeta} \\ u_k^i \oplus u_k^j = \frac{u_k^i u_k^j}{\zeta},\ a_k^i \oplus a_k^j = a_k^i. \end{gathered} \tag{3}$$

where $\zeta = u_j^i + u_k^j - u_j^i u_k^j > 0$. When $\zeta = 0$, $w_k^i \oplus w_k^j$ is defined by:

$$\begin{gathered} b_k^i \oplus b_k^j = \frac{\psi b_k^i + b_k^j}{\psi + 1},\ d_k^i \oplus d_k^j = \frac{\psi d_k^i + d_k^j}{\psi + 1} \\ u_k^i \oplus u_k^j = 0,\ a_k^i \oplus a_k^j = a_k. \end{gathered} \tag{4}$$

where $\psi = \lim(u_k^i / u_k^j)$. These discounting, $\otimes$, and consensus, $\oplus$, operators [14] are used to derive trust measures based on the trust opinions of relationships. Due to space constraint, we don't show an example of using these operators. Interested readers can be referred to [6].

In this work, we aim to derive a set of unknown opinions $\mathbf{x} = \{x_1, \cdots, x_n\}$ when a set of observed opinions $\mathbf{y} = \{y_1, \cdots, y_m\}$ is given where both opinions are represented by a binomial opinion with four dimensions, as described in Eq. (1) (i.e., w_{x_i} for $i = 1 \cdots n$ and w_{y_j} for $j = 1 \cdots m$).

B. Graph Convolutional Networks (GCN)

This section introduces a state-of-the-art GCN model [15] used in this work. Denote a graph by $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} = \{1, \cdots, n\}$ refers to the set of nodes and $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ refers to the set of edges. Let $\mathbf{A} \in \{0, 1\}^{n \times n}$ be an adjacency matrix, where $A_{i,j} = 1$ if $(i, j) \in \mathbb{E}$ and, otherwise, $A_{i,j} = 0$. The (unnormalized) graph Laplacian matrix is an $n \times n$ symmetric positive-semidefinite matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D}$ is the degree matrix and $D_{i,i}$ refers to the degree of node i and $D_{i,i} = 0$ for $i \neq j$.

The graph Laplacian has an eigen decomposition $\mathbf{L} = \Phi \Lambda \Phi^T$, where $\Phi = (\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n)$ are the orthonormal eigenvectors and $\Lambda = diag(\lambda_1, \cdots, \lambda_n)$ is the diagonal matrix of corresponding eigenvalues. The eigenvalues serve as the role of frequencies in classical harmonic analysis and the eigenvectors are interpreted as Fourier atoms. Given a signal $\mathbf{r} \in \mathbb{R}^n$ (or a vector of feature values) on the nodes of graph $\mathbb{G}$, where r_i refers to a feature value at node i, its graph Fourier transform is given by $\hat{\mathbf{r}} = \Phi^T \mathbf{r}$. Given two signals $\mathbf{r}$ and $\mathbf{b}$ on the graph, we can define their spectral convolution as the element-wise product of their Fourier transformations,

$$\mathbf{r} \star \mathbf{b} = \Phi^T (\Phi^T \mathbf{r}) \circ (\Phi^T \mathbf{b}) = \Phi diag(\hat{r}_1, \cdots, \hat{r}_n) \hat{\mathbf{b}}, \tag{5}$$

which is a property of the well-known *Convolutional Theorem* in the Euclidean case.

As a graph is irregular with nodes having widely different degrees, it is difficult to directly define a convolution on the nodes. Instead, Bruna et al. [5] used the spectral definition of convolution (see Eq. (5)) to generalize Convolutional Neural Networks (CNNs) on graphs, which has a spectral convolutional layer of the form as:

$$g_\theta \star \mathbf{r} = \Phi g_\theta \Phi^T \mathbf{r}. \tag{6}$$

The filter g_θ can be defined as a function of the eigenvalues of L, i.e., $g(\Lambda)$. Evaluation of Eq. (6) is computationally expensive because multiplication with the eigenvector matrix Φ is $O(n^2)$, in addition to the high computational cost in computing the eigendecomposition of L in the first place. To address this problem, Hammond et al. [11] suggest that $g_\theta(\Lambda)$ can be well-approximated by a truncated expansion in terms of Chebyshev polynomials $T_k(r)$ up to K-th order:

$$g_\theta(\Lambda) \approx \sum_{k=1}^{K} \theta_k T_k(\tilde{\Lambda}), \tag{7}$$

with a rescaled $\tilde{\Lambda} = 2\Lambda/\lambda_{max} - I_n$. λ_{max} refers to the largest eigenvalue of L. $\theta \in \mathbb{R}^K$ is a vector of Chebyshev coefficients. The Chebyshev polynomial can be recursively defined as $T_k(r) = 2xT_{k-1}(r) - T_{k-2}(r)$, with $T_0(r) = 1$ and $T_1(r) = r$. Applying the approximation based on Chebyshev polynomials, a convolution of a signal x with a filter g_θ now has the approximated form:

$$g_\theta \star \mathbf{r} \approx \sum_{k=1}^{K} \theta_k T_k(\tilde{L})\mathbf{r}. \tag{8}$$

By stacking multiple convolutional layers of the form of Eq. (8) in which each layer is followed by a point-wise non-linearity filter, we can therefore design a multi-layer convolutional neural network model based on graph convolutions.

For example, a two-layer GCN model [15] for the task of node classification on a network with a symmetric adjacency matrix $\mathbf{A}$ (binary or weighted) can be formulated as:

$$\mathbf{p} = g(\mathbf{r}, \mathbf{A}) = \text{softmax}\left(g_{\mathbf{W}^{(1)}} \star \text{ReLU}(g_{\mathbf{W}^{(0)}} \star \mathbf{r})\right), \tag{9}$$

where the output matrix $\mathbf{p} \in [0,1]^{n\times 2}$ provides the predicted probabilities of the binary classes of the n nodes. Here, $W^{(0)} \in \mathbb{R}^{1\times H}$ is an input-to-hidden weight vector with H feature maps. $W^{(1)} \in \mathbb{R}^{H\times F}$ is a hidden-to-output weight matrix and F refers to the number of classes. The softmax activation function, defined as $\text{softmax}(r_i) = \frac{1}{Z}\exp r_i$ with $Z = \sum_i \exp(r_i)$, is applied row-wise. A nonlinear activity function called as function *Rectified Linear Unit* (ReLU) is defined as $\text{ReLU}(r_i) = \max\{0, r_i\}$ and is used to introduce element-wise non-linearity.

IV. Problem Formulation

In this section, we describe an example to motivate a problem to solve in this work. We also show how to formulate a given uncertainty-based opinion inference problem.

A. Example Scenario

In this work, we aim to infer unknown opinions, given a set of known opinions, in terms of the applications in traffic congestion prediction in a road network. Given a network, defined as $\mathbb{G} = (\mathbb{V}, \mathbb{E}, y)$, where $\mathbb{V} = \{1, 2, \cdots, N\}$ is the set of vertices (i.e., intersections in the road network), $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ is the set of edges (i.e., road links), and y_i refers to a Boolean variable at node $i \in \mathbb{V}$, in which state 0 indicates 'non-congested' while state 1 refers to 'congested.'

Suppose that we are given the subjective opinions of the congestion variables $\{y_i\}_{i\in\mathbb{L}}$, $\boldsymbol{\omega}_{\mathbb{L}} = [\omega_i]_{i\in\mathbb{L}}$ that are estimated based on their historical observations. A subjective opinion ω_i is defined by a tuple of three components in Eq. (1): $\omega_i = (b_i, d_i, u_i)$. Given these information, we aim to predict the beliefs about the states of the congestion variables at the nodes without sensors (i.e., intersections without any camera), denoted as $\boldsymbol{\omega}_{\mathbb{V}\setminus\mathbb{L}} = [\omega_i]_{i\in\mathbb{V}\setminus\mathbb{L}}$.

B. Problem Statement

We formulate the problem of uncertainty-based inference by:

Problem 1 (Uncertainty-based opinion inference in network data)**:** *Let us define the following notations:*

- *Let $\mathbb{G} = (\mathbb{V}, \mathbb{E}, y)$ be an input network as defined above.*
- *Let $\omega_i = (b_i, d_i, u_i)$ be node i's subjective opinion of variable y_i where node $i \in \mathbb{V}$. Let $\mathbb{L} \subseteq \mathbb{V}$ be a subset of edges whose opinions are denoted by $\boldsymbol{\omega}_{\mathbb{L}} = [\omega_i]_{i\in\mathbb{L}}$.*

Given

- $\mathbb{G} = (\mathbb{V}, \mathbb{E}, y)$*, an input network;*
- $\boldsymbol{\omega}_{\mathbb{L}} = [\omega_i]_{e_i\in\mathbb{L}}$*, a vector of subjective opinions on* $\{y_i\}_{i\in\mathbb{L}}$.

Predict $\boldsymbol{\omega}_{\mathbb{V}\setminus\mathbb{L}} = [\omega_i]_{i\in\mathbb{V}\setminus\mathbb{L}}$*., unknown opinions on* $\{y_i\}_{i\in\mathbb{V}\setminus\mathbb{L}}$

V. DL-based Opinion Inference Model

In this section, we discuss the proposed DL-based opinion inference model and the details on how GCN and VAE are used to infer unknown opinions, given a set of known opinions.

Denote by $\mathbf{B} = [b_i]_{i\in\mathbb{V}}$ the vector of the belief masses on the nodes in $\mathbb{V}$. Denote by $\mathbf{U} = [u_i]_{i\in\mathbb{V}}$ the vector of the uncertainty masses on the nodes in $\mathbb{V}$. As shown in Figs. 1 and 2, our proposed approach consists of two design components: (1) GCN-based modeling for heterogeneous *structural* dependencies among node-level beliefs $\mathbf{B}$ and uncertainties $\mathbf{U}$; and (2) network-driven VAE to model inherent *relational* dependencies between $\mathbf{B}$ and $\mathbf{U}$ based on mapping of opinions to their equivalent PDFs.

A. GCN-based Opinion Model

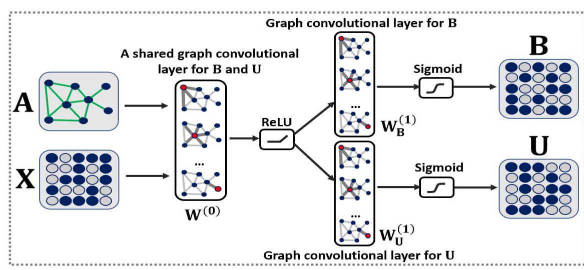


Fig. 1. An overview of our proposed GCN-based opinion model.

As shown in Fig. 1, we propose a two-layer GCN for modeling the node-level beliefs $\mathbf{B}$ and uncertainties $\mathbf{U}$:

$$\begin{aligned}[\mathbf{B}, \mathbf{U}] &= f(\mathbf{X}, \mathbf{A}; \theta) \\ &= \left[\text{sigmoid}\left([g_{\mathbf{W}_{\mathbf{B}}^{(1)}}, g_{\mathbf{W}_{\mathbf{B}}^{(1)}}] \star \text{ReLU}(g_{\mathbf{W}^{(0)}} \star \mathbf{X})\right)\right]\end{aligned} \tag{10}$$

where $\theta = \{\mathbf{W}_{\mathbf{B}}^{(1)}, \mathbf{W}_{\mathbf{U}}^{(1)}, \mathbf{W}^{(0)}\}$, $\mathbf{A}$ is the adjacency matrix of $\mathbb{G}$, and $\mathbf{X} \in \mathbb{R}^{n\times p}$ is a predefined feature matrix.. If the input network does not have node-level features, the feature matrix $\mathbf{X}$ is set to an identity matrix $\mathbf{I}$.

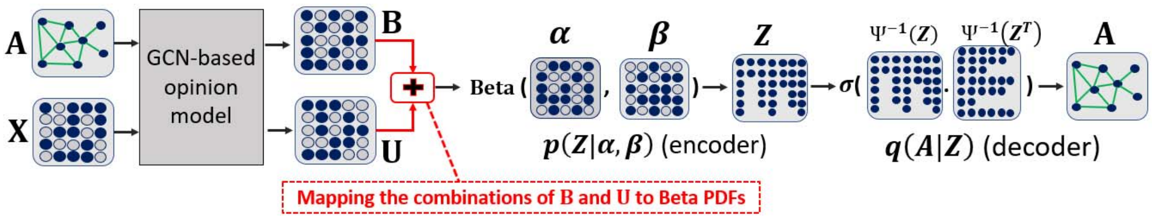


Fig. 2. An overview of our proposed VAE-based opinion model.

Two main design-related questions will be answered here to explain the principles of our proposed GCN model: (1) *how many conovluational layers could be sufficient* and (2) *how to model the heterogeneous dependencies among* $\boldsymbol{B}$ *and* $\boldsymbol{U}$. **For the former question**, this is in general considered as a hyper parameter that requires tuning for different datasets. For GCN, recent studies in several real-world datasets have demonstrated that two convolution layers have less risk of overfitting than more than two covoluation layers for GCN [15]. We hence consider two convolutional layers in our proposed GCN-based opinion model. **For the latter question**, we need to capture two types of dependencies, including *heterogeneous structural dependencies among node-level beliefs* $\boldsymbol{B}$ *and uncertainties* $\boldsymbol{U}$ and *inherent relational dependencies between* $\boldsymbol{B}$ *and* $\boldsymbol{U}$. The first type of dependencies can be well modeled using the two convolution layers in GCN. In particular, the first convolution layer is a shared layer for $\mathbf{B}$ and $\mathbf{U}$ in order to model higher order shared structural information for $\mathbf{B}$ and $\mathbf{U}$. After then, we designed two separate convolution layers with the parameters $\mathbf{W}_{\mathbf{B}}^{(1)}$ and $\mathbf{W}_{\mathbf{U}}^{(1)}$ for $\mathbf{B}$ and $\mathbf{U}$, respectively, in order to capture their own heterogeneous structural dependencies in the network.

The proposed GCN-based opinion model can be applied alone to predict the unknown opinions $\{\omega_i\}_{i\in\mathbb{V}\setminus\mathbb{L}}$. The parameters θ of the model can be estimated by minimize the following cross-entropy scores related to $\mathbf{B}$ and $\mathbf{U}$ based on the observed node-level opinions $\{\omega_i\}_{i\in\mathbb{L}}$:

$$\min_{\theta} \mathcal{L}_{\mathbf{B}}(\theta) + \mathcal{L}_{\mathbf{U}}(\theta), \tag{11}$$

where

$$\mathcal{L}_{\mathbf{B}}(\theta) = -\sum_{i\in\mathbb{L}} \Big[b_i \log f_{1,i}(\mathbf{X},\mathbf{A};\theta) + (1-b_i)\log(1-f_{1,i}(\mathbf{X},\mathbf{A};\theta))\Big] \tag{12}$$

$$\mathcal{L}_{\mathbf{U}}(\theta) = -\sum_{i\in\mathbb{L}} \Big[u_i \log f_{2,i}(\mathbf{X},\mathbf{A};\theta) + (1-u_i)\log(1-f_{2,i}(\mathbf{X},\mathbf{A};\theta))\Big] \tag{13}$$

However, we observed that the designed convolution layers in our proposed GCN-based opinion model are incapable of effectively handling the second type of dependencies (the inherent relational dependencies) between B and U, associated with the domain knowledge of opinions in SL. This motivates a VAE-based opinion model to be developed to enhance our proposed GCN-based opinion model with respect to accuracy prediction of unknown opinions.

B. VAE-based Opinion Model

The key underlying idea is to transform the combinations (opinions) of $\mathbf{B}$ and $\mathbf{U}$ to their equivalent Beta PDFs in order to model their inherent relational dependencies. Based on this transformation, a VAE-based opinion model can be naturally developed.

As introduced in Section III-A, each opinion $\omega_i = (b_i, d_i, u_i)$ can be defined in the equivalent form of a Beta PDF, Beta(α_i, β_i), where

$$\alpha_i = r_i + W\cdot a,\ \beta_i = s_i + W(1-a), \quad r_i = W\cdot b_i/u_i, s_i = W\cdot d_i/u_i, \tag{14}$$

and the parameters a and W are predefined. Denote by $\mathbf{z}_i \in [0,1]^P$ a vector of latent probability variables for each node $i \in \mathbb{V}$ that are sampled from the beta pdf:

$$z_{i,j} \sim \text{Beta}(\alpha_i, \beta_i),\ j = 1, \cdots, P.$$

Denote $\mathbf{Z} = (\mathbf{z}_1, \cdots, \mathbf{z}_N)$. As the probability values in $\mathbf{Z}$ are sampled from node-level opinions, the graph structural information encoded by node-level opinions can be well captured by $\mathbf{Z}$ as well. We can then explore this important pattern using the framework of VAE, in which $\mathbf{Z}$ is considered the encoded latent variables. The **encoder** of VAE can be defined as follows:

$$q(\mathbf{Z}|\mathbf{X},\mathbf{A}) = \prod_{i=1}^{N}\prod_{j=1}^{P} \text{Beta}(z_{i,j}|\alpha_i, \beta_i), \tag{15}$$

where the parameters α_i and β_i are calculated based on b_i and u_i. A **decoder** then uses the latent variables $\mathbf{Z}$ to recover the structural information in adjacency matrix $\mathbf{A}$:

$$p(\mathbf{A}|\mathbf{Z}) = \prod_{i=1}^{N}\prod_{j=\mathcal{N}_i} p(A_{i,j}|\mathbf{z}_i, \mathbf{z}_j), \tag{16}$$

where $p(A_{i,j} = 1|\mathbf{Z}) = \sigma([\psi^{-1}(\mathbf{z}_i)]^T[\psi^{-1}(\mathbf{z}_j)])$ and $p(A_{i,j} = 0|\mathbf{Z}) = 1 - p(A_{i,j} = 1|\mathbf{Z})$, $\psi^{-1}(\cdot)$ is the reverse CDF (cumulative density function) of a standard Gaussian distribution that converts a probability to a real value, and $\sigma(\cdot)$ is the logistic sigmoid function, $\mathcal{N}_i$ is the neighbor of node i. The resulting negative variational lower bound $\mathcal{L}$ or loss function with respect to the parameters θ has the form:

$$\mathcal{L} = -\mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}\Big[\log p(\mathbf{A}|\mathbf{Z})\Big] + \text{KL}\Big[q(\mathbf{Z}|\mathbf{X},\mathbf{A})\|p(\mathbf{Z})\Big] \tag{17}$$

where KL$[q(\cdot)\|p(\cdot)]$ is the Kullback-Leibler divergence between $q(\cdot)$ and $p(\cdot)$. We take a prior of Beta distribution $p(\mathbf{Z}) = \prod_{i=1}^{N}\prod_{j=1}^{P} p(z_{i,j}) = \prod_{i=1}^{N}\prod_{j=1}^{P} \text{Beta}(z_{i,j}|\alpha_0, \beta_0)$, where the parameters α_0 and β_0 are predefined.

C. Inference Algorithm for Predicting Unknown Opinions

In this section, we will combine our proposed GCN-based opinion model and VAE-based opinion model to allow effective and efficient prediction of unknown opinions. The key underlying idea is to jointly optimize the loss functions of these two model in order to estimate the parameters θ:

$$\min_{\theta} \lambda \mathcal{L}(\theta) + \mathcal{L}_{\mathbf{B}}(\theta) + \mathcal{L}_{\mathbf{U}}(\theta), \tag{18}$$

where λ is a trade-off parameter. Note that the expectation $\mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}[\log p(\mathbf{A}|\mathbf{Z})]$ in $\mathcal{L}$ does not have an analytic form and the expectation can be approximated by taking a number of sample $\{\mathbf{Z}^{(n)}\}_{n=1}^{K}$, where $\mathbf{Z}_n \sim q(\mathbf{Z}|\mathbf{X},\mathbf{A})$. We need to ensure that each sample $\mathbf{Z}^{(n)}$ is a function of θ, but the "*reparameterization trick*" is not applicable to Beta distribution, as it does not have the differentiable non-centered parametrization [17]. Instead, we use *Kumaraswamy* distribution to approximate each beta distribution $q(z|\alpha,\beta)$ with the same parameters. *Kumaraswamy* distribution with the parameters α and β has the pdf function is given by:

$$\tilde{q}(z|\alpha,\beta) = \alpha\beta(z)^{\alpha-1}(1-z^{\alpha})^{\beta-1}. \tag{19}$$

We can then have the reparameterization as:

$$z \sim (1-u^{1/\beta})^{1/\alpha}, \tag{20}$$

where $u \sim \text{Unif}(0,1)$. In order to generate the samples $\{\mathbf{Z}^{(n)}\}_{n=1}^{K}$, we first generate the samples $\{u_{i,j}^{(n)}\}$, where $i = 1,\cdots,N$, $j = 1,\cdots,P$, and $n = 1,\cdots,K$. Then the samples can be obtained as:

$$\left\{ z_{i,j}^{(n)} := \left(1 - (u_{i,j}^{(n)})^{1/\beta_i}\right)^{1/\alpha_i} \right\}. \tag{21}$$

Problem (18) can then be approximated as:

$$\min_{\theta} \lambda \tilde{\mathcal{L}}(\theta) + \mathcal{L}_{\mathbf{B}}(\theta) + \mathcal{L}_{\mathbf{U}}(\theta), \tag{22}$$

where

$$\tilde{\mathcal{L}}(\theta) = -\frac{1}{K}\sum_{n=1}^{K}\Big[\log p(\mathbf{A}|\mathbf{Z}^{(n)})\Big] + \text{KL}\Big[q(\mathbf{Z}|\mathbf{X},\mathbf{A})\|p(\mathbf{Z})\Big]. \tag{23}$$

Our inference algorithm to predict unknown opinions based on the hybrid approach combining the GCN-based opinion model with the VAE-based opinion model can be designed using the framework of back propagation. The key steps are described in Algorithm 1. After initialization of the mini-batch size K and the learning rate η, we first solve the parameter estimation problem (11) of our GCN-based opinion model to obtain initial setting of the parameters θ. Then we iterate forward and backward passes until convergence on the estimated parameters θ. The forward pass evaluates the loss function (23) based on the samples $\{u_{i,j}^{(n)}\}$ from $\text{Unif}(0,1)$. The backward pass obtains the gradient of the loss function (23) via the chain rule. The forward pass takes $O(KN)$ and the backward pass takes $O(KM)$ due to the approximations based on Chevyshev polynomials in GCN [15], where N and M refer to the numbers of nodes and edges in the input network, respectively. The total algorithmic running time is hence $(O(L(KN+KM)))$, where L is the number of iterations. As shown in our experiments, L scales constant with respect to M and accordingly our algorithm scales linearly with respect to the total number of edges.

Algorithm 1: GCN-AVE based Opinion Prediction

Input: $\mathbb{G} = (\mathbb{V}, \mathbb{E}, \mathbf{A}, y)$ and $\{\omega_i\}_{i\in\mathbb{L}}$
Output: $\{\omega_i\}_{i\in\mathbb{V}\setminus\mathbb{L}}$
$\ell = 1$;
$K = 16$; (Set the mini-batch size)
$\eta = 0.001$; (Set the learning rate)
Estimate the initial $\theta^{(\ell)}$ by solving Problem (11);
repeat
  Sample $\{u_{i,j}^{(n)}\} \sim \text{Unif}(0,1)$, for $i = 1,\cdots,N$, $j = 1,\cdots,P$, and $n = 1,\cdots,K$;
  Forward pass to compute **B**, **U**, then Calculate $\{\mathbf{Z}^{(n)}\}_{n=1}^{K}$ via Eq. (21) ;
  Backward pass via the chain-rule for gradient $g^{(\ell)} = \nabla_\theta[\tilde{\mathcal{L}}(\theta^{(\ell)}) + \mathcal{L}_{\mathbf{B}}(\theta^{(\ell)}) + \mathcal{L}_{\mathbf{U}}(\theta^{(\ell)})]$
  Update parameters using step size η via $\theta^{(\ell+1)} = \theta^{(\ell)} - \eta \cdot g^{(\ell)}$
  $\ell = \ell + 1$;
until *convergence*
$[\mathbf{B},\mathbf{U}] = f(\mathbf{X},\mathbf{A};\theta^{(\ell+1)})$
Calculate $\{\omega_i\}_{i\in\mathbb{V}\setminus\mathbb{L}}$ based on **B** and **U** via Eq. (14) **return** $\{\omega_i\}_{i\in\mathbb{V}\setminus\mathbb{L}}$

VI. Results and Analysis

A. Experimental Settings

1) Semi-synthetic Epinions dataset: We use the *Epinions* dataset [1] representing a who-trust-who in an online social network. This is a directed network consisting of 47,676 users (i.e., vertices) and 467,468 relationships (i.e., edges). As there are no ground truth opinions available from the dataset, we use a benchmark simulation model [19] to generate synthetic opinions. The simulation model has the following main steps:

1) **Initialization**: 10% of the edges are randomly selected and set the trust of the edges to 1's meaning that i trusts j (but not necessarily j trusts i) where i and j are users in the given directed network.
2) **Exploration**: 1,000 exploration steps are performed to update trust relationships based on the following trust rule:

$$\text{Trust}(a,b) = 1 \wedge \text{Trust}(b,c) = 1 \rightarrow \text{Trust}(a,c) = 1. \tag{24}$$

The exploration step is used to generate synthetic trust observations on the edges of the network. For each exploration step, we randomly select one edge, identify the rule instances associated with this edge, and generate one observation of the edge (0 or 1) based on the probability of the rule instances, where 1 refers to trust while 0 refers to distrust. By repeating the exploration step 1,000 times, we generate a realization of trust relationships on the edges in the network, in which the observations of 1000 randomly selected edges were generated, and the other edges do not have any observations in this realization. We then conduct the 2nd realization based on the previous one by randomly selecting 5% of the edges and swapping their most recent observations from 1 to 0 or from 0 to 1 that are considered as their new trust observations at the current realization. 1,000 exploration steps are conducted to generate observations to make them consistent with the trust rule. Following this procedure, we generate 2nd, $\cdots$, and T-th realizations.

3) **Performance evaluation**: After conducting the T realizations, each edge then has up to T trust observations and its opinion can be estimated based on its trust observations. We consider a set of candidate values of $T \in \{3, 6, 8, 11, 38\}$ corresponding to different uncertainty ranges that will be explained below. In order to conduct performance evaluation for different network sizes, we randomly sample sub-networks with the number of nodes $N \in \{500, 1000, 5000, 10000\}$ from the original Epinions network, respectively. The *testing edges* are randomly selected from all the edges with the percentages (or test ratios) $\in \{20\%, 40\%, 60\%, 80\%\}$ and are predicted based on the observations and known opinions of the other edges which are *training edges*.

2) Road traffic datasets: We collected live road traffic data from June 1, 2013 to March 31, 2014 across two cities from INRIX [2], Washington D.C. and Philadelphia (PA), as summarized in Table I. The raw INRIX dataset collected live traffic speed information from trucks per five-minute interval. A road link has a live speed measurement at a specific time interval if it has at least one truck traversing this link at the time interval; otherwise, it will be a missing speed value. In addition, the reference speed information for each road link per hour interval. A reference speed is defined as the "uncongested free flow speed" for each road segment [3]. It is calculated based upon the 85-th percentile of the measured speed for all time periods over a few years, where the reference speed serves as a threshold separating two traffic states, *congested* vs. *uncongested*. The road traffic dataset for each of the two cities has 43 weeks in total. An hour is represented by a specific combination of hours of a day ($h \in \{6, 9, 12, ..., 21\}$), days of a week ($d \in \{1, 2, 3, 4, 5\}$), and weeks ($w \in \{1, 2, ..., 43\}$): (h, d, w). We only considered work days from Monday ($d = 1$) to Friday ($d = 5$) and hours from 6AM ($h = 6$) to 9PM ($h = 21$).

TABLE I
Description of the three real-world datasets

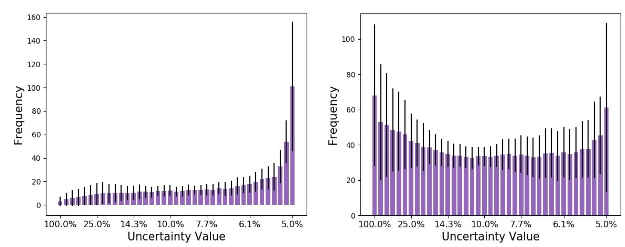
Dataset name	# nodes	# edges	# weeks	# snapshots (hours) in total
Epinions	47,676	477,468	-	-
Washington, D.C.	1,383	1,878	43	3440
Philadelphia	603	708	43	3440

Preprocessing of the networks: The congestion labels in these datasets refer to edges (i.e., road links), but not nodes (intersections). As our proposed approach is for node-level opinion inference, we converted the DC and PA road networks to new networks, in which each node represents a road link and each edge indicates its end nodes (i.e., road links) are adjacent in their original road network. We note that the same preprocessing is also conducted for the Epinions dataset.

Groundtruth opinions (beliefs and uncertainties) of training and testing edges in each dataset. For each road traffic dataset, the opinion of a specific (training or testing) link s at an hour (h, d, w) is estimated based on the observations of the same hour in previous T weeks $\{x_{s,h,d,w}, x_{s,h,d,w-1}, ..., x_{s,h,d,w-T+1}\}$ as the evidence, where $x_{s,h,d,w}$ refers to the congestion observation (0 or 1) of the link s at hour (h, d, w) and T refers to a predefined time window size. **Some of the observations were unobserved, as only a subset of the links were traversed by the delivery trucks.** Denote by T_s the number of observations within the T weeks for the link s and $0 \leq T_s \leq T$. The belief, disbelief, and uncertainty mass variables b_s, d_s, and u_s of a specific link s are estimated by:

$$\begin{aligned} b_s &= \left(\sum\nolimits_{t=0}^{T-1} x_{s,h,d,w-t} - W \cdot a\right) / (T_s + W) \\ d_s &= \left(T - \sum\nolimits_{t=0}^{T-1} x_{s,h,d,w-t} + W \cdot a\right) / (T_s + W) \\ u_s &= W / (T_s + W), \end{aligned} \tag{25}$$

where we set the non-informative prior weight (i.e., an amount of uncertain evidence with $W = 2$) and the base rate (i.e., prior knowledge with $a = 0.5$). As T is the maximum number of possible observations a link can have within a time window of size T, it can be used to calculate a lower bound on the uncertainty of a link as $W/(T+W)$, and the upper bound will be 100%. For example, for $T = 38$, the range of uncertainties of the links is $[5\%, 100\%]$.

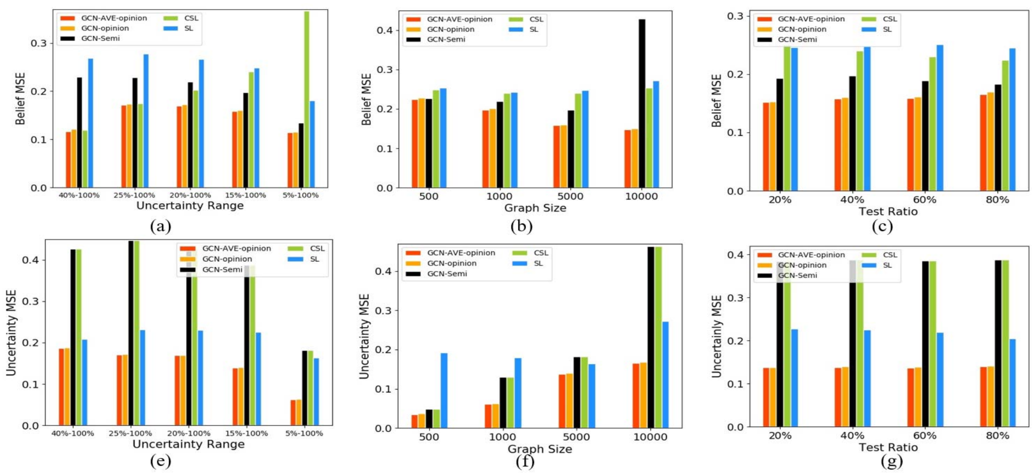


(a) Uncertainly value of PA Dataset (b) Uncertainly value of DC Dataset

Fig. 3. Uncertainly value distribution

Fig. 3 shows the distributions of means and standard deviations of uncertainties of all the road links in the DC and PA datasets for all the time windows of size 38, respectively. It indicates that the uncertainties of road links in PA are skewed towards small values (i.e., 5%) while the distribution of uncertainties in the DC dataset is more towards uniform. This allows us to more rigorously validate the performances of uncertain opinion prediction methods under two different distributions of opinions.

3) Parameter settings: The main parameters for all the datasets include T (time window size) and TR (test ratio or the percentage of edges that are tested). We tested different window sizes $T \in \{3, 6, 8, 11, 38\}$ corresponding to the the uncertainty ranges $[25\%, 100\%]$, $[20\%, 100\%]$,



Effect of uncertainty range under N = 5000 and TR=40%

Effect of graph size (N) under T = 11 and TR=40%

Effect of TR under N = 5000 T=11

Fig. 4. Belief-MSE and Uncertainty-MSE under the semi-synthetic network based on Epinions dataset.

$[15\%, 100\%]$, and $[5\%, 100\%]$, respectively. Due to space restriction, we only showed the results for $T = 38$ and the uncertainty region $[15\%, 100\%]$. For other time window sizes, we also observed similar trends. The values of TR are set to $\{20\%, 40\%, 60\%, 80\%\}$.

4) Performance metrics: Based on Eq. (25), the uncertainty mass, u_s, for each training or testing edge is a known and constant value, u, after the window size T is predefined, without the actual observations of this link. For this reason, the empirical analysis based on the road traffic datasets focuses on the comparison between the proposed methods and comparable methods based on the three main metrics: *Belief Mean Squared Error* (B-MSE), *Uncertainty Mean Squared Error* (U-MSE), and computation time (in sec.).

The metrics B-MSE and U-MSE are defined as:

$$\text{B-MSE}(\boldsymbol{\omega}_{\mathbb{V}\setminus\mathbb{L}}) = \frac{1}{N}\sum\nolimits_{i\in\mathbb{V}\setminus\mathbb{L}} |b_i - b_i^\star| \tag{26}$$

$$\text{U-MSE}(\boldsymbol{\omega}_{\mathbb{V}\setminus\mathbb{L}}) = \frac{1}{N}\sum\nolimits_{i\in\mathbb{V}\setminus\mathbb{L}} |u_i - u_i^\star| \tag{27}$$

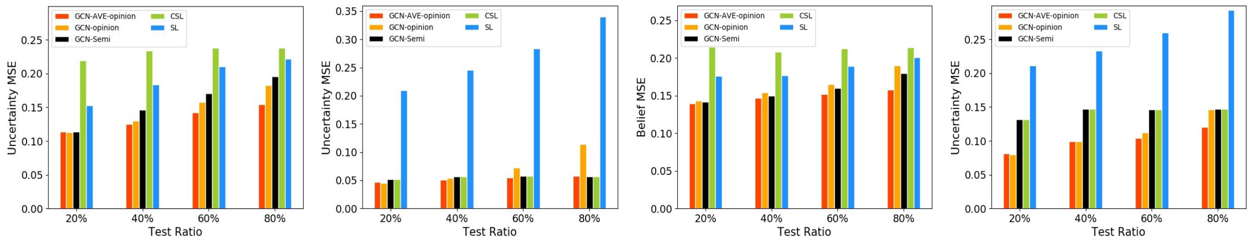
where $\omega_i = (b_i, d_i, u_i, a)$ and $\omega_j^\star = (b_i^\star, d_i^\star, u_i^\star, a)$ refer to the predicted and true opinions of a target variable y_i associated with node i, respectively.

5) Comparison methods: We notate our proposed GCN-based opinion model as *GCN-opinion* and our proposed opinion model based on the combination of the GCN and the AVE opinion models as *GCN-AVE-opinion*. We compared our proposed methods with the comparable two counterpart methods: *SL* [14], *CSL* [6], and *GCN-semi* for semi-supervised node classification [15]. Note that **the competitive method CSL** is not directly comparable to our proposed methods because CSL was designed for the scenario where all the node-level opinions in a network have the same uncertainties but different beliefs or disbeliefs. However, in this work, we consider varying uncertainties across nodes. We employed the following procedure to predict the missing values for the training edges, such that CSL can be used: For each road link i, we first estimated its opinion based on its available observations within the current time window of size T, and then use its equivalent Beta PDF to sample binary observations for its missing observations within the time window. After this procedure, each training edge has the number of observation (T) and hence the same uncertainty values. For **the competitive method GCN-Semi**, it was designed for semi-supervised node classification, but not directly for opinion inference. We made the following modifications to adapt GCN-Semi for opinion inference: For each time interval within a time window, we applied GCN-Semi to predict the congestion labels of the testing edges and the missing labels of the training edges simultaneously. Then, for each testing edge, we obtained its T observations within the time window of size T. In addition, we used these observations to directly estimate its opinion. Following this strategy, as all the testing links have the same number of observations (T), their predicted uncertainties will be identical. Note that it is not trivial to adapt GCN-Semi to predict varying node-level uncertainties.

6) Parameter Tuning: SL only has one hyperparameter that is the maximum length of its independent paths. We set this to 18 as we observed that the results of SL are almost the same for the maximum lengths equal to or greater than 18. CSL does not have hyperparameters for tuning. Our proposed methods, GCN-opinion and GCN-AVE-opinion, have three hyper parameters: λ (the trade-off parameter), η (the learning rate), K (the mini-bach size), and P (the dimensionality of the latent encoded vectors), and dropout (a parameter in GCN). We set $\lambda = 0.01$, $\eta = 0.001$, $K = P = 16$, and dropout = 0.1 for all the experiments. All these hyperparameters are estimated based on the observations of the training edges.

B. Experimental Results based on Semi-Synthetic Datasets

Fig. 4 shows the comparative analysis of our proposed GCN-opinion and GCN-AVE-opinion and the three counterpart methods in terms of Belief-MSE and Uncertainty-MSE under the semi-synthetic Epinions dataset. Fig. 4 (a) and (e) demonstrates that GCN-opinion and GCN-AVE-opinion



(a) Belief MSE, PA Dataset (b) Uncertainly MSE, PA Dataset (c) Belief MSE, DC Dataset (d) Uncertainly MSE, DC Dataset

Fig. 5. Comparison of our proposed methods (GCN-opinion and GCN-AVE-opinion) and counterpart methods: Belief and Uncertainty MSEs vs. Test Ratios

outperform all these other methods based on Belief-MSE and Uncertainty-MSE with respect to varying the ranges of uncertainties, including $[25\%, 100\%]$, $[20\%, 100\%]$, $[15\%, 100\%]$, and $[5\%, 100\%]$. As the lower bound of a uncertainty range decreases, both Belief-MSE and Uncertainty-MSE of GCN-opinion and GCN-AVE-opinion decrease. This implies that GCN-opinion and GCN-AVE-opinion performs even better under larger ranges of uncertainties. We note that the trend pattern of CSL is different from the others in that when the lower bound of the uncertainty range decreases, its Belief MSE increases. This can be explained that, as the original CSL does not support missing values in training links, we fill in the missing values with random numbers as discussed above. A large uncertainty range (e.g., [5%, 100%]) refers to a large window size (e.g., 38) that will lead to more random numbers, which may potentially cause a high Belief MSE of CSL.

Fig. 4 (b) and (f) shows the effect of a network size on Belief-MSE and Uncerainty-MSE under all comparing schemes. It is clear that GCN-opinion and GCN-AVE-opinion outperform their counterparts on both the metrics. There is an interesting pattern that when a graph size increases, Belief-MSE decreases for GCN-opinion and GCN-AVE-opinion, where Uncertainty-MSE increases for these two methods. It implies that the task of uncertainty prediction may be more challenging for larger network data. Fig. 4 (c) and (g) demonstrates the effect of test ratio on both MSE metrics of all compared schemes. Obviously, GCN-opinion and GCN-AVE-opinion outperform among all in both the MSE metrics, except that they perform comparable to CSL for some of the settings (TR = 20%, 60%, 80%) for Uncertainty-MSE. Different from the above effect of the uncertainty range and graph size, both the Belief-MSE and Uncertainty-MSE do not show clear sensitivity under different test ratios.

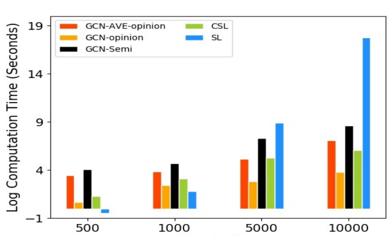


Fig. 6. Comparison of computation time on Epinion dataset.

Fig. 6 shows the log computation times as the number of nodes increases. Except SL whose computation time increases exponentially when the network size increases, the other methods almost scale linearly with respect to the network size. GCN-opinion and CSL are the most efficient methods among all the methods. More discussions about the computation times of these methods are presented in the below section for the real-world datasets, in which we summarized the observations for both the semi-synthetic and real-world datasets on the computation time.

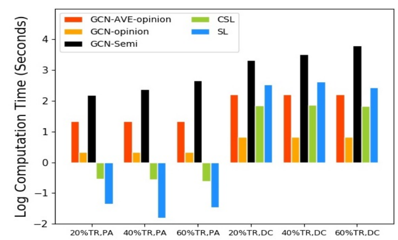


Fig. 7. Comparison of computation time on real traffic dataset.

C. Experimental Results based on Real-World Datasets

Fig. 5 compares the performances of our proposed GCN-based methods (GCN-opinion and GCN-AVE-opinion) with those of the three counterpart methods (i.e., CSL, SL, and GCN-Semi) with respect to Belief-MSE and Uncertainty-MSE based on two road traffic datasets (PA and DC). The results indicate that GCN-AVE-opinion performs the best among all in both prediction of beliefs and uncertainties. GCN-opinion performs the second best, but it is less sensitive than GCN-AVE-opinion across different test ratios and shows that both Belief-MSE and uncertainty-MSE in GCN-opinion significantly increase as test ratio increases. When the test ratio is low (i.e., 20%), both Belief-MSE and Uncertainty-MSE in these two methods are comparable under certain settings while GCN-AVE-opinion outperforms GCN-opinion and other methods when the test ratios exceeds 20%. Although the uncertainty-MSEs in CSL and GCN-Semi are comparable to that in GCN-AVE-opinion for the PA dataset as shown in Fig. 5 (b), their uncertainty-MSEs are more than 80% higher than that of GCN-AVE-opinion for the DC dataset as shown in Fig. 5 (d).

Fig. 7 shows the average snapshot-level log computation times across different test ratios (i.e., 20%, 40%, and 60%) on the two road traffic datasets obtained for PA and DC. When the network size increases, the running complexity of SL increases in an exponential order while those of the other methods (including GCN-opinion and GCN-AVE-opinion) increase in a linear order. In particular, GCN-opinion and

CSL show the lowest computation time. GCN-AVE-opinion shows lower computation time than CGN-Semi and SL, but higher computation time than GCN-opinion and CSL due the complexity of combining the GCN-based model with AVE-based model. However, GCN-AVE-opinion still scales almost linearly in proportion to network size. GCN-Semi has the highest computation time, because the model needs to run on the snapshot of the network for each time slot while the other methods only run once for each time window and the total number of time slots is 38 times the total number of time windows, where 38 is the fixed size of the time windows.

VII. Conclusion and Future Work

In this work, we propose a novel DL-based opinion inference approach based on GCN and AVE techniques to address the key challenges of scalability and handling heterogeneous opinions in network data. From the simulation experiments conducted in this work, our **key findings** are:

- Overall our proposed GCN-AVE-opinion method outperforms all other counterparts in both Belief-MSE and Uncertainty-MSE. In particular, our GCN-AVE-opinion method shows less sensitivity over a wide range of test ratios, implying high resilience, compared to GCN-opinion, SL, CSL, and GCN-Semi.
- The performance order in Belief-MSE follows: GCN-AVE-opinion $>$ GCN-opinion $>$ GCN-Semi $>$ SL $>$ CSL. The performance order in Uncertainty-MSE follows: GCN-AVE-opinion $>$ GCN-opinion $>$ GCN-Semi $\approx$ CSL $>$ SL.
- The higher performance of GCN-based methods is because they are capable of modeling heterogeneous structural dependencies among node-level beliefs and uncertainties.
- The higher performance of GCN-AVE-opinion over GCN-AVE is because GCN-AVE-opinion integrates an AVE-based opinion model to model the inherent relational dependencies between beliefs and uncertainties based on mapping of their combinations (opinions) to their equivalent PDFs.
- AVE-GCN-opinion scales almost linearly in proportion to the network size and is scalable for large-scale network data.

In our future work, we plan to conduct: (1) the validation the performance of our DL-based approach based on more real-world datasets (e.g., cybersecurity datasets); and (2) the extension of our proposed work to address uncertainty-based online opinion inference problems.

Acknowledgments

This work is partially supported by ARL's Competitive Basic Research Program under Computational and Information Sciences Directorate and by the US Army Research Office under grant number W911NF1720129. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of ARL or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

[1] "Epinions," http://www.trustlet.org/downloaded_epinions.html.

[2] "Inrix," http://inrix.com/publicsector.asp.

[3] "Reference speed for congestion evaluation," http://www.inrix.com/scorecard/methodology.asp/.

[4] S. H. Bach, M. Broecheler, B. Huang, and L. Getoor, "Hinge-loss markov random fields and probabilistic soft logic," *arXiv preprint arXiv:1505.04406*, 2015.

[5] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv preprint arXiv:1312.6203*, 2013.

[6] F. Chen, C. Wang, and J.-H. Cho, "Collective subjective logic: Scalable uncertainty-based opinion inference," in *IEEE BigData*, 2017, pp. 7–16.

[7] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *NIPS*, 2016, pp. 3844–3852.

[8] P. Frasconi, M. Gori, and A. Sperduti, "A general framework for adaptive processing of data structures," *ITNN*, vol. 9, no. 5, pp. 768–786, 1998.

[9] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *IJCNN'05*, pp. 729–734.

[10] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NIPS*, 2017, pp. 1025–1035.

[11] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *ACHA*, vol. 30, no. 2, pp. 129–150, 2011.

[12] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," *arXiv preprint arXiv:1506.05163*, 2015.

[13] M. Ivanovska, A. Jøsang, L. Kaplan, and F. Sambo, "Subjective networks: Perspectives and challenges," in *GSKPR*. Springer, 2015, pp. 107–124.

[14] A. Jøsang, *Subjective Logic: A Formalism for Reasoning Under Uncertainty*. Springer, 2016.

[15] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *CoRR*, abs/1609.02907, 2016.

[16] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein, "Geometric deep learning on graphs and manifolds using mixture model cnns," in *Proc. CVPR*, vol. 1, no. 2, 2017, p. 3.

[17] E. Nalisnick and P. Smyth, "Stick-breaking variational autoencoders," in *ICLR*, 2017.

[18] C. E. Rasmussen, "Gaussian processes in machine learning," in *ALMR*. Springer, 2004, pp. 63–71.

[19] M. Richardson, R. Agrawal, and P. Domingos, "Trust management for the semantic web," in *International semantic Web conference*. Springer, 2003, pp. 351–368.

[20] G. Shafer *et al.*, *A mathematical theory of evidence*. Princeton university press Princeton, 1976, vol. 1.

[21] R. Singh, J. Ling, and F. Doshi-Velez, "Structured variational autoencoders for the beta-bernoulli process," Technical report, 2017.

[22] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.