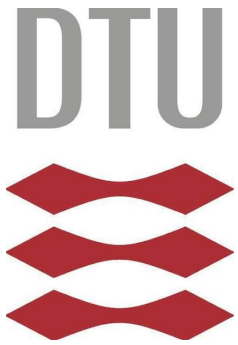

DANMARKS TEKNISKE UNIVERSITET

Project - Report

May 27, 2021

Authors:
Electra Zarafeta, s202238
Maria Garofalaki, s202378

42186 Model - Based Machine Learning

DTU Compute

Contents

1 Introduction

The objective of this project is to infer to a graph's latent space using Variational Inference in order to successfully represent an undirected graph into a latent space and predict the class of each node. Graphs represent objects and their relationships in the real world, popular examples are the social networks, biological networks, road networks and many more can be represented using graphs. The non-regularity of data structures have led to advancements in Graph Neural Networks in relation to tasks such as classification, predictions, etc. Recently, Kipf and Welling [T. N. Kipf and Welling 2017] proposed the Graph Convolutional Network (GCN), which is considered one of the basic Graph Neural Network variants. In the GCNs the model learns the features by inspecting the neighboring nodes. Inspired by the paper "Neural Relational Inference for Interacting Systems" [T. Kipf et al. 2018] we implemented a Variational Encoder-Decoder Structured Classifier which receives as input a graph, meaning its adjacency matrix and dictionary, and outputs the classification result. The relaxation of the discrete latent state will be executed by using the Concrete distribution (CONtinuous disCRETE) proposed in the paper "Variational Inference for Graph Convolutional Networks in the Absence of Graph Data and Adversarial Settings" [Elinas, Bonilla, and Tiao 2020]. Our goal is to develop a model which will learn a good latent representation of graphs, while accurately predicting the node's subject.

2 Dataset and Exploratory Data Analysis

The dataset which will be used is the **Cora** dataset which consists of 2708 scientific publications classified into one of the seven subjects. The possible values of the subjects are: "Case Based", "Genetic Algorithms", "Neural Networks", "Probabilistic Methods", "Reinforcement Learning", "Rule Learning" and "Theory". The specific dataset illustrates the citations (5429 links) between different scientific publications. Each publication in the dataset is described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary and the dictionary consists of 1433 unique words[1]. In Figure 1 the number of nodes-scientific publications and the number of edges-citations are presented and compared for the various subjects of the dataset. By inspecting this figure we can conclude that the subject "Probabilistic Methods" shows the highest values of nodes and connections which makes the dataset quite unbalanced.

3 Model description

3.1 Model Architecture

In Figure 2 the architecture of the neural network is being presented. The Variational Encoder-Decoder Classifier is consisted of an encoder and a decoder. Its goal is to find a way to encode the input dataset into a compressed form (latent space) in such a way that the classified labels are as close as possible to the target. More precisely the input of our neural network consists of a dictionary, containing the 2708 scientific publications where each publication is described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary (1433), and a adjacency matrix, which corresponds to the citation network of 5429 links-edges. The Adjacency matrix represents the edges or the connections between the nodes.

The **Encoder** consists of 3-layers Graph Convolutional NNs which follow the architecture 1433-500-40-5 with fully connected layers and ReLU nonlinearities. This results to the extraction of the hyperparameters for the prior $p(z)$, the logits, and by using the Gumbel-Softmax distribution we move to the **Latent Space Representation** of the graph. As mentioned in the Section 1 the chosen distribution to sample the latent variable is the Gumbel-Softmax (proposed in the paper [Elinas, Bonilla, and Tiao 2020]) which is considered the ideal distribution in order to achieve the relaxation over both the prior and the approximate posterior of the discrete latent state. The reason for using a Concrete distribution is that the estimation of the posterior over the adjacency matrix under a highly non-linear likelihood (given by the GCN's output) is intractable and even in the approximate inference world, carrying out posterior estimation over a very large discrete combinatorial space can prove extremely hard [Elinas, Bonilla, and Tiao 2020]. The **Decoder** follows the architecture of 2-layers Graph Convolutional NNs with 7-7 neurons and ReLU nonlinearities, in order to classify the 2708 scientific publications over the 7 subjects. Furthermore, the number of epochs was set to 1000 and for the optimization of the model the Adam with the learning rate set to $1e-2$ was used.

[1] `https://relational.fit.cvut.cz/dataset/CORA`

3.2 Stochastic Variational Inference (SVI)

In order to perform the classification, we introduce latent variables $z \in Z$ and model the joint distribution as $p_\theta(y|z)p(z)$, where $p(z)$ is a simple distribution that is usually assumed to be known. The complex conditional distribution (Decoder) $p_\theta(y|z)$ parametrized by θ is modeled in terms of a deep network. When learning the model, the Variational Classifier approximates the posteriors $p_\theta(z|x)$ by an amortized inference model (Encoder) $q_\phi(z|x)$ parametrized by ϕ. During the training, the Evidence Lower BOund (ELBO) of the model's log-likelihood $L(\theta) = E_p \log p_\theta(y)$ is maximized. The **ELBO** can be expressed in the following equivalent form:

$$L(\theta, \phi) = E_p[E_q \log p_\theta(y|z) - D_{KL}(q_\phi(z|x)||p(z))]$$

The ELBO is a lower bound to this log evidence and it can be shown that the gap between the ELBO and the log evidence is given by the KL divergence between the guide and the posterior. In other words, ELBO represents a measure of "closeness" between two distributions. For a fixed θ, we take steps in ϕ space that increase the ELBO and we decrease the KL divergence between the guide and the posterior, i.e. we move the guide towards the posterior.

3.3 Probabilistic Generative Model

The **Model** is a subject classifier $q_\phi(y|x)$ that randomly "fills in" y_i given a latent representation z_i. Each y_i is generated by a latent random sample z_i. The non-linear dependency between y_i and z_i is parameterized by the neural network with parameters θ. In addition, we sample from multinomial (or categorical) prior for the class label. Based on the Probabilistic Graphical Model (PGM) for $p_\theta(y|z)$ presented in Figure 3, the Generative process would be:

1. Draw $\theta \sim N(\theta|0, \lambda I)$
2. For each random variable z_n
   (a) Draw $y_n \sim Categorical(y_n|z_n, \theta)$

Consequently in order to infer in the above model we need to specify a flexibly guide. The basic role of the **Guide** is to "fill in" latent random variables. The guide $q_\phi(z|x)$ is parameterized by a global parameter ϕ shared by all the datapoints. The goal of inference is to guess "good" values for the latent random variables and thus the following conditions are satisfied:

1. the log evidence $\log_\theta(y)$ is large so as our model to be a good fit to the data
2. the guide $q_\phi(z|x)$ provides a good approximation to the posterior

Both in the Model and the Guide the prior $p(z)$ follows the Gumbel-Softmax distribution since the dataset is discrete and it is essential to apply the relaxation of the latent space as explained in the Section 3.1. In Figure 4 the Probabilistic Graphical Model for the Guide is presented and above its Generative Process is introduced:

1. Draw $\phi \sim N(\phi|0, \lambda I)$
2. For each random variable x_n
   (a) Draw $z_n \sim Gumbel - Softmax(z_n|x_n, \phi)$

4 Results and Discussion

The results from using VI in order to successfully represent an undirected graph into a latent space and predict the class of each node can be seen in Figures 5 and 6. We trained our model with a combination of the train and validation set in order to have a sufficient number of observations during training and in Figure 5 the reduction of the training loss is presented while running inference with `Pyro`. With the use of the `Predictive` class from `Pyro` we extracted samples for the test set and achieved an accuracy of 68.2%. Finally, so as to better understand our results we used the T-SNE method (Figure 6) to reduce the dimensionality of the latent z and visualize the various subjects of each node-scientific publication.

Given the achieved accuracy, as well as the latent state representation, it is noticeable that several adjustments could be taken into consideration in order to improve the performance of the Classifier. In regards to the future directions, it would be beneficial to modify the NN architecture and, for example, use other Graph Convolutional NNs such as the `SAGEConv` from `Pytorch`. Moreover, except of inferring into the latent space to predict the classes, it would interesting to investigate the impact of inferring into the NN's hyperparameters (i.e. weight, bias) as well.

5 Contribution

Report:

- *Electra Zarafeta* Sections 1, 2, 3.1, 4, Visualizations
- *Maria Garofalaki* Sections 2, 3.2, 3.3

Notebook: Both Electra Zarafeta and Maria Garofalaki worked on the implementation of the presented objectives. Given the project's complexity the members worked together in the coding procedure while reading useful material and thus the contribution was equal in all the different parts that the notebook is consisted of, i.e. the data analysis, the NN combined with VI and the results presentation.

6 Appendix

A Distribution of nodes and edges per subject

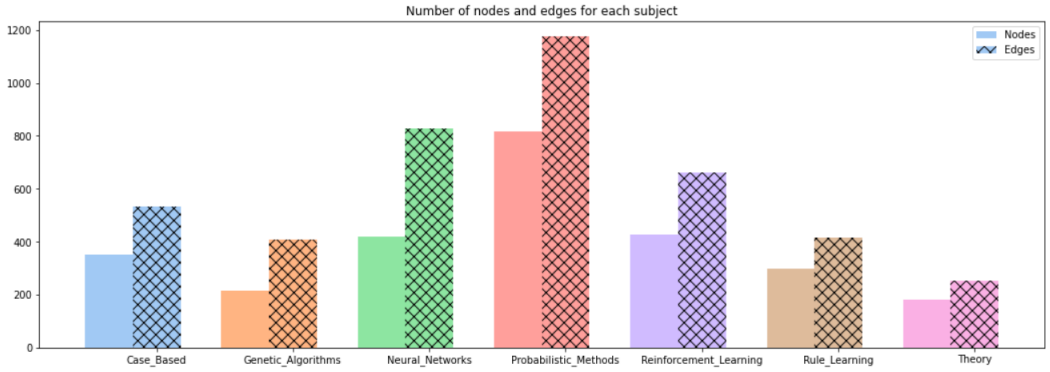

Figure 1: Number of nodes and edges for each subject

B Model Architecture

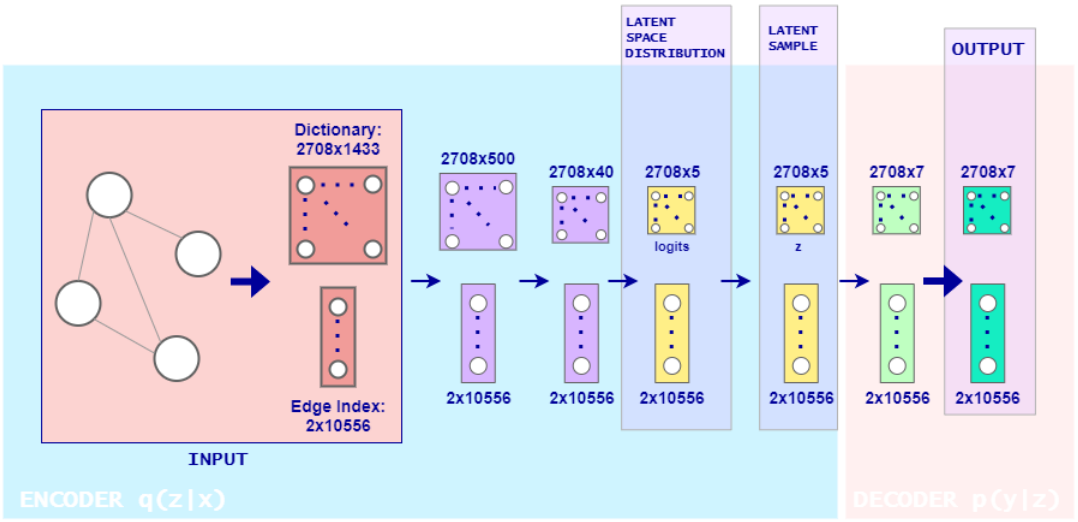

Figure 2: Variational Encoder-Decoder Structured Classifier Model Architecture

C PGM for the model and the guide

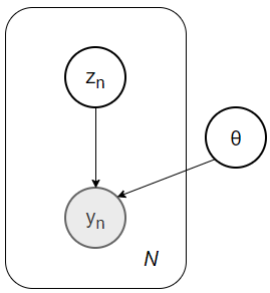

Figure 3: Probabilistic Graphical Model for the model - $p_\theta(y|z)$

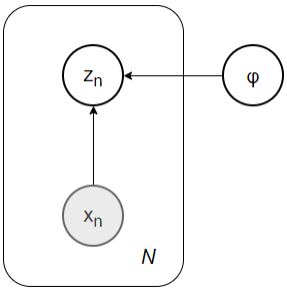

Figure 4: Probabilistic Graphical Model for the guide - $q_\phi(z|x)$

D Training Loss

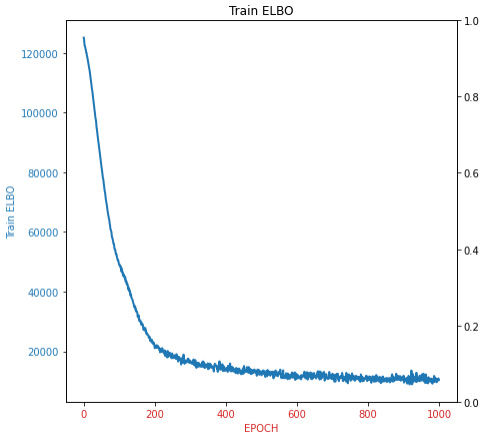

Figure 5: Training ELBO

E Latent Space Representation

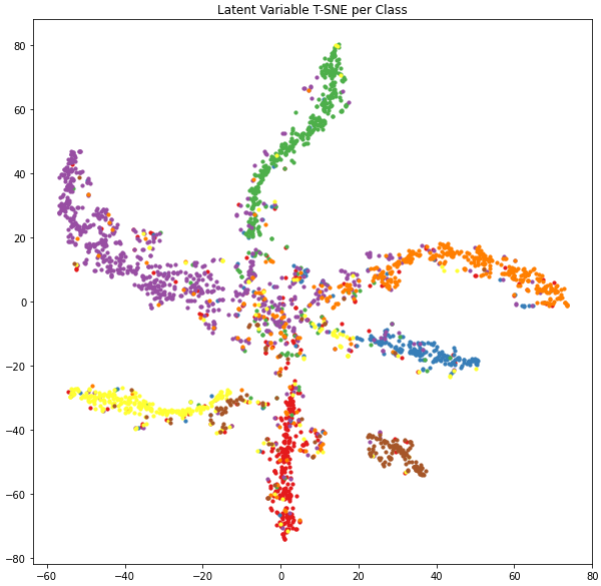

Figure 6: Latent Space Representation for the entire dataset

7 References

[1] Pantelis Elinas, Edwin V. Bonilla, and Louis Tiao. *Variational Inference for Graph Convolutional Networks in the Absence of Graph Data and Adversarial Settings*. 2020. arXiv: 1906.01852 [cs.LG].

[2] Thomas Kipf et al. "Neural Relational Inference for Interacting Systems". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by Jennifer Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, Oct. 2018, pp. 2688–2697. URL: http://proceedings.mlr.press/v80/kipf18a.html.

[3] Thomas N. Kipf and Max Welling. *Semi-Supervised Classification with Graph Convolutional Networks*. 2017. arXiv: 1609.02907 [cs.LG].