PGM PROJECT - DELIVERABLE 1

Mohammad Nur AMIN
mdnur.amin@etu.univ-st-etienne.fr

Jatin BABBAR
jatin.babbar@etu.univ-st-etienne.fr

Shirin BASHIRIAMID
shirin.bashiriamid@etu.univ-st-etienne.fr

Madhubhani RANCHA GODAGE
madhubhani.rancha.godage@etu.univ-st-etienne.fr

Orhan SOLAK
orhan.solak@etu.univ-st-etienne.fr

January 28, 2021

1 Problem Intuition

The problem is related to multi-class classification, where we have a job description in the form of text along with a gender of a candidate, and the objective is to predict the job category while maintaining the model fairness. The dataset for this task is a set of job descriptions and the gender of candidates. The train set contains 217,197 samples, and the test set, which is used for the evaluation of the challenge, has 54,300 samples. There are a total of 28 unique labels (job categories).

Based on the frequency of sample for each job category, the train dataset is very biased, with the most frequent category being professor (class 19) with 70,016 samples, to the least frequent category being rapper (class 21) with 783 samples. Consequently, we use a subset of the original data, which is sampled so that there are same number of samples belonging to each class.

We represent the probability distribution for the observed dataset and the goal is taking advantage of them to learn the distribution that describes well the current data based on probabilistic model.To make predictions for unlabeled test set using probabilistic models, we use a Multinomial Naive Bayes model and a Latent Dirichlet Allocation model [Blei et al., 2003] [Pritchard et al., 2000] and Gibbs Sampling.

2 Bayesian Network Representation

2.1 Multinomial Naive Bayes

In the text classification based on probabilistic model the goal is finding the category that a description belongs to, also we need to find θ_i that generates the description.

In the first step we have documents with a certain category and vocabularies as an input as the output we gain $(\theta_1, \theta_2, ...\theta_k)$ that is a set of category which is a word distribution and θ_i represents category i.

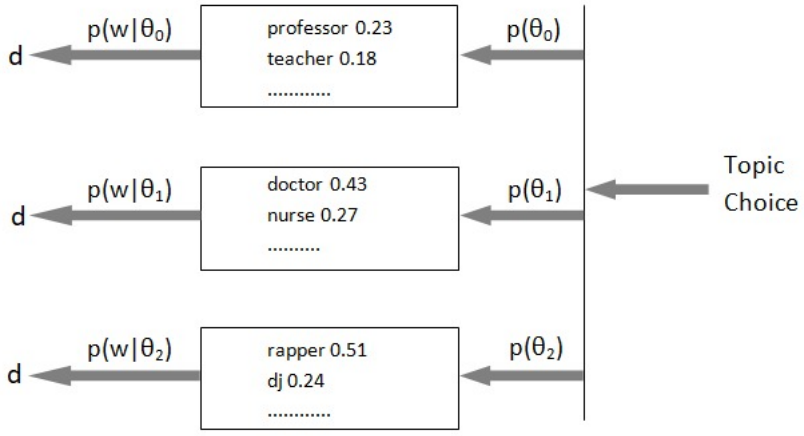

Figure 1: Generation

$P(\theta_i)$ is our prior that estimates which category is more likely before observation any descriptions.

$P(w|\theta_i)$ is a likelihood probability of content of descriptions.

$$Category(description) = argMax_i P(\theta_i|d)$$

where $P(\theta_i|d)$ is the posterior, the probability of the category after we observed descriptions.

$$\begin{aligned} \mathrm{p}\left(\theta_{\mathrm{i}} \mid \mathrm{d}\right) &= \frac{\mathrm{p}\left(\mathrm{d} \mid \theta_{\mathrm{i}}\right) \mathrm{p}\left(\theta_{\mathrm{i}}\right)}{\mathrm{p}(\mathrm{d})} \\ &= \frac{\mathrm{p}\left(\mathrm{d} \mid \theta_{\mathrm{i}}\right) \mathrm{p}\left(\theta_{\mathrm{i}}\right)}{\sum^{\mathrm{k}} p\left(\mathrm{d}+\theta_{\mathrm{j}}\right) \mathrm{p}\left(\theta_j\right)} \end{aligned}$$

A Naive Bayes classifier uses probability theory to classify data. The key point of Bayes theorem is that the probability of an event can be adjusted as new observation is introduced. The reason for we call it naive is, all properties of a data point under consideration are independent of each other. The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as **tf-idf** may also work. [mul,]

Multinomial Naive Bayes is a probabilistic learning method and the probability of a document being in class computed as:

$$P(\theta \mid d) \propto P(\theta) \prod_{1 \leq k \leq n_d} P\left(t_k \mid \theta\right)$$

- d : document
- n_d: number of tokens in d.
- $P(\theta)$: The prior probability of a document occurring in class θ
- $P\left(t_k \mid \theta\right)$: The conditional probability of term t_k occurring in a document of class θ.
- $\langle t_1, t_2, \ldots, t_{n_d} \rangle$: The tokens in d are part of the vocabulary which are used for classification.

We interpret $P\left(t_k \mid \theta\right)$ as a measure of how much evidence t_k contributes that θ is the correct class. If a document's terms do not provide clear evidence for one class versus another, we choose the one that has a higher prior probability. For example, $\langle t_1, t_2, \ldots, t_{n_d} \rangle$ for the one-sentence in a document "He runs a boutique design studio attending clients in the United States" might be $\langle$ run, boutique, design, studio, attend, client, United, States $\rangle$, with $n_d = 8$, after eliminating stop words, applying stemming and lemmatization.[Christopher D. Manning and Schütze, 2008]

In multiclass text classification, our goal is to find the best class for the each observation. The best class in NB classification is the most likely or maximum a posteriori (MAP) class c_{map} :

$$\theta_{\text{map}} = \arg\max_{\theta \in \mathbb{C}} P(\theta \mid d) = \arg\max_{\theta \in \mathbb{C}} P(\theta) \prod_{1 \leq k \leq n_d} P\left(t_k \mid \theta\right)$$

2.2 Latent Dirichlet Allocation

The Latent Dirichlet Allocation (LDA) is a popular example of topic models used in the field of natural language processing among other applications in machine learning. Using this model, each document can be represented as a combination of different topics (classes), and each topic is represented as a combination of different words.

In the case of text classification, the assumption is that each document is a combination of a small set of topics, and each topic is a small set of frequently used words. The LDA builds a topic per document model and words per topic model as prior Dirichlet distributions.

We define the LDA model using the following variables:

- V: index for the number of words in the vocabulary
- K: number of topics (in our case, 28 classes)
- M: number of documents (in our case, 21,924 samples)
- N_i: number of words in document i
- $Dir(\alpha)$: prior Dirichlet distribution for topics in a document
- $Dir(\beta)$: prior Dirichlet distribution for words in a topic
- θ: multinomial distribution for topics in a document
- ψ: multinomial distribution for words in a topic
- z_{ij}: topic for word i in document j
- w_{ij}: word i in document j

The process of LDA is as follows:

1. Choose $\psi \sim Dir(\alpha)$
2. Choose $\theta \sim Dir(\beta)$
3. For each i=1,....,M and j=1,...,N_i
   (a) Choose topic $z_{ij} \sim Multinomial(\theta_i)$
   (b) Choose topic $w_{ij} \sim Multinomial(\psi_{z_{ij}})$

So, with this process, we are maximizing the likelihood of the posterior state estimation for each document and the words associated to each topic [Lee et al., 2018].

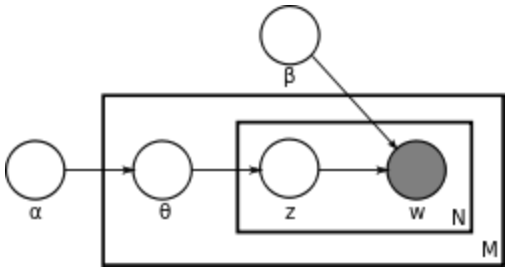

Figure 2: Bayesian Network Representation for the Latent Dirichlet Allocation Model

2.3 Gibbs sampling

Gibbs sampling is based on Markov Chain Monte Carlo (MCMC). A Markov Chain is a stochastic process in which future states are independent of past states, given the current state. MCMC are used when it is not possible to draw samples from $p(x)$ but rather only able to evaluate $p(x)$ up to a normalizing constant.

2.3.1 Generative Model

We represent each document as a bag of words. Given an unlabeled document W_j, our goal is to pick the best label $\mathrm{L}_j = 0.....27$. Our goal is to choose the label L_j for W_j that maximizes $P(L_j jW_j)$. Applying Bayes's Rule,

$$\begin{aligned}\mathrm{L}_j = \mathrm{argmax}_L \, P\left(L \mid \mathbf{W}_j\right) &= \underset{L}{\mathrm{argmax}} \frac{P\left(\mathbf{W}_j \mid L\right) P(L)}{P\left(\mathbf{W}_j\right)} \\ &= \underset{L}{\mathrm{argmax}} P\left(\mathbf{W}_j \mid L\right) P(L),\end{aligned}$$

where the denominator $P\left(\mathbf{W}_j\right)$ is omitted because it does not depend on L. This application of Bayes's rule allows us to think of the model in terms of a generative story that accounts for how documents are created. According to that story, we first pick the class label of the document, $P\left(\mathbf{L}_j\right)$; our model will assume that's done by throwing a 28 sided dice whose probability is some value $P\left(\mathbf{L}_j = 1...27\right)$. We can express this a little more formally as

$$\mathrm{L}_j \sim \mathrm{Categorical}(\pi)$$

Then, for every one of the R_j word positions in the document, we pick a word w_i independently by sampling randomly according to a probability distribution over words. Which probability distribution we use is based on the label L_j of the document, so we'll write them as $\theta_{0...27}$. Formally one would describe the creation of document j's bag of words as

$$\mathrm{L}_j \sim \mathrm{Multinomial}(R_j, \theta)$$

This is generative story for the creation of a whole set of labeled documents $\langle \mathbf{W}_n, \, \mathrm{L}_n \rangle$ according to the Naïve Bayes model, is that this simple document-level generative story gets repeated N times, as indicated by the N.

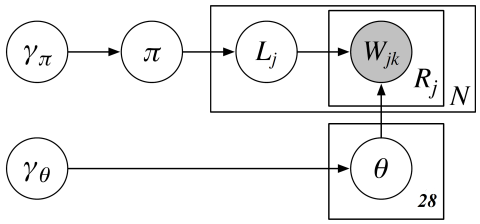

Figure 3: Naive Bayes plate diagram

2.3.2 Priors

Before the whole process begins, we assume that π is picked randomly and sampled from a distribution with γ, referred to as hyperparameters that is used to pick parameters of the model. In Figure 3 these hyperparameters are represented as a vector γ_π. About θ which is described in the sections bellow.

2.3.3 State Space and Initialization

Gibbs sampler walks through a k-dimensional state space defined by the random variables $\langle \mathbf{Z}_1, \mathbf{Z}_2, ..\mathbf{Z}_k \rangle$ in the model. Every point in that walk is a collection of $\mathbf{z}_1, \mathbf{z}_2, ..\mathbf{z}_k$ of values for this variables. In a Naive Bayes model the variables that define the state space are the scaler variable π, vector variable θ and labels L for each N documents. We also have one vector variable W_j for each of the N documents, but these are observed variables and their values are already known. For the initialization a value π is picked by sampling from the γ_π distribution. For each j success probability π, a label is assigned which is the label of the document j at $\mathrm{L}_j^{(0)}$ iteration based on the outcome. At the same time, θ is also initialized sampling from γ_θ

2.3.4 Deriving and simplifying the joint distribution

For each iteration, t of sampling, every variable is updated defining the state space by sampling from its conditional distribution given the other variables. The joint distribution for the entire document collection is

$$P\left(\mathbb{C}, \mathbf{L}, \pi, \boldsymbol{\theta} | \gamma_\pi, \boldsymbol{\gamma}_\theta\right)$$

The joint distribution can decomposed as follows

$$P(\pi \mid \gamma_\pi)\, P(\mathbf{L} \mid \pi) P(\boldsymbol{\theta} \mid \gamma_\theta)\, P(\mathbb{C} \mid \boldsymbol{\theta}, \mathbf{L})$$

3 Gender Debiasing

Gender debiasing can be performed either by preprocessing text before fitting the classifier or by post processing after fitting the classifier. To address the gender biasing issue, we have adopted the Local Interpretable Model-Agnostic Explanations (LIME) approach in the preprocessing stage. In LIME, the instances are perturbed around its neighborhood while model's predictions behavior is being observed. These instances are then weighted by their proximity to the original example. Afterwards, an interpretable model is learnt on those associated predictions. The interpretable model provides us with the probable features (both positively and negatively impacted) that were most impacted during making the predictions.

References

[mul,] Naive bayes classifier for multinomial models. `https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html`. Accessed: 2021-01-27.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

[Christopher D. Manning and Schütze, 2008] Christopher D. Manning, P. R. and Schütze, H. (2008). Text classification and naive bayes. Ch 13:6–7.

[Lee et al., 2018] Lee, J., Kang, J.-H., Jun, S., Lim, H., Jang, D., and Park, S. (2018). Ensemble modeling for sustainable technology transfer. *Sustainability*, 10(7):2278.

[Pritchard et al., 2000] Pritchard, J. K., Stephens, M., and Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959.