Mestrado em Engenharia Informática
Dissertação
Relatório Final

Supervised Topic Models with Multiple Annotators

Mariana Rodrigues Lourenço
mrlouren@student.dei.uc.pt

Orientador:
Bernardete Ribeiro
Filipe Rodrigues
Data: 6 de julho de 2015

FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Mestrado em Engenharia Informática
Dissertação – Relatório Final

Estágio: [1856] – Supervised Topic Models with Multiple Annotators
Autor: Mariana Rodrigues Lourenço (mrlouren@student.dei.uc.pt)
Orientador DEI: Bernardete Ribeiro (bribeiro@dei.uc.pt)
Juri Arguente: Ernesto Jorge Fernandes Costa (ernesto@dei.uc.pt)
Juri Vogal: Filipe Araujo (filipius@dei.uc.pt)

Abstract

We live in an era where information overflows. Yet, for this information to become knowledge, it has to be given meaning. This thesis focuses on a machine learning approach that evolved from probabilistic graphical models, which automatically extracts knowledge from vast amounts of data by assigning themes to documents: topic modeling. Topic models are an emergent technique used for both descriptive and predictive tasks. As a result, it was soon extended to other goals that do not only model topics, but also target variables.

This work presents a supervised topic model that is able to learn from crowds. That is, we consider the case where the label set of the data was provided by multiple annotators. In the multi-annotator setting, the ground truth labels need to be modeled from several noisy versions of them given by the different annotators. To address this sort of problems, it is often assumed that all labelers are equally reliable through the use of voting techniques, which was proven to be an unrealistic conjecture. On the contrary, the proposed model takes into account the different levels of expertise and biases of annotators, by jointly modeling them together with the topics and the true labels. In order to make this process computationally tractable, a variational inference algorithm was developed, which provides an efficient approximate inference method.

We finalize by showing how general supervised topic models can be used to predict demand in special events by correlating internet search query data with real measurements of transport usage, thus, motivating the usage of the topic models in real-world applications.

Acknowledgements

Firstly, I would like to express my gratitude to my advisor, Prof. Bernardete Ribeiro, and to my co-advisor, Filipe Rodrigues. I have been given the best guidance by my advisor and an endless support from my extremely patient co-advisor. I have grown and learned a lot this year thanks to you.

I have to thank Prof. Francisco Câmara Pereira for giving me the opportunity to be working in Singapore MIT Alliance for Research and Technology centre, for his precious advices and, of course, for the motivational speeches.

I am also grateful to Prof. Ana Alves, who introduced me to the research project InfoCrowds. InfoCrowds opened me the doors for the scientific research world and ended up guiding me to this thesis project.

An obvious thanks to my family for the encouragement, enthusiasm and for always showing pride in me. I really will try to pay you all the support back someday.

My thanks also goes to my friends: João Girão, Bruno Nabais, João Guilherme, Daniel Pimentel, Manuel Gaspar, Rúben Costa, Adriana Ferrugento and Bruno Correia, who helped me a lot this year. I thank you for being present and for trying to seem interested in my work. Especially to Bruno Correia, I have to thank you for making it possible for this thesis to be delivered while I was 14 hours flight away.

To my grandfather José.

We are drowning in information and starving for knowledge. John Naisbitt.

Contents

List of Tables

List of Figures

Nomenclature

m^d	Variational probability of the mean of the target x^d
α	Dirichlet corpus-level parameter
α^r	Sensitivity of the r^{th} annotator
$\bar{z}^d$	Mean of all z_n^d of the document d
β^r	Specificity of the r^{th} annotator
β_k	Word proportions of topic k
$\beta_{i,j}$	Probability of word j under the i^{th} topic
$\boldsymbol{\beta}$	Set of all β_k
$\boldsymbol{\eta}$	Set of all η_c
$\boldsymbol{\gamma}$	Set of all γ^d
$\boldsymbol{\lambda}$	Set of all λ^d
$\boldsymbol{\nu}$	Set of all ν^d
$\boldsymbol{\phi}_{1:D}$	Set of all ϕ_n^d
$\boldsymbol{\pi}^r$	Set of all $\pi_{c,l}^r$ of annotator r
$\boldsymbol{\theta}$	Set of all θ^d
$\boldsymbol{c}$	Set of all c^d
$\boldsymbol{m}$	Set of all m^d
$\boldsymbol{w}^d$	Set of all w_n^d of document d
$\boldsymbol{w}_{1:D}$	Set of all w_n^d
$\boldsymbol{y}^d$	Set of all $y^{d,r}$ of document d
$\boldsymbol{y}_{1:D}$	Set of all $y^{d,r}$
$\boldsymbol{z}^d$	Set of all z_n^d of the document d
$\boldsymbol{z}_{1:D}$	Set of all z_n^d
η_c	Coefficient of class c
$\mathbb{E}_q[.]$	Expectation under the variational distribution q
$\Gamma(.)$	Gamma function
γ_d	Variational probability of topic proportions θ^d

λ^d	Variational probability of latent class c^d
$\mathcal{D}$	Dataset
$\mathcal{L}$	Lower bound
ν^d	Variational probability of the variance of the target x^d
ϕ_n^d	Variational probability of topic assignment z_n^d
$\phi_{n,i}^d$	Variational probability that the n^{th} word is generated by latent topic i
$\pi_{c,l}^r$	Probability that the annotator r provides the label l, given a true class c^d
$\Psi(.)$	Digamma function
θ^d	Topic proportions of document d
b^r	Bias of annotator r
C	Number of classes
c^d	Class of document d
c^n	Class of the n^{th} instance
D	Number of documents
$H(q)$	Entropy of the variational distribution
K	Number of topics
N	Number of instances
N^d	Number of words in document d
R	Number of annotators
T	Class-label-dependent linear transformation
V	Size of the vocabulary
v^r	Variance of annotator r
w_n^d	n^{th} word of document d
$w_{n,j}^d$	Indicator if the j^{th} word under the vocabulary is equal to the n^{th} word of document d
x^d	Target value of document d
x^n	n^{th} instance
$y^{d,r}$	Label assigned by the annotator r to the document d

$y^{n,r}$	Label assigned by the annotator r to the n^{th} instance
z_n^d	Topic assignment for the n^{th} word of document d
z_n^d	Topic assignment of the n^{th} word of document d
$z_{n,i}^d$	Probability that the n^{th} word is generated by latent topic i

Chapter 1

Introduction

1.1 Motivation

As the phenomenon of big data grows, the need for automated methods of data analysis becomes increasingly evident. Machine learning provides an answer to this problem by comprising a set of theoretical concepts and methods capable of managing complex, dynamical and heterogeneous sorts of data. Furthermore, this field of artificial intelligence allows data to be explained, patterns about the data to be uncovered and it gives the possibility to make predictions about the future given some past data. In fact, machine learning algorithms can be applied to solve a broad spectrum of previously unsolved problems, which may explain its spreading across several areas of science and industry.

This thesis focuses on a statistical approach on machine learning: topic modeling. Topic models are a powerful tool to automatically assign "themes" to text instances, i.e., weighted lists of semantically coherent words. This can be extremely useful for summarization, information retrieval, categorization, dimensionality reduction and prediction tasks. This means that problems like indexing articles by theme or suggesting a book to a user based on other books he likes can be easily addressed by employing topic modeling techniques. Models like Latent Dirichlet Allocation (LDA) brought an efficient way of analyzing large corpora and, consequently, are being explored and extended for many different purposes. In fact, they have now many applications that go beyond their original goal of modeling textual data, such as analyzing images, videos, survey data or social networks data. However, since the data to be modeled is frequently associated with other variables such as labels, tags or ratings and considering that the distributions of the documents over topics may act as feature sets to train a predictor, an extension to classifi-

cation/regression problems was an obvious next step in topic models. As we will show, by combining the learning of the topics distributions with a regression or classification model, better prediction performances are obtained comparing to the separate use of the two methods.

Supervised topic models, like Supervised LDA (sLDA), explore this combination by jointly modeling the topics and the target variables of the documents, which means that the topics are influenced by words co-occurrences and the relationship between documents and their labels. However, these models assume the existence of at least one label per document, which may represent a real challenge when it is too expensive to label every instance of the data or even when there are no ground truth targets defined.

This sort of challenges motivates this work. A specific example is the participation of multiple medical experts in diagnosis as a substitute for expensive medical procedures, such as biopsies, since the identification of the actual disease or condition might require costly invasive tests. Instead of the biopsy, labels can be assigned by multiple experts. For instance, when determining if a patient has cancer or not, a group of radiologists may examine images of the suspicious region and individually give an opinion about it. Unfortunately, experts may be specialized in different fields, which can result in non-consensual opinions. Actually, in nearly all sorts of contexts it is highly unlikely to have annotators with equal behavior. Hence, annotator-aware models are needed, in order to get the best out of noisy answers.

Other examples of the necessity for annotator-aware models are tasks related to product rating in online stores and webpage or image tagging, in websites like LabelMe and Delicious[1]. This type of tasks are naturally thought to be fulfilled by crowdsourcing, either because they are too subjective to be considered by only one labeler or because the size of the data makes it impossible for a single person to examine it. Product rating or semantic analysis, for example, are subjective tasks, since two people are likely to have different and possibly biased views on the same product or media object.

Similarly, Named Entity Recognition, Keyphrase Extraction, Word Sense Disambiguation and Handwritten Character Recognition are tasks usually more accurately performed by humans than by current artificial methods. For instance, Passonneau et al. (2010) presented a study on Word Sense Disambiguation involving multiple annotators and Huang & Suen (1995) developed a work on recognition of handwritten numerals combining multiple experts.

On the other hand, a dataset comprising millions of instances would be too time consuming to be labeled by a single individual. Consider the case

[1]http://www.delicious.com

of LabelMe: how would the project build such a large labeled dataset (about 700 thousand objects) if there was only one volunteer assigned to that task?

Crowd-sourced data is, indeed, a solution to bear in mind when dealing with such scenarios, either because it is a simple inexpensive way to answer these challenges or because it is demonstrated that learning from labels provided by multiple annotators can be as good as learning from the labels of a single expert (Snow et al., 2008). This is why there are now online plataforms, like Amazon Mechanical Turk, which make it easy to post HITs (Human Intelligence Tasks) to obtain annotations from multiple workers. Particularly in an era where web is becoming increasingly social, it becomes inevitable to follow this shift by taking advantage of it, but at the same time adapting to its needs. In this context, following this shift means designing models that are able to deal with multiple annotators.

But how can the labels from multiple annotators be integrated to produce a single label? The majority voting method could be applied. However, this would be relying on the assumption that all labelers are equally good, which was demonstrated to be an unrealistic supposition in Snow et al. (2008) and Rodrigues et al. (2013). Some annotators may be trustworthy, others may me be ill-intentioned. Furthermore, not everyone is equally good across all subjects. If the majority of the labelers are not reliable, the resulting labels would be too noisy.

In other words, motivating this work is the inevitable adaptation of statistical models to crowdsourced data. This means breaking the assumption that all labelers are equivalent, which, expectedly, could lead to the generation of flawed models.

Finally, a motivation for this work is also the fact that it is part of the research project InfoCrowds. InfoCrowds's primary objective is to exploit online information about public events, mobility data and event-specific surveys to build interpretative and predictive models of flows of people and their transportation mode in the city. Besides the fact that this information comes mainly from the Internet, which enhances the need for models capable of learning from the crowds, the challenges related to transport planning and operations in large events for transit agencies are an example of how supervised topic models can predict on real-world problems. These large events not only imply stress to the system on an irregular basis, but their associated mobility behavior is also difficult to predict. The importance of prognosis on non-habitual transport overcrowding for a transit agency is undeniable and textual probabilistic approaches like supervised topic models are very well suited to its solution.

1.2 Objectives

This thesis aims to generalize supervised LDA in such a way that multiple annotators with different levels of expertise and biases are considered. This is possible by designing a supervised topic model that introduces new latent variables to account for the heterogeneity of the multiple annotators in terms of reliability and developing a new Bayesian inference algorithm. The resultant model jointly learns the topics and the true classes of each document, as well as the annotators accuracies and biases, even in the absence of the ground truth labels.

Moreover, we want to show how powerful supervised topic models can be on real-world applications, by using internet search query data to predict overcrowding hotspots. Thus, we propose a way for transportation companies to start planning special events as early as they are announced on the web. This implies studying which model fits better the problem, how the data should be processed and how to interpret the results in order to improve them.

1.3 Thesis structure

This thesis starts with Chapter 2, which describes the state of the art by briefly explaining probabilistic graphical models, approximate inference and by presenting some relevant examples of both topic models and learning from crowds methods.

Chapters 3 and 4 introduce the proposed model: multiple-annotator supervised latent Dirichlet allocation in its both classification and regression versions, respectively. In Chapter 5, the experiences conducted to evaluate them are presented and the results discussed.

Following, in Chapter 6 we present the application of supervised topic models in the real-world problem of using internet search queries to predict human mobility in social events.

In Chapter 7, the work done in the first and second semesters is described, comparing the plan and objectives outlined with the ones achieved.

Finally, Chapter 8 draws the final remarks.

Chapter 2

State of the art

This thesis covers concepts of topic modeling techniques and learning from crowds methods, strongly connecting these two areas. These are, in turn, closely related to the field of probabilistic graphical models and, consequently, to inference mechanisms, which is how the unknown variables of the graphical models are estimated given the observed ones. This means that, in order to present the supervised topic model that learns from multiple annotators that is the main contribution of this thesis, it is, firstly, necessary to frame it as a probabilistic graphical model, to explain its inference process and distinguish it from some similar works. Therefore, this chapter starts by providing background concepts on probabilistic graphical models and approximate inference, continues with a discussion about topic models and presents some methods for learning from crowds.

Since this thesis also proposes an application of supervised topic models for human mobility prediction, this chapter ends with a briefly review about travel demand modeling state of the art approaches.

2.1 Probabilistic graphical models

Probabilistic graphical models are a framework for establishing relationships between variables. The idea is to combine probability theory with graphs in a way that allows statistical models to be represented intuitively and illustratively. A graphical model is formed by nodes, edges that connect them and plates. Nodes correspond to variables in the domain. Shaded nodes (like c in Figure 2.1) represent observed variables and unshaded nodes the latent ones. Edges represent statistical dependencies between variables and plates are rectangles that indicate that the variables inside them are

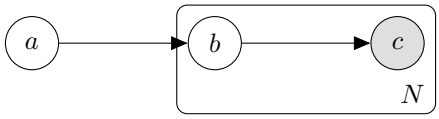


Figure 2.1: Example of a graphical model.

repeated the number of times denoted in the bottom right corner. This means that a graphical model encodes a probability distribution over a high-dimensional space. A well known subclass of probabilistic graphical models are Bayesian networks, whose graph is directed. In Bayesian networks, each node only depends on the ones that point to them. For instance, looking at Figure 2.1, it can be seen that c depends on b and b depends on a. Therefore, the dependencies between variables in the distribution are clearly stated in the graph, which makes the factorization of the joint distribution $p(a, \boldsymbol{b}, \boldsymbol{c})$ intuitive. In the case of the example of Figure 2.1, the factorization would be:

$$p(a, \boldsymbol{b}, \boldsymbol{c}) = p(a) \prod_{n=1}^{N} p(b_n|a)p(c_n|b_n), \tag{2.1}$$

where $\boldsymbol{b} = \{b_n\}_{n=1}^{N}$ and $\boldsymbol{c} = \{c_n\}_{n=1}^{N}$ are vectors (of size N) and, thus, represented in bold.

So, instead of assigning every possible values to all the variables in the model, the joint probability of all random variables becomes a product of conditional distributions. Consider the case in which a, b and c are binary-valued variables. In this setting, the factorization 2.1 asserts that the joint probability is obtained by multiplying $1 + 2 \times N$ numbers, instead of having $2 \times (2 \times 2)^N$ possible values. Consequently, instead of $2 \times (4)^N - 1$ non-redundant parameters, this parametrization requires only $1 + (2 + 2) \times N$ non-redundant parameters.

2.2 Approximate inference

A probabilistic graphical model can be used to answer various types of questions about the data being modeled. Since it defines the joint probability distribution of the variables, it becomes possible to estimate unknown quantities from known ones, i.e., to compute the posterior distribution of random variables given observed ones. This process is called probabilistic inference,

in particularly, if Bayes' rule is used to estimate unknown variables, it is called Bayesian inference.

From the Bayes' theorem, we know that the probability of the quantity of interest, z, given all the data collected about it, x, is formulated by:

$$\overbrace{p(z|x)}^{posterior} = \frac{p(x,z)}{p(x)} = \frac{\overbrace{p(x|z)}^{likelihood}\overbrace{p(z)}^{prior}}{\underbrace{p(x)}_{evidence}}. \tag{2.2}$$

Therefore, supposing that z is a disease needed to be inferred and x is the observed symptom, $p(z|x)$ (the probability of the disease given the symptom) can be calculated by multiplying $p(z)$ (a measurable quantity of the probability of the disease) and $p(x|z)$ (the probability of the symptom given the disease, which can be obtained from the case histories of the disease). Fortunately, $p(x)$ does not need to be measured, since, according to the sum rule of probability and considering that z is discrete:

$$p(x) = \sum_z p(x|z)p(z). \tag{2.3}$$

However, there are cases in which exact answers are infeasible to compute, which means that the only solution is to approximate those answers. Approximate inference algorithms turn the computation of posterior distributions in probabilistic graphical models into a tractable problem, by trading off computation time for accuracy.

Two large classes of approximate inference algorithms for high- dimensional distributions are Markov Chain Monte Carlo (MCMC) and variational inference. The MCMC approach is based on Monte Carlo approximations, whose main idea is to use repeated sampling to obtain the desired distribution. MCMC iteratively constructs a Markov chain of samples, which, at the some point, converges. At this stage, the sample draws are close to the true posterior distribution $p(z|x)$, meaning that samples can then be collected to approximate the required expectations.

On the other hand, variational inference methods are deterministic. Given the observed data x and the latent variables z, the goal of variational Bayesian inference is to pick an approximation to the distribution from a tractable family of distributions $q(z)$ and make it as close as possible to the true posterior distribution $p(z|x)$. A tractable family can be obtained by relaxing some constraints in the true distribution. Then, the inference problem is to optimize the parameters of the new distribution (variational parame-

ters) so that the approximation becomes as close as possible to the true posterior. This can be achieved by minimizing the Kullback-Leibler divergence $\mathrm{KL}\big(q(z)||p(z|x)\big)$ between the true distribution and the approximated one, which can be equivalently formulated as maximizing a lower-bound on the log probability of the observations $p(x)$. The Kullback-Leibler divergence is defined to be the following integral:

$$D_{\mathrm{KL}}(q(z)\|p(z|x)) = \int_{-\infty}^{\infty} q(z)\, \ln \frac{q(z)}{p(z|x)}\, \mathrm{d}x. \tag{2.4}$$

Since the logarithmic function is concave, the variational objective function can be defined by using Jensen's inequality to lower bound the log likelihood, as follows:

$$\begin{aligned}
\log p(x) &= \log \int_z p(x,z) \\
&= \log \int_z p(x,z) \frac{q(z)}{q(z)} \\
&= \log \left(\mathbb{E}_q \left[\frac{p(x,z)}{q(z)} \right] \right) \\
&\geq \mathbb{E}_q \left[\log \left(\frac{p(x,z)}{q(z)} \right) \right] \\
&= \mathbb{E}_q \Big[\log p(x,z) \Big] - \mathbb{E}_q \Big[\log q(z) \Big].
\end{aligned} \tag{2.5}$$

This function can be maximized using a coordinate ascent algorithm, which iteratively optimizes each variational distribution, by setting its derivative to zero, keeping the others fixed. The batch coordinate ascent variational inference algorithm can be summarized as follows:

Input : Corpus, variational parameters
Output: Updated variational parameters
1 Initialize global parameters randomly
2 **repeat**
3 **for** *each local variational parameter* **do**
4 Update its estimate using the estimated global variational parameters
5 **end**
6 **for** *each global variational parameter* **do**
7 Update it using the estimated local variational parameters.
8 **end**
9 **until** *convergence*;

Algorithm 1: Batch coordinate ascent variational algorithm

Recently, it was developed a more scalable version of the variational inference algorithm. Stochastic variational inference (Hoffman et al., 2013) is faster than the regular variational inference method (batch variational inference), since it updates the variational parameters using a subsample of the data, instead of the whole dataset. The problem solved by Hoffman et al. (2013)) is the inefficiency caused by the local steps 3 and 4 of the batch coordinate ascent variational inference algorithm, since to move on to the step 6, the entire dataset has to be processed.

Returning to the graphical model example illustrated in Figure 2.1, one can distinguish the global variables (a) from the local ones (b and c). In this case, the batch variational inference algorithm requires that every N points of the data are analyzed before the global variational parameter of a is updated. However, the variational parameter of a may be continuously estimated as more data is observed, especially if it is considered that subsets of the data (*mini-batches*) can provide a noisy representation of the entire dataset. In fact, this is the main difference between the batch variational inference algorithm and the stochastic algorithm. The stochastic variational inference algorithm is the following:

Input : Corpus, variational parameters
Output: Updated variational parameters

```
Initialize global parameters randomly
Compute the step-size schedule ρ^t
repeat
    Subsample one or more data points from the corpus
    repeat
        for each local variational parameter do
            Update its estimate using the current estimated global
            variational parameters
        end
    until variational local parameters converge;
    for each global variational parameter do
        Using the current estimate of the local variational
        parameters, compute its intermediate value
        Update it partially, putting weight ρ^t on the new estimate
        and 1 − ρ^t on the old estimate
    end
until convergence;
```

Algorithm 2: Stochastic coordinate ascent variational algorithm

In this setting, ρ^t is the step-size schedule that depends on a forgetting rate κ, which controls how quickly old estimates are forgotten, and a delay d, that down-weights early iterations: $\rho^t = (t + d)^{-\kappa}$. Suppose that: the dataset N is divided in 10 mini-batches of size S; the algorithm is on its 2^{nd} iteration t; the value of the delay and the forgetting rate is, respectively, 1 and 0.75 and let $\hat{a}$ be the intermediate global parameter. In this setting, ρ is calculated as $\rho = (2 + 1)^{-0.75}$ and the global parameter a^t is equal to $(1 - \rho)a^{t-1} + \rho \times S \times \hat{a} = (1 - 3^{-0.75}) \times a + 3^{-0.75} \times \frac{N}{10} \times \hat{a}$. The reason why the term $S = \frac{N}{10}$ is multiplied with the intermediate global parameter $\hat{a}$ is to make up for the fact that the size of the mini-batch is lower than the size of the entire dataset. Thus, this product simulates that the mini-batch is the same size of the whole data.

Stochastic variational inference gives the model the efficiency needed to handle massive data sets, since it only needs to fit in memory a subsample at a time. Of all the three algorithms presented, it is the most scalable one. Moreover, although MCMC is often easier to implement and applicable to a broader range of models than variational inference, the latter is usually faster (see Bishop et al. (2006)), it is deterministic and it is easy to determine when to stop.

2.3 Topic models

Topic models are a family of algorithms capable of discovering the "topics" of a collection of documents. By extracting text patterns of the corpus, topic models disclose the themes that compose the document collection, which, in turn, make this suite of algorithms an emerging field in machine learning for analyze structured data. The simplest kind of topic model is, at the same time, the most popular one: Latent Dirichlet Allocation (Blei et al., 2003a). Given the importance that it early received, many extensions and adaptations were made to its original algorithm. This section describes both LDA and some of its extensions.

2.3.1 Unsupervised topic models

Latent Dirichlet allocation (Blei et al., 2003a) is a probabilistic graphical model that reveals the semantic properties of words and documents by probabilistic topics. In latent Dirichlet allocation (LDA), each topic is a pattern represented by a distribution over the words present in the vocabulary. Its result is a set of topics and topic proportions associated with each document, meaning that documents are mixtures of topics and topics mixtures of words. Therefore, LDA allows the documents to be represented heterogeneously through its latent semantics.

In Figure 2.2, an article about how data analysis is used to determine the genes an organism needs to survive is illustrated. In the left side of the figure, the extracted topics of this article are listed. As it can be seen, there are four topics: the yellow one is about genetics, the pink one about evolutionary biology, the topic in green is about neuroscience and the blue one is related to data analysis. The words highlighted match the topic of its color, which explains the topic proportions and assignments represented in the right side of the picture. Naturally, one can understand why the most probable topic is the one related to genetics and the absence of the green topic in the histogram.

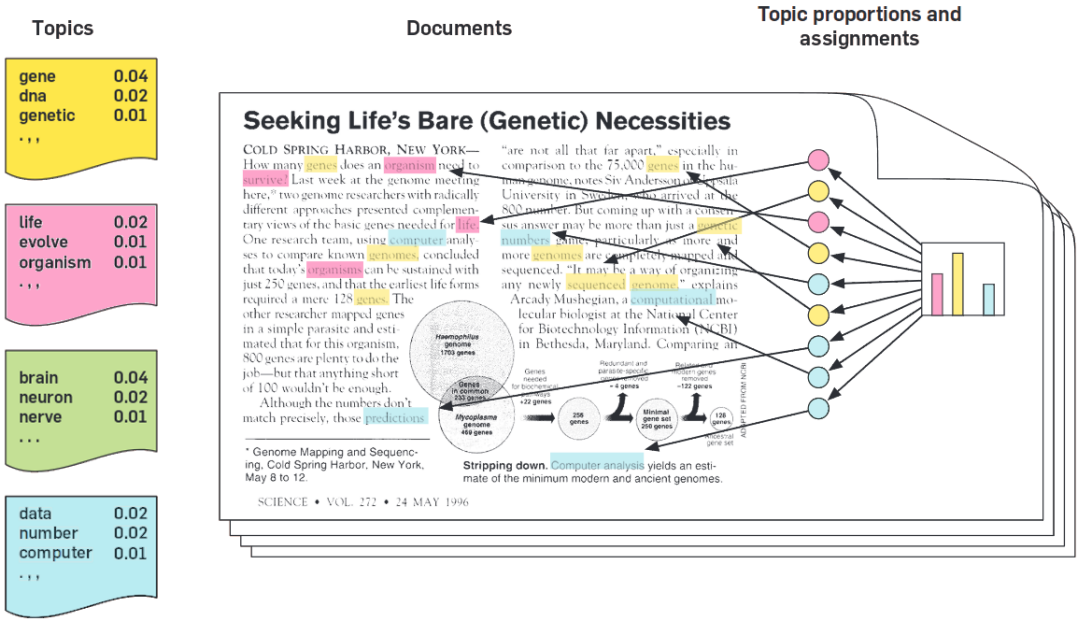


Figure 2.2: Topics example (extracted from Blei (2012)).

Topic 247		**Topic 5**		**Topic 43**		**Topic 56**	
word	**prob**	**word**	**prob**	**word**	**prob**	**word**	**prob**
drugs	.069	red	.202	mind	.081	doctor	.074
drug	.060	blue	.099	thought	.066	dr.	.063
medicine	.027	green	.096	remember	.064	patient	.061
effects	.026	yellow	.073	memory	.037	hospital	.049
body	.023	white	.048	thinking	.030	care	.046
medicines	.019	color	.048	professor	.028	medical	.042
pain	.016	bright	.030	felt	.025	nurse	.031
person	.016	colors	.029	remembered	.022	patients	.029
marijuana	.014	orange	.027	thoughts	.020	doctors	.028
label	.012	brown	.027	forgotten	.020	health	.025
alcohol	.012	pink	.017	moment	.020	medicine	.017
dangerous	.011	look	.017	think	.019	nursing	.017
abuse	.009	black	.016	thing	.016	dental	.015
effect	.009	purple	.015	wonder	.014	nurses	.013
known	.008	cross	.011	forget	.012	physician	.012
pils	.008	colored	.009	recall	.012	hospitals	.011

Table 2.1: Example of four topics extracted from the TASA corpus in Steyvers & Griffiths (2007).

Topic 77		**Topic 82**		**Topic 166**	
word	**prob**	**word**	**prob**	**word**	**prob**
music	.090	literature	.031	**play**	.136
dance	.034	poem	.028	ball	.129
song	.033	poetry	.027	game	.065
play	.030	poet	.020	playing	.042
sing	.026	plays	.019	hit	.032
singing	.026	poems	.019	played	.031
band	.026	**play**	.015	baseball	.027
played	.023	literary	.013	games	.025
sang	.022	writers	.013	bat	.019
songs	.021	drama	.012	run	.019
dancing	.020	wrote	.012	throw	.016
piano	.017	poets	.011	balls	.015
playing	.016	writer	.011	tennis	.011
rhythm	.015	shakespeare	.010	home	.010
albert	.013	written	.009	wonder	.010
musical	.013	stage	.009	field	.010

Table 2.2: Example of three topics extracted from the TASA corpus in Steyvers & Griffiths (2007).

In Table 2.1, another example of a topic model outcome is shown. More particularly, those are four topics inferred from the Touchstone Applied Science Associates corpus (Zeno et al., 1995). The words are downwardly sorted by their probability under the topic, which means that the words that best represent each topic are in the top positions. Clearly, we can observe that topics join the words semantically related. In the topic 247 are words related to drugs, in the topic 5, to colors, in the 43^{rd} topic, to mind and, in the topic 56, words relate to medical visits. Since each document is assigned to a distribution over topics, a document about color theory would have topic 5 as its main topic and a medical article would probably have the 56^{th} and 247^{th} topics as its most likely topics.

Moreover, topics may be useful for disambiguation tasks (e.g. Li et al. (2010)). For instance, in Table 2.2, it can be seen that the word "play" appears in the topics in three different senses. According to the context given by the remaining words, we can infer that the documents generated by the topic 77 use play in "playing music" sense, in the topic 82 "play" means a "theater play" and, in the 166^{td} topic, "play" is associated with sports.

All of these properties are conferred to the LDA model by its generative nature. A generative probabilistic graphical model randomly generates observable data through its latent variables. The idea in generative models is,

for example, given two variables a and b, to estimate the joint distribution $p(a, b)$ and, subsequently, use this distribution to evaluate the conditional $p(b|a)$ in order to make predictions of b for new values of a. In contrast, one could estimate the conditional distribution $p(b|a)$ directly by following a discriminative approach. However, a discriminative approach provides a model only for the response variables conditioned on the observed variables. When fitting a generative model, the goal is to find the best set of latent variables that can explain the observed data.

The generative process under the LDA model for each document $\boldsymbol{w^d} = \{w_n^d\}_{n=1}^{N^d}$ in a corpus $\mathcal{D} = \{\boldsymbol{w}^d\}_{d=1}^{D}$ is the following:

1. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$
   (b) For the n^{th} word
      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Multinomial(\beta_{z_n^d})$

In practice, to sample a new document, a distribution over topics is chosen. Then, for every word in the document, a topic is selected according to the distribution over topics picked and, finally, a word from that topic is chosen.

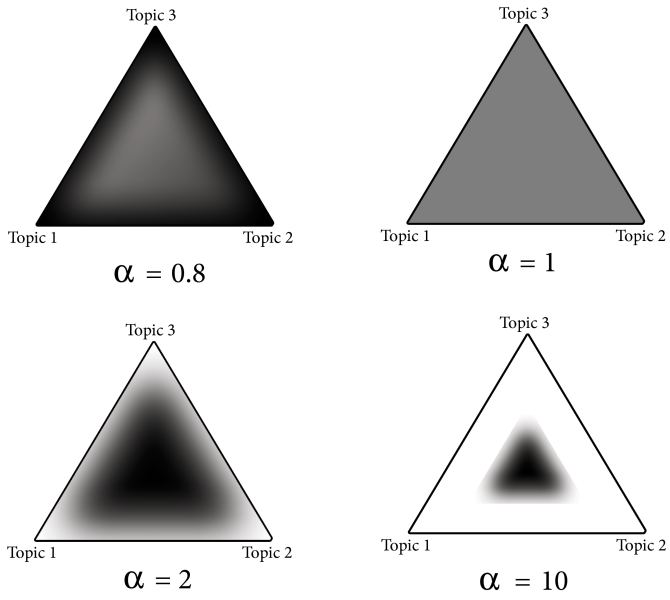


Figure 2.3: Symmetric Dirichlet distribution for three topics on two-dimensional simplex.

The topics are drawn from a Dirichlet distribution parameterized by α, which controls the width of the peak of the documents distributions over topics. High values of α lead to smooth distributions, while small values ($\alpha \leq 1$) mean that the modes of the Dirichlet distribution are in the corners of the simplex, resulting in sparse topic distributions. Figure 2.3 illustrates this phenomenon.

The reason behind the choice of the Dirichlet distribution is the fact that it is a convenient distribution on the simplex. Firstly, because the Dirichlet distribution by itself is a density over K positive numbers, so it can be used to draw parameters for a multinomial distribution. For instance, we can see the advantage of using the Dirichlet as a prior for the multinomial distribution, by multiplying $p(\theta^d|\alpha)$ and $p(z_n^d|\theta^d)$, to obtain $p(\theta^d|z_n^d, \alpha)$, as follows:

- θ is chosen from a Dirichlet distribution parameterized by α, so, according to the Dirichlet definition:

$$p(\theta^d|\alpha) = \frac{\Gamma(\sum_{j=1}^{K} \alpha_j)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} (\theta_i^d)^{(\alpha_i - 1)} \tag{2.6}$$

- z_n^d is chosen from a multinomial distribution parameterized by θ^d, hence, according to the multinomial definition:

$$p(z_n^d|\theta^d) = \prod_{i=1}^{K} \left(\theta_i^d\right)^{z_{n,i}^d} \tag{2.7}$$

Therefore:

$$\begin{aligned}
\overbrace{p(\theta^d|z_n^d, \alpha)}^{posterior} &\propto \overbrace{p(\theta^d|\alpha)}^{prior} \overbrace{p(z_n^d|\theta^d)}^{likelihood} \\
&= \frac{\Gamma(\sum_{j=1}^{K} \alpha_j)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} (\theta_i^d)^{(\alpha_i - 1)} \times \prod_{i=1}^{K} \left(\theta_i^d\right)^{z_{n,i}^d} \\
&= \frac{\Gamma(\sum_{j=1}^{K} \alpha_j)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} (\theta_i^d)^{n_k + (\alpha_i - 1)},
\end{aligned} \tag{2.8}$$

where n_k is the number of times that $z_{n,i}^d$ appeared, i.e., the number of times the topic i was associated to the word n in the document d. As it can be noticed, the posterior $p(\theta^d|z_n^d, \alpha)$ is also a Dirichlet distribution, given by a similar probability density function to the prior $p(\theta^d|\alpha)$.

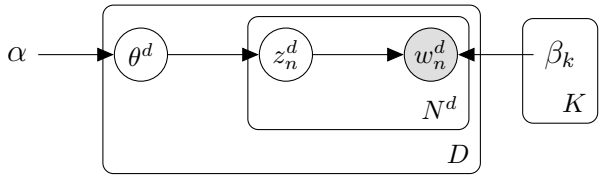


Figure 2.4: Graphical model of LDA.

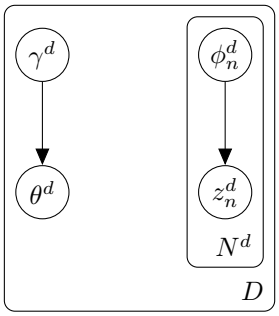


Figure 2.5: Graphical model of the variational distribution.

Secondly, because sufficient statistics, like n_k, can be applied to fully summarize the data. This is justified by the fact that both Dirichlet and multinomial distributions are from the exponential family. That is, the posterior $p(\theta^d|z_n^d, \alpha)$ can be calculated without knowing all the individual values of z_n^d. Instead, it can be written as dependent on the sufficient statistic of z_n^d, n_k.

Figure 2.4 shows the graphical model of LDA, where D is the number of documents, N^d is the number of words in the document d, θ^d is topic distribution of the d^{th} document, z_n^d is the word-topic assignments and w_n^d represents the word n of the document d. The model parameters (α and β_k) are represented following the notation used by Bishop et al. (2006). As it can be seen in the graphical model of Figure 2.4, the joint distribution is given by:

$$p(\boldsymbol{\theta}, \boldsymbol{z}_{1:D}, \boldsymbol{w}_{1:D}|\alpha, \boldsymbol{\beta} = \prod_{d=1}^{D} p(\theta^d|\alpha) \left(\prod_{n=1}^{N^d} p(z_n^d|\theta^d) p(w_n^d|z_n^d, \boldsymbol{\beta}) \right). \tag{2.9}$$

The posterior distribution over the latent variables $\theta_{1:D}$ and $\boldsymbol{z}_{1:D}$ is then given

by:

$$p(\boldsymbol{\theta}, \mathbf{z}_{1:D}|\mathbf{w}_{1:D}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \mathbf{w}_{1:D}|\alpha, \boldsymbol{\beta})}{p(\mathbf{w}_{1:D}|\alpha, \boldsymbol{\beta})}$$
$$= \frac{\prod_{d=1}^{D} p(\theta^d|\alpha)\left(\prod_{n=1}^{N_d} p(z_n^d|\theta^d)p(w_n^d|z_n^d, \boldsymbol{\beta})\right)}{\int_\theta^d \prod_d^D p(\theta^d|\alpha) \sum_{\mathbf{z}} \left(\prod_{n=1}^{N_d} p(z_n^d|\theta)p(w_n^d|z_n^d, \boldsymbol{\beta})\right)}, \quad (2.10)$$

which is intractable. If one considers expanding the denominator:

$$p(\mathbf{w}_{1:D}|\alpha, \boldsymbol{\beta}) = \frac{\Gamma(\sum_{j=1}^{K} \alpha_j)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \int_{\theta^d} \left(\prod_{i=1}^{k} (\theta^d)_i^{a_i - 1}\right) \left(\prod_{n=1}^{N} \sum_{i=1}^{k} \prod_{j=1}^{V} (\theta_i^d \beta_{i,j})^{(w_n^d)^j}\right), \quad (2.11)$$

it can be seen that there is a problematic coupling between θ and $\boldsymbol{\beta}$. Furthermore, θ is continuous, which makes the integral intractable to compute.

Hence, it is applied variational inference, whose goal is to select an approximation distribution from a tractable family of distributions and optimize its parameters, in order to make it as close as possible to the true posterior distribution. In LDA, this tractable family can be obtained by relaxing some constraints in the model. Since the edges that connect θ, $\boldsymbol{z}$ and $\boldsymbol{w}$ are the reason for the problem to be intractable, dropping them (as shown in Figure 2.5) results in a tractable family of distributions on the variables $\boldsymbol{z}$ and θ:

$$q(\theta, \boldsymbol{z}_{1:D}|\gamma, \phi) = \prod_{d=1}^{D} q(\theta^d|\gamma^d)\left(\prod_{n=1}^{N^d} q(z_n^d|\phi_n^d)\right), \quad (2.12)$$

where γ and ϕ are the variational parameters.

The next step is to define the variational objective function to minimize the Kullback-Leibler divergence between the true posterior $p(\boldsymbol{\theta}, \mathbf{z}_{1:D}|\mathbf{w}_{1:D})$ and the approximation $q(\boldsymbol{\theta}, \boldsymbol{z}_{1:D}|\gamma, \phi)$ (see Bishop (1998)). This approximate posterior distribution allows the estimation of the topics $\boldsymbol{\beta}$ and the Dirichlet prior α, using the variational Bayes Expectation-maximization algorithm (Bernardo et al., 2003), which is a Bayesian approach to the regular Expectation-maximization algorithm (Dempster et al., 1977).

Expectation-maximization (EM) consists in two steps that are iteratively alternated: in the first step (E-step) the latent variables are inferred given the current parameters, while in the second (M-step) the goal is to

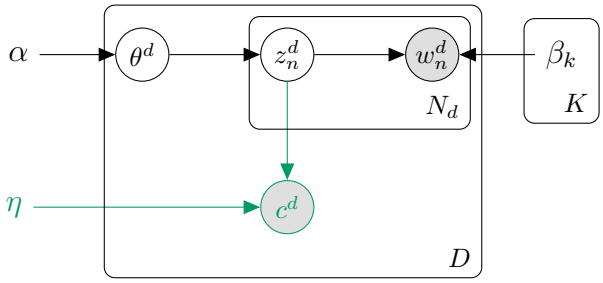


Figure 2.6: Graphical model of sLDA.

find the single best value for each parameter, given the current posterior over the latent variables. The LDA parameters are α and $\boldsymbol{\beta}$, whose best values are the ones that maximize the log likelihood computed on E-step. This maximization relies on finding the approximate maximum likelihood estimates of α and $\boldsymbol{\beta}$ using the expected sufficient statistics computed in the E-step. In other words, the parameter estimation works with the statistics calculated on the current distributions of the latent variables. The resulting parameter-estimates are, then, used in the next E-step.

The difference between VBEM and EM is that, in the E-step of VBEM, variational Bayesian inference is applied.

2.3.2 Supervised topic models

Supervised topic models are a family of supervised learning methods built on top of LDA. Since documents are frequently associated with other variables such as labels, tags or ratings, the main purpose of supervised topic modeling is to take advantage of this extra information to guide the topics discovery process. In contrast to procedures that apply the documents topic proportions obtained by LDA as features to train a classifier in an isolated fashion, supervised topic models take into account the label of the document in the topics generation. Moreover, notice that the topics are meant to disclose the themes of the text. Supposing that documents are descriptions of products and their labels are scores of users reviews, for instance, we can not expect that the theme of the product explains its score.

Supervised LDA (Blei & McAuliffe, 2007) was one of the first LDA extensions made for the purpose of treating labeled data. The difference between supervised LDA (sLDA) and LDA can be perceived by comparing Figure 2.4 and 2.6: a response variable c^d associated with each document d is added, along with a set of coefficients $\boldsymbol{\eta}$ to parameterize this relationship, i.e.,

the nodes and edges represented in green are responsible for the supervised learning in the model. In order to obtain the optimal response variables, sLDA jointly models the documents and the responses, hence, the extracted topics have a fundamental role in the prediction process. The generative process of the sLDA version to classification problems is the following:

1. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$
   (b) For the n^{th} word:
      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta}_{z_n^d} \sim Multinomial(\beta_{z_n})$
   (c) Draw class $c^d|\boldsymbol{z}^d, \boldsymbol{\eta} \sim Softmax(\bar{z}^d, \boldsymbol{\eta})$ where $\bar{z}^d = \frac{1}{N_d}\sum_{n=1}^{N_d} z_n^d$ and

$$p(c^d|\bar{z}^d, \boldsymbol{\eta}) = \frac{\exp(\eta_c^T \bar{z}^d)}{\sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d)}. \tag{2.13}$$

The posterior distribution over the latent variables $\boldsymbol{\theta}_{1:D}$ and $\boldsymbol{z}_{1:D}$ is then given by:

$$p(\boldsymbol{\theta}, \mathbf{z}_{1:D}|\mathbf{w}_{1:D}, \mathbf{c}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \mathbf{w}_{1:D}|\alpha, \boldsymbol{\beta})}{p(\mathbf{w}_{1:D}|\alpha, \boldsymbol{\beta})} \tag{2.14}$$

$$= \frac{\prod_{d=1}^{D} p(\theta^d|\alpha)\left(\prod_{n=1}^{N_d} p(z_n^d|\theta^d)p(w_n^d|z_n^d, \boldsymbol{\beta})\right)p(c|\boldsymbol{z}^d, \boldsymbol{\eta})}{\int_\theta^d \prod_d^D p(\theta^d|\alpha)\sum_{\mathbf{z}}\left(\prod_{n=1}^{N_d} p(z_n^d|\theta)p(w_n^d|z_n^d, \boldsymbol{\beta})\right)\sum_c p(c^d|\boldsymbol{z}^d, \boldsymbol{\eta})} \tag{2.15}$$

which, just like the LDA posterior, is intractable. Therefore, it is approximated by the variational distribution of the latent variables given by 2.12.

Since there are only a few differences between LDA and sLDA models, the variational Bayesian Expectation-Maximization algorithm is similar:

- E-step: For each document, optimize the variational parameters γ and ϕ.
- M-step: Find the maximum likelihood estimates for the topics $\boldsymbol{\beta}$ and the class coefficients $\boldsymbol{\eta}$. The Dirichlet parameter α can also be estimated, but, in practice, it is common to fix its value.

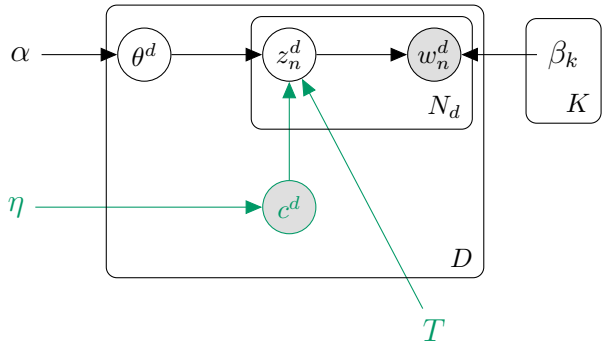


Figure 2.7: Graphical model of DiscLDA.

Supervised LDA was originally developed for predicting continuous response values, through a linear regression. The difference between it and the multi-class version previously presented is in the Step 1c of sLDA's generative process. In this step, the classification version of sLDA draws the classes from a multinomial distribution, whose parameters are obtained by a softmax function. The softmax function is a generalization of the logistic function that transforms a K-dimensional vector of arbitrary real values to a K-dimensional vector of real values in the range (0, 1). The regression variant, in turn, samples the target value x^d from a Gaussian distribution:

$$x^d | \boldsymbol{z}_{1:N}, \boldsymbol{\eta}, \sigma^2 \sim Normal(\boldsymbol{\eta}^T \bar{z}, \sigma^2), \tag{2.16}$$

where, here, $\boldsymbol{\eta}$ are the regression coefficients on the empirical frequencies of the topics in the d^{th} document $\boldsymbol{z}^d$.

Figure 2.7 shows the graphical model of another supervised topic model: Discriminative LDA (Lacoste-julien et al., 2009). As its name suggests Discriminative LDA (DiscLDA) is a discriminative probabilistic graphical model, unlike LDA and sLDA that are generative models. Nevertheless, both sLDA and DiscLDA assume that a label is generated from each document empirical topic mixture distribution. However, they have different learning methods: sLDA is trained by maximizing the joint likelihood of the data and response variables, while DiscLDA tries to maximize the conditional likelihood of the response variables. Furthermore, DiscLDA associates an additional class-label-dependent linear transformation (T) parameter with each document. Again, the bottom part of Figure 2.7, colored in green, is the difference between LDA and DiscLDA, that is, the components of the supervised learning distinguish the two models.

Another important supervised topic model is Maximum Entropy Dis-

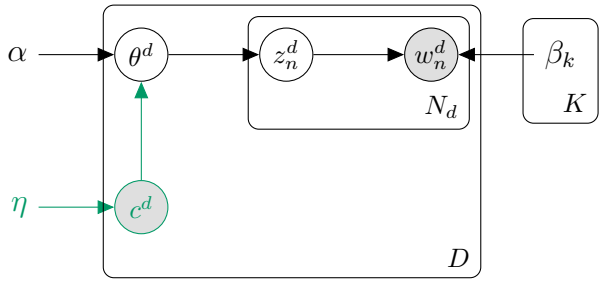


Figure 2.8: Graphical model L-LDA.

crimination LDA (Zhu et al., 2009). Maximum Entropy Discrimination LDA (MedLDA) is not a fully generative model, like sLDA, since it is trained using the max-margin principle. While the M-step is similar between the two, in the E-step of MedLDA, the posterior distribution of the latent variables is inferred by optimizing a constrained objective function.

MedLDA trains by looking for the topics that enable the maximum possible margin, which means that MedLDA is a combination of max-margin learning, which is well known for its use in support vector machines, and Bayesian topic models. Although it could have some advantages over sLDA, it is not as suitable as sLDA for the development of the multi-annotator generalization we propose. The extension introduced in this thesis requires a completely probabilistic model that returns the response variables in the form of distributions over labels, which makes the MedLDA single value output a hindrance.

In Figure 2.8 it is represented the graphical model of the Labeled LDA (Ramage et al., 2009), whose main improvement over the previous topic models is the capability to learn over multi-label data. Comparing the graphical models of sLDA and Labeled LDA (L-LDA), the most noticeable difference is, in green, the connection of the response variable: in sLDA the per-word topic assignment is linked to the response variable, while L-LDA makes the responses dependent directly on the per-document topic proportions. The reason why sLDA generates the response after z_n^d, instead of being drawn by the documents distribution over topics θ^d, is to use the topic frequencies that truly occurred in the document. It can be shown that the word-topics assignments z_n^d is more suited to prediction than θ^d, which is a mean distribution exchangeable with the words (Blei & McAuliffe, 2007). Nevertheless, the response variable c^d has a different purpose in L-LDA: to restrict the topics of a document to that document's label set. This means that each label is matched with a topic.

Multi-Modal LDA (Putthividhy et al., 2010), Prior-LDA and Dependency-LDA (Rubin et al., 2012) are further examples of supervised topic models for multi-label classification that will not be described here for not being relevant to the purpose of this work.

2.4 Learning from the crowds

The problem of learning from multiple annotators has been studied for a long time. In 1979, Dawid & Skene (1979) proposed an Expectation-Maximization (Dempster et al., 1977) approach to estimate the error rates of the responses of patients (labelers) to various medical questions, based on the true symptoms of the disease. The resultant model was also used by Spiegelhalter & Stovin (1983) to quantify the residual uncertainty of labels related to rejections in cardiac transplantations. Both of these works considered label error estimation and building a classifier as two separate processes.

However, there is now a special interest in designing classifiers from multi-annotator data. A common approach to this problem relies on repeated labeling, whose goal is to identify which labels should be reacquired so that the classification performance or data quality is enhanced. This implies having the same set of annotators labeling the same set of instances. Smyth et al. (1995), Sheng et al. (2008) and Donmez & Carbonell (2008) used this approach, since it is relatively cheap to obtain labels in the tasks they were focusing on. Nevertheless, as stated in Dekel & Shamir (2009), repeated labeling may wastefully decrease the size of the training set. Moreover, in Sheng et al. (2008), it was assumed that all annotators had equivalent reliability, thus, majority voting was used to infer the ground truth. Such an assumption may be reasonable in homogeneous environments, however, environments such as Amazon Mechanical Turk [1] are highly heterogeneous and the quality of the annotators can vary significantly (Rodrigues et al., 2013).

The approach presented in this thesis does not depend on repeated labeling. Contrarily, it gives the possibility of having only one annotator answer per data point. Another advantage is that the annotators error rates, the true label and the classifier are jointly estimated by a latent variable model using the variational Bayesian Expectation-Maximization algorithm, in a way that resembles the EM method presented by Raykar et al. (2009).

In Figure 2.9, the graphical model proposed by Raykar et al. (2009) is illustrated, in which the nodes and edges drawn in dark red are the elements

[1] http://www.mturk.com

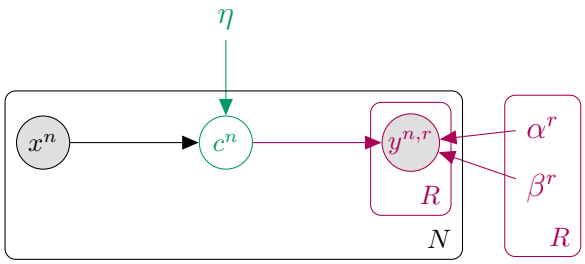


Figure 2.9: Graphical model of Raykar et al. (2009)'s framework.

related to the multi-annotator learning and the green ones are, like in the previous section, associated with the supervised training [2]. Therefore, the $y^{n,r}$ node represents the label assigned by the annotator r to instance x^n and c^n its true (unobserved) label. Assuming a binary classification scenario and that each annotator provides a noisy version of the hidden true label, the sensitivity of the r^{th} annotator (α^r) and the specificity (β^r)[3] are given by:

$$\alpha^r = p(y^r = 1|c = 1) \tag{2.17}$$
$$\beta^r = p(y^r = 0|c = 0) \tag{2.18}$$

It is also assumed that the N instances are independently sampled, that α^r and β^r do not depend on the instance x^n and that all the R annotators make their decisions independently, thus, the generative process of the Raykar et al. (2009)'s model is the following:

1. For each instance x^n
   (a) Draw latent (true) class $c^n|x^n, \boldsymbol{\eta} \sim Softmax(\bar{\boldsymbol{x}}^n, \boldsymbol{\eta})$
   (b) For the r^{th} annotator
      i. If $c^n = 1$: Draw annotator's answer $y^r \sim Bernoulli(\alpha^r)$
      ii. If $c^n = 0$: Draw annotator's answer $y^r \sim Bernoulli(\beta^r)$

Therefore, by reading the graphical model of Figure 2.9, the likelihood is written as:

$$p(\mathcal{D}|\theta) = \prod_{n=1}^{N} \left(p(c^n|x^n, \boldsymbol{\eta}) \prod_{r=1}^{R} p(y^{n,r}|c^n, \alpha^r, \beta^r) \right), \tag{2.19}$$

[2]This color notation will be used throughout the whole thesis.

[3]These parameters α^r and β^r should not be confused with the Dirichlet parameter α and with the per-topic word proportions parameter $\boldsymbol{\beta}$ of the previous sections.

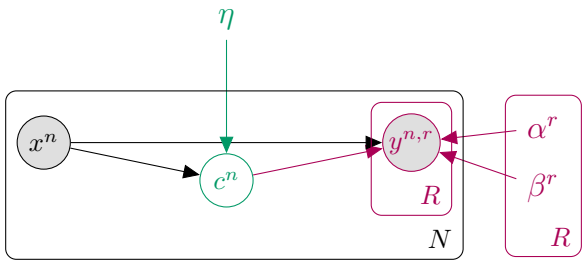


Figure 2.10: Graphical model of Yan et al. (2010)'s framework.

where $\theta = \{\boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\beta}\}$ and $\mathcal{D} = \{y^{n,1}, ..., y^{n,R}, x^n\}_{n=1}^{N}$.

Then, the maximum-likelihood estimator is found by maximizing the log-likelihood:

$$\hat{\theta}_{ML} = \arg\max_{\theta}\{\ln p(\mathcal{D}|\theta)\} \tag{2.20}$$

Since this maximization problem is intractable to compute, the EM algorithm is used to find a maximum likelihood estimate.

Yan et al. (2010) later proposed a model that does not rely on the assumption that labelers reliability is consistent across all the input data. Contrarily to the Raykar et al. (2009)'s model, it is taken into account that some annotators are better at labeling certain types of data points, which means that the annotation y^r of the r^{th} annotator depends on the (unobserved) true label c and, also, on the instance x. That is, in the probabilistic graphical model, there is an edge connecting the node x to the node representing the label provided by the annotator r to instance x, like it is exhibited in Figure 2.10.

Nevertheless, in sequence labeling problems, such as part-of-speech tagging (a task that consists in assigning a part-of-speech to each word in an input sentence or document) there is a connection between the data points (e.g words), i.e., they are not independent. This represents a real challenge for the previously presented approaches, since the unobserved ground truth labels, which are now sequences, are explicitly treated as latent variables. Considering that these latent variables must be marginalized out for this sort of problems, this means marginalizing over a potentially huge number of label sequences. Regarding this kind of problems, Dredze et al. (2009) presented a method for sequence learning from data with multiple labels in the presence of noise. Besides the fact that this model is multi-label and not multi-annotator, the experimental results obtained by it evince that the

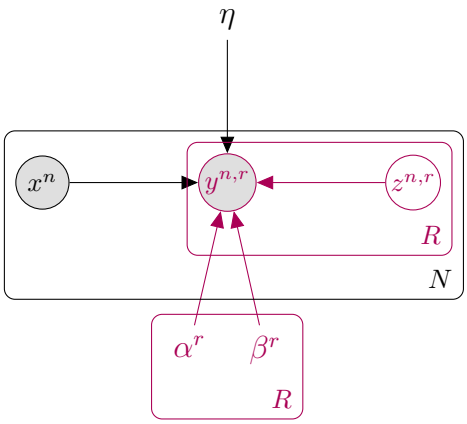


Figure 2.11: Graphical model of Rodrigues et al. (2013)'s framework.

method is only appropriate for scenarios where the amount of training data is low and when the labels are noisy.

Recently, Rodrigues et al. (2013) presented a multi-annotator solution to the same problem, in which the annotators reliabilities are treated as latent variables, so it bypasses the problem of the high number of possible labellings to marginalize over. Furthermore, focusing on the annotators and including the unknown reliabilities of the annotators as latent variables leads to simpler models that are less prone to overfitting.

Indeed, this approach was empirically shown to outperform approaches in which majority voting or labeled data from all the annotators concatenated is used. Its graphical model is shown in Figure 2.11, where $y^{n,r}$ is the label assigned by the annotator r to instance x^n and $z^{n,r}$ indicates whether the r^{th} annotator labeled the instance x^n correctly or not.

Despite the fact that the approach proposed in this thesis considers true labels as latent variables (similarly to some of the presented methods), it differs by the fact that it is incorporated in a supervised topic model, so that both topics and true labels are jointly estimated.

2.5 Travel demand modeling for special events

Travel demand modeling (TDM) for special events has been recognized as highly relevant to prevent congestion, overcrowding and delay (Lei-Lei et al., 2012; Jingbo et al., 2009; Born et al., 2014; Sall & Bhat, 2007), and, thus, it has received some attention during the past few years. The major focus has been on large and mega events, such as the Olympic games, large

concerts or big soccer matches. These are undoubtedly disruptive and they are often given their own transport planning resources. On the other hand, smaller ones are subtle to understand even though they can also raise serious problems, particularly when happening simultaneously (Pereira et al., 2015).

An interesting aspect is that the travel patterns due to such events have a quite typical behavior, with two subsequent waves of demand (Lei-Lei et al., 2012). The first one is caused by going to the event, while the second one is when leaving. Taking advantage of this fact, in Kwoczek et al. (2014), the data from the first wave was used to predict the second one. In this work, a K-Nearest Neighbors (K-NN) technique was applied to find the most similar past events, in terms of traffic pattern (as measured by GPS probes) as well as event category (concert, entertainment, sports, comedy), in order to derive a prediction of the impact of the second wave of traffic. The authors used a database of twenty nine events hosted in the Cologne LANXESS arena from June to December 2013 in the inner city of Cologne, Germany. They showed that the category alone shows high differences in observed average traffic delays (e.g. comedy events incurred in much less delay than concerts). This comes from a combination of different attendance size and travel mode shares.

While Kwoczel's approach is relevant for traffic operations monitoring and decision making, TDM can benefit much more from earlier stage predictions, weeks or months in advance to allow for planning. We can group methodologies into two general classes: one is the discrete choice framework (Ben-Akiva & Lerman, 1987), where we represent individual behavior choices (e.g. transport mode, departure time, path choice) through logistic or probit regression models, as a function of individual's characteristics (e.g. age, gender) and alternative choice properties (e.g. cost, duration), another is machine learning, where we model an aggregate response variable (e.g. travel delay, total attendance) as a function of available data (e.g. location, time of day, event category).

For their discrete choice model, Shahin et al. (2014) analysed surveys conducted at three Turkish stadiums in advance and after matches, to estimate a binary logit model of mode choice (private car or public transport). Their decision models consider individual characteristics (age, gender, income and ownership of season ticket) and trip characteristics (trip cost, travel time). To apply this type of model in TDM, there are two major challenges: assumptions on population distribution, namely the share of participants coming from each geographical area and the total expected attendance, and generation of (unobserved) alternatives. In the discrete choice framework, each decision is simulated and the agent decides among a set of alternatives,

only one of which is observable from the data. In Shahin et al. (2014), the authors only estimated the individual choice model, leaving the actual demand prediction open. Also, they did not ask in the survey for considered alternatives, so they generated them from all set of possible modes.

Also based on the discrete choice framework, in Jingbo et al. (2009), parking lot occupancy is estimated in a case study from 2008 Olympic games, by predicting mode choice from (given) attendance totals, together with estimates of distance and cost for each individual trip. A more general approach was followed by Born et al. (2014), focusing on weekend discretionary event type participation, duration of participation and accompaniment type jointly in a simultaneous equations model system. The authors grouped events into four categories (social/recreational activity, visit friends/family, go out/hang out and visit public place). A joint discrete-continuous modeling framework was, then, formulated for analyzing these dimensions as a choice bundle. The data used comes from the 2008-2009 National Household Travel Survey (NHTS) conducted in the United States.

Two other recent examples of this approach include Kuppam et al. (2011) and Chang & Lu (2013), who proposed a four-step model approach, where they predict, for each event, the number of trips by type, trip time-of-day, trip origins/destinations (OD), mode and vehicle miles travelled/transit boardings generated due to the events. The data was obtained through questionnaire surveys at the venue gates, that, besides social-demographics questions, also inquire about transportation choices (e.g. mode, costs, type of vehicle, origin, etc.). This data was, then, used for calibration of utility maximization choice models through a maximum likelihood approach.

Although these works seem behaviourally sound and provide plenty of detail, they are highly dependent on survey response and, in fact, consider event characteristics on a very superficial way. For example, they rarely go deeper than general event category (e.g. sports, concerts). Finally, these models treat each event individually and ignore interactions with other simultaneous events or even routine trip behaviour, with the exception of Born et al. (2014).

On the machine learning realm, some research has been done to analyse and predict travel demand long time in advance. Using a neural network, Calabrese et al. (2010) showed the high correlation between event category and public home area distribution, as observable by a large telecommunications dataset from the city of Boston. Also using a neural network model, Pereira et al. (2013b) used three weeks of smartcard data from Singapore to predict half-hourly demand in five different locations. In this work, the events and their categories were extracted from the internet and popularity

features, like number of hits in Google and Facebook likes, were built. However, these features did not show relevant predictive power, possibly due to excessive noise (e.g. search not being specific enough). In a follow up work (Pereira et al., 2015), it is presented an approach that accounts for multiple simultaneous events and a Bayesian additive linear model is developed, which breaks down observed attendance into individual event contributions, each event being defined by the topics extracted from event descriptions, as well as spatio-temporal data. The same model is, then, used to predict future demand.

As these models evolved, the data about events has become increasingly rich and the processes to get them more complex. For this thesis, we take advantage of web search to simplify this process, yet, maintaining high semantic richness. In this way, moving the model between cities, or even considering other domains (e.g. stock market), becomes a simple matter.

Chapter 3

Multi-annotator supervised latent Dirichlet allocation for classification

In this chapter, the multi-class supervised topic model with multiple annotators proposed in this thesis is presented. The chapter starts with a brief description of the model, which is followed by two sections that show both the approximation of the posterior distribution over the model's latent variables and the estimation of the model parameters. We then demonstrate how stochastic variational inference was applied to the proposed model. As we mention in Section 2.2, stochastic variational inference is more scalable and, therefore, faster than the regular variational inference algorithm, especially when dealing with large data sets. Finally, we show how to use the learned model to classify new instances.

3.1 Proposed model

The proposed model was built in C++ on top of the multi-class supervised latent Dirichlet allocation (Blei & McAuliffe, 2007) model. It is able to learn from crowds, by taking into account multiple answers given by distinct annotators. Therefore, considering an annotated dataset $\mathcal{D} = \{\boldsymbol{w}^d, \boldsymbol{y}^d\}_{d=1}^D$ of size D, in which a set of annotations $\boldsymbol{y}^d = \{y_r^d\}_{r=1}^R$ given by R different labelers is assigned to each document $\boldsymbol{w}^d$, the multiple-annotator supervised latent Dirichlet allocation model (MA-sLDA) estimates both the topics $\boldsymbol{\beta}$ and the true classes $\boldsymbol{c} = \{c^d\}_{d=1}^D$. This is achieved by treating the documents as arising from a set of latent topics, each topic being defined as a distribution

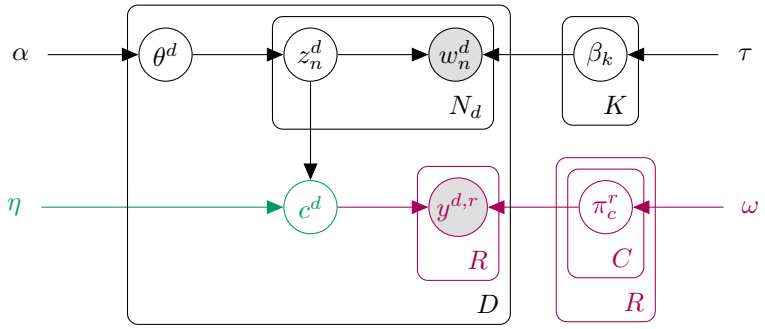


Figure 3.1: Graphical model of MA-sLDA for classification.

over the words in the vocabulary, and associate each one of the documents with a latent class c^d. Each class c^d is, in turn, related to a set of annotations $\boldsymbol{y}^d = \{y_r^d\}_{r=1}^R$ that are modeled assuming that, given a true class c^d, each annotator r provides the label l with some probability $\pi_{c,l}^r$. This means that MA-sLDA generalizes sLDA by observing noisy labels given by multiple labelers and estimating the true classes, while modeling the annotators' different levels of expertise and correcting their potential biases.

The multi-class MA-sLDA graphical model is exhibited in Figure 3.1, where D is the size of the corpus, R is the number of annotators, C denotes the number of classes, K is the number of topics and N^d is the number of words in the d^{th} document. As it can be seen, each word w_n^d in a document d is provided a discrete topic-assignment z_n^d, which is drawn from the documents distribution over topics θ^d and a topic distribution $\boldsymbol{\beta}$, just like in LDA. Moreover, there is a class c^d generated from the mean topic assignment of the document $\bar{z}^d$ and by coefficients $\boldsymbol{\eta}$, as in sLDA. This class c^d and the per-annotator confusion matrix $\boldsymbol{\pi^r}$ are assumed to give origin to each annotator's label, which distinguishes MA-sLDA from the previously mentioned approaches.

The generative process under the MA-sLDA model can be summarized as follows:

1. For each annotator r
   (a) For each class c
      i. Draw annotator reliability parameter $\pi_c^r|\omega \sim Dirichlet(\omega)$
2. For each topic k
   (a) Draw topic distribuition $\beta_k|\tau \sim Dirichlet(\tau)$

3. For each document d

   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$

   (b) For the n^{th} word

      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$

      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Multinomial(\beta_{z_n^d})$

   (c) Draw latent (true) class $c^d|\mathbf{z}^d, \eta \sim Softmax(\bar{z}^d, \eta)$ where $\bar{z}^d = \frac{1}{N_d}\sum_{n=1}^{N_d} z_n^d$ and

$$p(c^d|\bar{z}^d, \eta) = \frac{\exp(\eta_c^T \bar{z}^d)}{\sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d)}. \tag{3.1}$$

   (d) For each annotator r

      i. Draw annotator's answer $y^{d,r}|c^d, \boldsymbol{\pi}^r \sim Mult(\pi_{c^d}^r)$

According to their definitions, the distributions are given by:

$$p(\theta^d|\alpha)] = \frac{\Gamma(\sum_{j=1}^{K}\alpha_j)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}(\theta_i^d)^{(\alpha_i-1)} \tag{3.2}$$

$$p(z_n^d|\theta^d) = \prod_{i=1}^{K}\left(\theta_i^d\right)^{z_{n,i}^d} \tag{3.3}$$

$$p(w_n^d|z_n^d, \boldsymbol{\beta}) = \prod_{j=1}^{V}(\beta_{z_n^d,j})^{w_{n,j}^d} \tag{3.4}$$

$$p(y^{d,r}|c^d, \boldsymbol{\pi}^r) = \prod_{l=1}^{C}(\pi_{c^d,l}^r)^{y_l^{d,r}} \tag{3.5}$$

$$p(\beta_i|\tau) = \frac{\Gamma(\sum_{k=1}^{V}\tau_k)}{\prod_{j=1}^{V}\Gamma(\tau_j)}\prod_{j=1}^{V}(\beta_{i,j})^{(\tau_j-1)} \tag{3.6}$$

$$p(\pi_c^r|\omega) = \frac{\Gamma(\sum_{t=1}^{C}\omega_t)}{\prod_{l=1}^{C}\Gamma(\omega_l)}\prod_{l=1}^{C}\left(\pi_{c,l}^r\right)^{(\omega_l-1)} \tag{3.7}$$

where V is the size of the vocabulary.

We also developed a simpler version of the MA-sLDA, where the parameters are obtained through maximum likelihood estimation. This model is described in Appendix C. In this chapter, however, it is introduced a fully Bayesian approach for estimating the reliabilities and biases of the different

annotators $\boldsymbol{\pi}$ as well as the topic distributions $\boldsymbol{\beta}$. This means that, instead of finding just the optimal values for these parameters, such that the likelihood is maximized, it is performed variational Bayesian inference to produce smooth posteriors. Thus, we have now three Dirichlet parameters: τ on $\boldsymbol{\beta}$, ω on $\boldsymbol{\pi}$ and α. All of these priors are assumed to be fix-valued. In a practical manner, the new parameters τ and ω set the sparsity of $\boldsymbol{\beta}$'s and $\boldsymbol{\pi}$'s distributions, respectively. Small values lead to sparse multinomial distributions, while high values result in smooth distributions, as it is explained in Section 2.3.1.

3.2 Approximate inference

The goal of the generative probabilistic model of MA-sLDA is to estimate latent variables and the model's parameters from the observed data. Therefore, given a dataset $\mathcal{D}$, it is defined the joint distribution of the model's variables and, consequently, it is formulated the posterior distribution over the latent variables $\boldsymbol{\theta}, \mathbf{z}_{1:D}$ and $\boldsymbol{c}$. Nevertheless, as it can be perceived by defining the model parameters $\Theta = \{\alpha, \eta, \tau, \omega\}$ and deriving the joint distribution:

$$p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta) \tag{3.8}$$

$$= \left(\prod_{i=1}^{K} p(\beta_i|\tau)\right)\left(\prod_{r=1}^{R}\prod_{c=1}^{C} p(\pi_c^r|\omega)\right)\prod_{d=1}^{D} p(\theta^d|\alpha)\left(\prod_{n=1}^{N_d} p(z_n^d|\theta^d)p(w_n^d|z_n^d, \boldsymbol{\beta})\right)$$

$$\times\, p(c^d|\boldsymbol{z}^d, \eta)\prod_{r=1}^{R} p(y^{d,r}|c^d, \boldsymbol{\pi}^r), \tag{3.9}$$

to obtain the posterior:

$$p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \mathbf{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\mathbf{w}_{1:D}, \mathbf{y}_{1:D}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta)}{p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta)} \tag{3.10}$$

$$= \frac{\left(\prod_{i=1}^{K} p(\beta_i|\tau)\right)\left(\prod_{r=1}^{R}\prod_{c=1}^{C} p(\pi_c^r|\omega)\right)}{\int_{\beta_i}\left(\prod_{i=1}^{K} p(\beta_i|\tau)\right)\int_{\pi_c^r}\left(\prod_{r=1}^{R}\prod_{c=1}^{C} p(\pi_c^r|\omega)\right)}$$

$$\times \frac{\prod_{d=1}^{D} p(\theta^d|\alpha)\left(\prod_{n=1}^{N_d} p(z_n^d|\theta^d)p(w_n^d|z_n^d,\boldsymbol{\beta})\right)p(c^d|\boldsymbol{z}^d,\eta)\prod_{r=1}^{R} p(y^{d,r}|c^d,\boldsymbol{\pi}^r)}{\int_\theta^d \prod_d^D p(\theta^d|\alpha)\sum_{\mathbf{z}}\left(\prod_{n=1}^{N_d} p(z_n^d|\theta)p(w_n^d|z_n^d,\boldsymbol{\beta})\right)\sum_c p(c^d|\mathbf{z}^d,\eta)\prod_{r=1}^{R} p(y^{d,r}|c^d,\boldsymbol{\pi}^r)}, \tag{3.11}$$

exact inference is computationally intractable, for the same reasons of LDA. Thus, it is applied variational Bayesian inference to approximate the posterior distribution.

Let $q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})$ denote a variational distribution of the latent variables. Since we are using a fully-factorized (mean-field) approximation, we have that:

$$q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}) = \left(\prod_{i=1}^{K} q(\beta_i|\zeta_i)\right)\left(\prod_{r=1}^{R}\prod_{c=1}^{C} q(\pi_c^r|\xi_c^r)\right)\prod_{d=1}^{D} q(\theta^d|\gamma^d)\left(\prod_{n=1}^{N_d} q(z_n^d|\phi_n^d)\right)$$
$$\times\, q(c^d|\lambda^d), \tag{3.12}$$

where $\boldsymbol{\gamma}, \boldsymbol{\phi}_{1:D}, \boldsymbol{\lambda}, \boldsymbol{\zeta}, \boldsymbol{\xi}_{1:R}$ are the variational parameters.

The variational objective function (or the evidence lower bound or ELBO, as explained in Section 2.2) is, then, given by:

$$\log p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\alpha, \eta, \tau, \omega)$$
$$= \log \int_\pi \int_\beta \int_\theta \sum_{\mathbf{z}} \sum_c \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta)q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})}{q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})} \tag{3.13}$$

$$\geqslant \mathcal{L}(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta)$$
$$= \mathbb{E}_q[\log p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta)] \underbrace{-\mathbb{E}_q[\log q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})]}_{H(q)} \tag{3.14}$$

$$= \sum_{i=1}^{K} \mathbb{E}_q[\log p(\beta_i|\tau)] + \sum_{r=1}^{R}\sum_{c=1}^{C} \mathbb{E}_q[\log p(\pi_c^r|\omega)] + \sum_{d=1}^{D}\Bigg(\mathbb{E}_q[\log p(\theta^d|\alpha)]$$
$$+ \sum_{n=1}^{N_d} \mathbb{E}_q[\log p(z_n^d|\theta^d)] + \sum_{n=1}^{N_d} \mathbb{E}_q[\log p(w_n^d|z_n^d, \boldsymbol{\beta})] + \mathbb{E}_q[\log p(c^d|\bar{z}^d, \eta)]$$

$$+ \sum_{r=1}^{R} \mathbb{E}_q[\log p(y^{d,r}|c^d, \boldsymbol{\pi}^r)] \Bigg) + H(q), \tag{3.15}$$

where the entropy $H(q)$ of the variational distribution is defined as:

$$H(q) = -\sum_{r=1}^{R}\sum_{c=1}^{C} \mathbb{E}_q[\log q(\pi_c^r|\xi_c^r)] - \sum_{i=1}^{K} \mathbb{E}_q[\log q(\beta_i|\zeta_i)]$$
$$- \sum_{d=1}^{D} \Bigg(\mathbb{E}_q[\log q(\theta^d|\gamma^d)] - \sum_{n=1}^{N_d} \mathbb{E}_q[\log q(z_n^d|\phi_n^d)] - \mathbb{E}_q[\log q(c^d|\lambda^d)] \Bigg). \tag{3.16}$$

We will now analyse and calculate each term of the lower bound individually.

$$\begin{aligned}\mathbb{E}_q[\log p(\beta_i|\tau)] &= \mathbb{E}_q\Bigg[\log \frac{\Gamma(\sum_{k=1}^{V} \tau_k)}{\prod_{j=1}^{V} \Gamma(\tau_j)} \prod_{j=1}^{V} \beta_{i,j}^{(\tau_j - 1)} \Bigg] \\ &= \log \Gamma\Bigg(\sum_{k=1}^{V} \tau_k \Bigg) - \sum_{j=1}^{V} \log \Gamma(\tau_j) + \sum_{j=1}^{V} (\tau_j - 1)\mathbb{E}_q[\log \beta_{i,j}]\end{aligned} \tag{3.17}$$

$$\begin{aligned}\mathbb{E}_q[\log p(\pi_c^r|\omega)] &= \mathbb{E}_q\Bigg[\log \frac{\Gamma(\sum_{t=1}^{C} \omega_t)}{\prod_{l=1}^{C} \Gamma(\omega_l)} \prod_{l=1}^{C} \left(\pi_{c,l}^r\right)^{(\omega_l - 1)} \Bigg] \\ &= \log \Gamma\Bigg(\sum_{t=1}^{C} \omega_t \Bigg) - \sum_{l=1}^{C} \log \Gamma(\omega_l) + \sum_{l=1}^{C} (\omega_l - 1)\mathbb{E}_q[\log \pi_{c,l}^r]\end{aligned} \tag{3.18}$$

$$\begin{aligned}\mathbb{E}_q[\log p(\theta^d|\alpha)] &= \mathbb{E}_q\Bigg[\log \frac{\Gamma(\sum_{j=1}^{K} \alpha_j)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{i=1}^{K} (\theta_i^d)^{(\alpha_i - 1)} \Bigg] \\ &= \sum_{j=1}^{V}\sum_{i=1}^{K} w_{n,j}^d \phi_{n,i}^d \mathbb{E}_q[\log \beta_{i,j}]\end{aligned} \tag{3.19}$$

$$\mathbb{E}_q[\log p(y^{d,r}|c^d, \boldsymbol{\pi}^r)] = \mathbb{E}_q\Bigg[\log \prod_{l=1}^{C} (\pi_{c^d,l}^r)^{y_l^{d,r}} \Bigg] = \sum_{c=1}^{C}\sum_{l=1}^{C} \lambda_c^d y_l^{d,r} \mathbb{E}_q[\log \pi_{c,l}^r]. \tag{3.20}$$

Similarly, the corresponding terms of the variational distribution are derived as:

$$\mathbb{E}_q[\log q(\pi_c^r|\xi_c^r)] = \log\Gamma\bigg(\sum_{t=1}^{C}\xi_{c,t}^r\bigg) - \sum_{l=1}^{C}\log\Gamma(\xi_{c,l}^r) + \sum_{l=1}^{C}(\xi_{c,l}^r - 1)\mathbb{E}_q[\log\pi_{c,l}^r] \tag{3.21}$$

$$\mathbb{E}_q[\log q(\beta_i|\zeta_i)] = \log\Gamma\bigg(\sum_{k=1}^{V}\zeta_{i,k}\bigg) - \sum_{j=1}^{V}\log\Gamma(\zeta_{i,j}) + \sum_{j=1}^{V}(\zeta_{i,j} - 1)\mathbb{E}_q[\log\beta_{i,j}] \tag{3.22}$$

$$\mathbb{E}_q[\log q(\theta^d|\gamma^d)] = \log\Gamma\bigg(\sum_{j=1}^{K}\gamma_j^d\bigg) - \sum_{i=1}^{K}\log\Gamma(\gamma_i^d) + \sum_{i=1}^{K}(\gamma_i^d - 1)\mathbb{E}_q[\log\theta_i^d] \tag{3.23}$$

$$\mathbb{E}_q[\log q(z_n^d|\phi_n^d)] = \sum_{i=1}^{K}\phi_{n,i}^d\log\phi_{n,i}^d \tag{3.24}$$

$$\mathbb{E}_q[\log q(c^d|\lambda^d)] = \sum_{l=1}^{C}\lambda_l^d\log\lambda_l^d. \tag{3.25}$$

For a detailed version of these derivations, see Appendix A.

The expectations of the log of the Dirichlet that appears in various of the equations above are given by:

$$\mathbb{E}_q[\log\theta_i^d] = \Psi(\gamma_i^d) - \Psi\bigg(\sum_{j=1}^{K}\gamma_j^d\bigg) \tag{3.26}$$

$$\mathbb{E}_q[\log\beta_{i,j}] = \Psi(\zeta_{i,j}) - \Psi\bigg(\sum_{k=1}^{V}\zeta_{i,k}\bigg) \tag{3.27}$$

$$\mathbb{E}_q[\log\pi_{c,l}^r] = \Psi(\xi_{c,l}^r) - \Psi\bigg(\sum_{t=1}^{C}\xi_{c,t}^r\bigg), \tag{3.28}$$

where $\Psi(\cdot)$ is the digamma function. See Appendix A.1 in Blei et al. (2003a) for the derivation of this standard result.

The only term left to analyse in the lower bound is:

$$\mathbb{E}_q[\log p(c^d|\bar{z}^d,\eta)] = \mathbb{E}_q\left[\log \frac{\exp(\eta_{c^d}^T \bar{z}^d)}{\sum_{l=1}^C \exp(\eta_l^T \bar{z}^d)}\right]$$
$$= \left(\mathbb{E}_q[\eta_{c^d}^T \bar{z}^d] - \mathbb{E}_q\left[\log \sum_{l=1}^C \exp(\eta_l^T \bar{z}^d)\right]\right). \tag{3.29}$$

The first term can be easily computed as:

$$\mathbb{E}_q[\eta_{c^d}^T \bar{z}^d] = \mathbb{E}_q\left[\sum_{j=1}^K \eta_{c^d,j} \bar{z}_j^d\right] = \frac{1}{N_d}\sum_{l=1}^C \lambda_l^d \sum_{n=1}^{N_d} \eta_l^T \phi_n^d. \tag{3.30}$$

As for the second term, it is intractable to compute (Murphy, 2012). We address this issue in similar fashion to Wang et al. (2009), i.e. by applying again Jensen's inequality to lower bound this term as follows:

$$-\mathbb{E}_q\left[\log \sum_{l=1}^C \exp(\eta_l^T \bar{z}^d)\right] \geqslant -\log \sum_{l=1}^C \mathbb{E}_q[\exp(\eta_l^T \bar{z}^d)]$$
$$= -\log \sum_{l=1}^C \mathbb{E}_q\left[\exp(\eta_l^T \frac{1}{N_d}\sum_{n=1}^{N_d} z_n^d)\right]$$
$$= -\log \sum_{l=1}^C \prod_{n=1}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right)$$
$$= -\log \left(\phi_j^d\right)^T \underbrace{\sum_{l=1}^C \exp(\eta_l \frac{1}{N_d}) \prod_{n=1,n\neq j}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right)}_{=h}$$
$$= -\log h^T \phi_j^d \tag{3.31}$$

where we defined $h = \sum_{l=1}^C \exp(\eta_l \frac{1}{N_d}) \prod_{n=1,n\neq j}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right)$.

Now, suppose we have a previous value $(\phi_n^d)^{old}$. We know $\log(x) \leqslant \epsilon^{-1}x + \log(\epsilon) - 1, \forall x > 0, \epsilon > 0$, where equality holds if and only if $x = \epsilon$. If we set $x = h^T\phi_n^d$ and $\epsilon = h^T(\phi_n^d)^{old}$ then, for an individual parameter ϕ_n^d, we have that:

$$-\log(h^T\phi_n^d) \geqslant -(h^T(\phi_n^d)^{old})^{-1}(h^T\phi_n^d) - \log(h^T(\phi_n^d)^{old}) + 1. \tag{3.32}$$

This lower bound is tight when $\phi_n^d = (\phi_n^d)^{old}$.

Putting it all together, we can rewrite the evidence lower bound of Equation 4.21 as:

$$
\begin{aligned}
&\mathcal{L}(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta) \\
&= \sum_{i=1}^{K}\left(\log\Gamma\Big(\sum_{k=1}^{V}\tau_k\Big) - \sum_{j=1}^{V}\log\Gamma(\tau_j) + \sum_{j=1}^{V}(\tau_j-1)\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right)\right) \\
&+ \sum_{r=1}^{R}\sum_{c=1}^{C}\left(\log\Gamma\Big(\sum_{t=1}^{C}\omega_t\Big) - \sum_{l=1}^{C}\log\Gamma(\omega_l) + \sum_{l=1}^{C}(\omega_l-1)\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right)\right) \\
&+ \sum_{d=1}^{D}\left(\log\Gamma\Big(\sum_{j=1}^{K}\alpha_j\Big) - \sum_{i=1}^{K}\log\Gamma(\alpha_i) + \sum_{i=1}^{K}(\alpha_i-1)\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right)\right) \\
&+ \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi_{n,i}^d\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right) \\
&+ \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{j=1}^{V}\sum_{i=1}^{K}w_{n,j}^d\phi_{n,i}^d\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right) \\
&+ \sum_{d=1}^{D}\left(\frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\sum_{n=1}^{N_d}\eta_l^T\phi_n^d - \sum_{n=1}^{N_d}(h^T(\phi_n^d)^{old})^{-1}(h^T\phi_n^d) - \sum_{n=1}^{N_d}\log(h^T(\phi_n^d)^{old}) + N_d\right) \\
&+ \sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\lambda_c^d y_l^{d,r}\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right) \\
&- \sum_{r=1}^{R}\sum_{c=1}^{C}\left(\log\Gamma\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) - \sum_{l=1}^{C}\log\Gamma(\xi_{c,l}^r) + \sum_{l=1}^{C}(\xi_{c,l}^r-1)\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right)\right) \\
&- \sum_{i=1}^{K}\left(\log\Gamma\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) - \sum_{j=1}^{V}\log\Gamma(\zeta_{i,j}) + \sum_{j=1}^{V}(\zeta_{i,j}-1)\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right)\right) \\
&- \sum_{d=1}^{D}\left(\log\Gamma\Big(\sum_{j=1}^{K}\gamma_j^d\Big) - \sum_{i=1}^{K}\log\Gamma(\gamma_i^d) + \sum_{i=1}^{K}(\gamma_i^d-1)\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right)\right) \\
&- \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi_{n,i}^d\log\phi_{n,i}^d - \sum_{d=1}^{D}\sum_{l=1}^{C}\lambda_l^d\log\lambda_l^d
\end{aligned}
\tag{3.33}
$$

To optimize the bound w.r.t the variational parameters γ, ϕ and λ, the

coordinate ascent algorithm was used, so that the bound becomes as tight as possible to the true posterior. Gathering only the terms in the bound that contain the variational parameter of the documents' topic proportions γ gives:

$$\mathcal{L}_{[\gamma]} = \sum_{d=1}^{D}\sum_{i=1}^{K}\Psi(\gamma_i^d)\left(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\right) - \sum_{d=1}^{D}\sum_{i=1}^{K}\Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big) \times \left(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\right) - \sum_{d=1}^{D}\log\Gamma\Big(\sum_{j=1}^{K}\gamma_j^d\Big) + \sum_{d=1}^{D}\sum_{i=1}^{K}\log\Gamma(\gamma_i^d). \tag{3.34}$$

Taking derivatives w.r.t. γ_i^d yields:

$$\frac{\partial\mathcal{L}_{[\gamma]}}{\partial\gamma_i^d} = \Psi'(\gamma_i^d)\Big(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\Big) - \Psi'\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\sum_{j=1}^{K}\left(\alpha_j + \sum_{n=1}^{N_d}\phi_{n,j}^d - \gamma_j^d\right). \tag{3.35}$$

Setting this derivative to zero in order to get a maximum, we get the solution:

$$\gamma_i^d = \alpha_i + \sum_{n=1}^{N^d}\phi_{n,i}^d, \tag{3.36}$$

which can be easily verified by substituting the value for γ_i^d above in the expression for the partial derivatives. This update equation is the same as in standard LDA (Blei et al., 2003a) and Supervised LDA (Blei & McAuliffe, 2007).

Similarly, to optimizing the lower bound w.r.t. the variational parameter of the words' topic assignment $\phi_{n,i}^d$, only the terms in the bound that contain $\phi_{n,i}^d$ are collected. However, this is a constrained maximization problem, since $\sum_{k=1}^{K}\phi_{n,k}^d = 1$, which is necessary for it to be a valid probability distribution. Hence, we need to also add the necessary Lagrange multipliers.

The Lagrangian is then given by:

$$\begin{aligned}\mathcal{L}_{[\phi^d_{n,i}]} &= \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi^d_{n,i}\left(\Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right)\right) \\ &+ \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{j=1}^{V}\sum_{i=1}^{K} w^d_{n,j}\phi^d_{n,i}\left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right)\right) \\ &+ \sum_{d=1}^{D}\left(\frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\sum_{n=1}^{N_d}\eta_l^T\phi^d_n - \sum_{n=1}^{N_d}(h^T(\phi^d_n)^{old})^{-1}(h^T\phi^d_n) + N_d\right) \\ &- \sum_{d=1}^{D}\sum_{n=1}^{N_d}\log(h^T(\phi^d_n)^{old}) - \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi^d_{n,i}\log\phi^d_{n,i} + \mu\left(\sum_{k=1}^{K}\phi^d_{n,k} - 1\right) \\ &. \end{aligned} \tag{3.37}$$

Taking derivatives w.r.t. $\phi^d_{n,i}$ gives:

$$\begin{aligned}\frac{\partial\mathcal{L}_{[\phi^d_{n,i}]}}{\partial\phi^d_{n,i}} &= \Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right) + \sum_{j=1}^{V} w^d_{n,j}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w^d_{n,j}\Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right) \\ &+ \frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\eta_{l,i} - (h^T(\phi^d_n)^{old})^{-1}h_i - \log\phi^d_{n,i} - 1 + \mu. \end{aligned} \tag{3.38}$$

Setting this derivative to zero and solving for $\phi^d_{n,i}$ yields:

$$\begin{aligned}&\Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right) + \sum_{j=1}^{V} w^d_{n,j}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w^d_{n,j}\Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right) \\ &+ \frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\eta_{l,i} - (h^T(\phi^d_n)^{old})^{-1}h_i - \log\phi^d_{n,i} - 1 + \mu = 0 \\ &\Leftrightarrow \phi^d_{n,i} = \exp\left(\Psi(\gamma_i) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right) + \sum_{j=1}^{V} w^d_{n,j}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w^d_{n,j}\Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right)\right. \\ &\left. + \frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\eta_{l,i} - (h^T(\phi^d_n)^{old})^{-1}h_i - 1\right)\exp(\mu). \end{aligned} \tag{3.39}$$

Then, plugging this expression in the constraint and solving for μ (or $\exp(\mu)$)

gives:

$$\sum_{k=1}^{K} \phi_{n,k}^{d} = 1$$

$$\Leftrightarrow \exp(\mu) = \frac{1}{\sum_{k=1}^{K} \exp\Big(\Psi(\gamma_k) - \Psi\big(\sum_{j=1}^{K} \gamma_j\big) + \sum_{j=1}^{V} w_{n,j}^{d} \Psi(\zeta_{i,j})\Big)}$$

$$\times \frac{1}{\sum_{k=1}^{K} \exp\Big(-\sum_{j=1}^{V} w_{n,j}^{d} \Psi\big(\sum_{k=1}^{V} \zeta_{i,k}\big) + \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,k} - (h^T(\phi_n^d)^{old})^{-1} h_k - 1\Big)}. \tag{3.40}$$

Finally, plugging this expression back in the expression for $\phi_{n,i}^{d}$ gives the solution:

$$\phi_{n,i}^{d} \propto \exp\left(\Psi(\gamma_i) + \sum_{j=1}^{V} w_{n,j}^{d} \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d} \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right)\right) \times \exp\left(\frac{\sum_{l=1}^{C} \lambda_l^d \eta_{l,i}}{N_d} - (h^T(\phi_n^d)^{old})^{-1} h_i\right). \tag{3.41}$$

The optimization of the variational parameters of the documents latent true classes λ_l^d is performed in the same way $\phi_{n,i}^{d}$ is: collecting the terms in the bound that contain it and, since $\sum_{k=1}^{C} \lambda_k^d = 1$, because it is a probability distribution, the Lagrange multipliers are necessary. After we take the derivative w.r.t. λ_l^d out of the Lagrangian and setting it to zero, we plug the resultant expression in the constraint and solve it for μ (or $\exp(\mu)$), which gives:

$$\exp(\mu) = \frac{1}{\sum_{k=1}^{K} \exp\left(\eta_k^T \overline{\phi^d} + \sum_{r=1}^{R} \sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r)\right)} \times \frac{1}{\sum_{k=1}^{K} \exp\left(-\sum_{r=1}^{R} \sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right) - 1\right)}. \tag{3.42}$$

Then, by plugging this expression back in the expression for λ_l^d, we get:

$$\lambda_l^d \propto \exp\left(\eta_l^T \overline{\phi^d} + \sum_{r=1}^{R} \sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r) - \sum_{r=1}^{R} \sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right)\right). \tag{3.43}$$

The ξ and ζ optimizations are not constrained maximisation problems, which means that the procedures they require are identical to the γ optimization. Hence, for the sake of simplicity, the derivations needed to achieve their final forms are omitted (presented in Appendix A) . The obtained updates for ξ and ζ parameters are:

$$\zeta_{i,j} = \tau_j + \sum_{d=1}^{D} \sum_{n=1}^{N_d} w_{n,j}^d \phi_{n,i}^d \tag{3.44}$$

$$\xi_{c,t}^r = \omega_t + \sum_{d=1}^{D} \lambda_c^d y_t^{d,r} \tag{3.45}$$

The purpose of these final forms of the variational parameters is to update them in such way that the evidence lower bound becomes as tight as possible to the true posterior. In other words, the variational inference algorithm consists of iteratively optimize each one of these variational parameters in turn until a maximum number of iterations or a given interval of convergence are achieved.

3.3 Parameter estimation

Given a corpus of D documents labeled by R different annotators, $\mathcal{D} = \{\mathbf{w}^d, \mathbf{y}^d\}_{d=1}^D$, maximum likelihood estimates for the class coefficients $\boldsymbol{\eta}$ are found. In order to do this, variational Bayesian EM is used, which replaces the E-step of the Expectation-Maximization algorithm with variational Bayesian inference to find an approximate posterior for the latent variables of each document. In the M-step, as in exact EM, we find maximum likelihood estimates of the parameters using the expected sufficient statistics computed in the E-step. We assume the parameters α, τ and ω are fixed Dirichlet priors, on account of the simplicity of the model.

The corpus-level log-likelihood is given by:

$$\mathcal{L}(\mathcal{D}) = \sum_{d=1}^{D} \log p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D} | \alpha, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\pi}_{1:R}), \tag{3.46}$$

where $\log p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D} | \alpha, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\pi}_{1:R})$ is approximated by equation 4.21, i.e. the lower bound.

As it was done to optimize the variational parameters, in order to obtain maximum likelihood estimates of $\boldsymbol{\eta}$, the terms in the log-likelihood

(equation 3.46) that contain it are gathered. Its objective function is given by:

$$\mathcal{L}_{[\eta_{l,i}]} = \sum_{d=1}^{D} \frac{1}{N^d} \sum_{l=1}^{C} \lambda_l^d \sum_{n=1}^{N^d} \eta_l^T \phi_n^d - \log \sum_{l=1}^{C} \lambda_l^d \prod_{n=1}^{N^d} \left(\sum_{i=1}^{K} \phi_{n,i}^d \exp\left(\eta_{l,i} \frac{1}{N^d}\right)\right). \tag{3.47}$$

Taking derivatives w.r.t. $\eta_{l,i}$ results in:

$$\begin{aligned}\frac{\partial \mathcal{L}_{[\eta_{l,i}]}}{\partial \eta_{l,i}} =& \lambda_l^d \sum_{d=1}^{D} \overline{\boldsymbol{\phi}_n}^d \times \lambda_l^d \sum_{d=1}^{D} \left(- \frac{\sum_{n=1}^{N^d} \left[\frac{1}{N^d} \phi_{n,i}^d \exp\left(\frac{1}{N^d} \eta_{c,i}\right)\right]}{\sum_{l=1}^{C} \lambda_l^d \prod_{n=1}^{N^d} \left(\sum_{i=1}^{K} \phi_{n,i}^d \exp\left(\eta_{l,i} \frac{1}{N^d}\right)\right)} \right) \\ & \times \lambda_l^d \sum_{d=1}^{D} \left(\frac{\prod_{j=1}^{N^d} \left[\sum_{i=1}^{K} \phi_{j,i}^d \exp\left(\frac{1}{N^d} \eta_{c,i}\right)\right]}{\sum_{i=1}^{K} \phi_{n,i}^d - \exp\left(\frac{1}{N^d} \eta_{c,i}\right)} \right). \end{aligned} \tag{3.48}$$

Setting this derivative to zero does not lead to a closed-form solution, hence it is used a numerical method, namely L-BFGS (Wright & Nocedal, 1999), to find an optimum.

3.4 Stochastic variational inference

Stochastic variational inference is a scalable algorithm for approximating posterior distributions. It differs from the variational inference method developed in Section 3.2 in that it updates the variational parameters using a subsample of the data, instead of the entire dataset. While the batch coordinate ascent algorithm for variational inference previously presented iterates between analyzing every document in the corpus to infer the local hidden structure and estimating the model parameters, stochastic variational inference does not require a full pass through the data at each iteration. As it is explained in Section 2.2, the principle behind stochastic optimization (Robbins & Monro, 1951) is that subsets of the data (mini-batches) can provide a noisy representation of the whole dataset. Applying this idea to MA-sLDA, we can find unbiased estimates of the model's variables by subsampling a

mini-batch of documents from the corpus and using it to compute these variables as if that document was observed D times. More specifically, given an uniformly sampled mini-batch, we use the current posterior distributions of the global latent variables $\boldsymbol{\beta}$ and $\boldsymbol{\pi}_{1:R}$ to compute the posterior distribution over the local hidden variables θ^d, $\boldsymbol{z}^d$, c^d, using equations 3.36, 3.41 and 3.43, respectively. These estimates are, then, used to update the global variational parameters, $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}_{1:R}$ by taking a step of size ρ^t in the direction of the stochastic gradients. This process is summarized as follows:

Input : Corpus, variational parameters
Output: Updated variational parameters

Initialize $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}_{1:R}$ randomly and $t = 0$
repeat
  Set $t = t + 1$
  Subsample one or more data points from the corpus $\{\boldsymbol{w}^d\}$
  repeat
    for $\boldsymbol{w}^d \in \{\boldsymbol{w}^d\}$ **do**
      for $n \in \{1...N^d\}$ **do**
        Update ϕ_n^d using equation 3.41
      end
      Update γ^d using equation 3.36
      Update λ^d using equation 3.43
    end
  until ϕ_n^d, γ^d *and* λ^d *converge*;
  Compute the step-size schedule $\rho^t = (t + delay)^{-\kappa}$
  for $k \in \{1...K\}$ **do**
    for $n \in \{1...N^d\}$ **do**
      Update $\zeta_{i,j}^{(t)} = (1 - \rho^t)\zeta_{i,j}^{(t-1)} + \rho^t(\tau_j + D\sum_{n=1}^{N^d} w_{n,j}^d \phi_{n,i}^d)$
    end
  end
  for $r \in \{1...R\}$ **do**
    for $c \in \{1...C\}$ **do**
      for $l \in \{1...C\}$ **do**
        Update
        $\xi_{c,l}^r{}^{(t)} = (1 - \rho^t)\xi_{c,l}^r{}^{(t-1)} + \rho^t(\omega_j + D\sum_{n=1}^{N^d} \lambda_c^d y^{d,r})$
      end
    end
  end
until *convergence*;

Algorithm 3: Stochastic coordinate ascent variational algorithm

3.5 Prediction

After training the model, by inferring the latent variables and estimating the parameters, the proposed model is able to predict the labels for new (unobserved) documents. Therefore, the variables related to the annotators' labels (y, c and π) are ignored from the model's joint distribution and the approximate posterior distribution over the latent variables θ^d and $\boldsymbol{z}^d$ is computed. Letting the topic distribution over words estimated during training be $\boldsymbol{\beta}$, the joint distribution for a single document is given by:

$$p(\theta^d, \boldsymbol{z}^d) = \int p(\boldsymbol{\beta}) p(\theta^d|\alpha) \prod_{n=1}^{N^d} p(z_n^d|\theta^d) p(w_n^d|z_n^d, \boldsymbol{\beta}) d\boldsymbol{\beta}. \tag{3.49}$$

The posterior distribution over $q(\theta^d, \boldsymbol{z}^d) = q(\theta^d|\gamma^d) \prod_{n=1}^{N^d} q(z_n^d|\phi_n^d)$ is computed by deriving a mean-field variational inference algorithm, which results in the same fixed-point updates as in standard LDA (Blei et al., 2003a):

$$\gamma_i^d = \alpha_i + \sum_{n=1}^{N^d} \phi_{n,i}^d$$

$$\phi_{n,i}^d \propto \sum_{j=1}^{V} w_{n,j}^d \beta_{k,j} \exp\left(\Psi(\gamma_i) + \frac{\sum_{l=1}^{C} \lambda_l^d \eta_{l,i}}{N^d} - (h^T (\phi_n^d)^{old})^{-1} h_i \right).$$

Then, using the inferred posteriors and the coefficients η estimated during training, we can make predictions as follows:

$$c_*^d = \arg\max_c \eta_c^T \bar{\phi}^d. \tag{3.50}$$

Chapter 4

Multi-annotator supervised latent Dirichlet allocation for regression

In Chapter 3, we presented a supervised topic model with multiple annotators, that, for each document of an annotated dataset, predicts its true class label c, which belongs to a discrete set of categories. However, there is a wide domain of problems in which the instances are labeled with continuous data, i.e., the label set is formed by real numbers. In this chapter, we describe a version of the MA-sLDA model that handles this sort of data. Therefore, it begins with the definition of the model, continues with the mechanism used for the inference of the model's variables and the estimation of the model's parameters and it concludes with the development of a stochastic variational inference algorithm for this model.

4.1 Proposed model

Similarly to the proposed model for classification, the regression variant of the MA-sLDA model estimates the topics $\boldsymbol{\beta}$ while learns the true label set from multiple answers given by distinct annotators. Hence, analogously to the previously presented model, we consider an annotated dataset $\mathcal{D} = \{\boldsymbol{w}^d, \boldsymbol{y}^d\}_{d=1}^D$ of size D, in which a set of annotations $\boldsymbol{y}^d = \{y^{d,r}\}_{r=1}^R$ given by R different labelers is assigned to each document $\boldsymbol{w}^d$. Yet, in a regression scenario, there is no discrete set of labels, instead, each data point is labeled with a real target value $x^d \in \mathbb{R}$, which belongs to $\boldsymbol{x} = \{x^d\}_{d=1}^D$. This means that the answers of the annotators are also real valued numbers and that each annotator is now associated with a bias b^r and a variance v^r. In other words, what

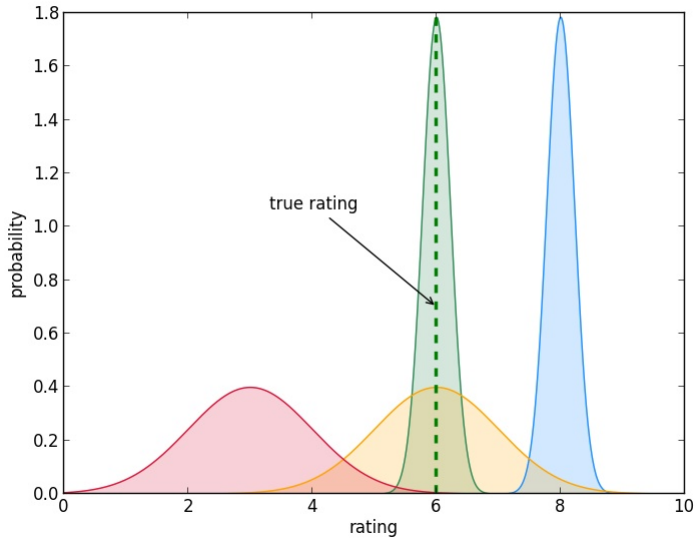


Figure 4.1: Different annotators' variances and biases.

characterizes an annotator now is his probability of giving an answer $y^{d,r}$ given that the true target is x^d and that his noisy version of the document's label is drawn from a normal distribution parametrized by $x^d + b^r$ and v^r. Just like the classification MA-sLDA, this regression model was implemented in C++.

In Figure 4.1, four different kinds of annotators can be distinguished. Imagining the scenario in which each annotator has to predict the number of stars (from 0 to 10) that a movie got from a reviewer based on his review text, we can look at Figure 4.1 and interpret each normal curve as the probability of an annotator's answer given that the true rating was 6. Therefore, the "green annotator" is the best one, since he is right on the target and his answers vary very little (low bias, high precision). The "yellow annotator" has a low bias, but his answers are very uncertain, as they can vary a lot. Contrarily, the "blue annotator" is very precise, but consistently over-estimates the true target (high bias, high precision). Finally, the "red annotator" corresponds to the worst kind of annotator: with high bias and low precision.

The graphical model of MA-sLDA for regression is depicted in Figure 3.1, where the notation is the same as the remaining of this document. In the same way as the standard LDA, it can be seen in the black part of the model that each word w_n^d in a document d is assigned a discrete topic-assignment z_n^d and a topic distribution $\boldsymbol{\beta}$ and that each z_n^d is drawn from the documents

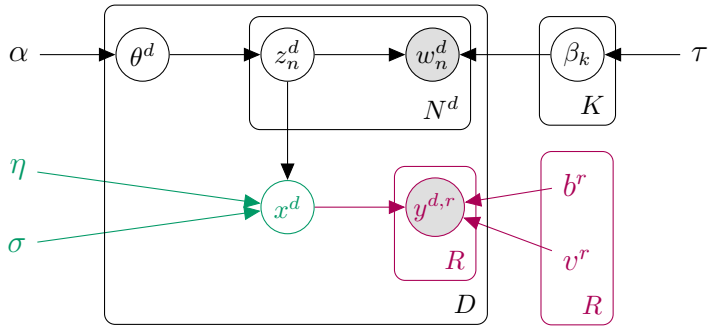


Figure 4.2: Graphical model of MA-sLDA for regression.

distribution over topics θ^d. As for the green elements of the graphical model, we have that the target of the d^{th} document x^d depends on the topic assignment z_n^d and on the variables η and σ that parametrize a normal distribution. This target x^d and the per-annotator biases $\boldsymbol{b}$ and variances $\boldsymbol{v}$ are assumed to give origin to each annotator's answer that is observed $y^{d,r}$, which distinguishes MA-sLDA from the above mentioned approaches. To clarify the learning method behind MA-sLDA, its generative process is described next.

1. For each topic k
   (a) Draw topic distribuition $\beta_k|\tau \sim Dirichlet(\tau)$
2. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$
   (b) For the n^{th} word
      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Multinomial(\beta_{z_n^d})$
   (c) Draw latent (true) value $x^d|\mathbf{z}^d, \eta, \sigma \sim Normal(x^d|\eta^T \bar{z}^d, \sigma)$
   (d) For each annotator r
      i. Draw his answer $y^{d,r}|x^d, b^r, v^r \sim Normal(y^{d,r}|x^d + b^r, v^r)$

According to their definitions, the distributions used in the model are given by:

$$p(\theta^d|\alpha)] = \frac{\Gamma(\sum_{j=1}^{K}\alpha_j)}{\prod_{i=1}^{K}\Gamma(\alpha_i)}\prod_{i=1}^{K}(\theta_i^d)^{(\alpha_i-1)} \tag{4.1}$$

$$p(z_n^d|\theta^d) = \prod_{i=1}^{K}\left(\theta_i^d\right)^{z_{n,i}^d} \tag{4.2}$$

$$p(w_n^d|z_n^d,\boldsymbol{\beta}) = \prod_{j=1}^{V}(\beta_{z_n^d,j})^{w_{n,j}^d} \tag{4.3}$$

$$p(y^{d,r}|x^d,b^r,v^r) = \frac{1}{v^r\sqrt{2\pi}}\,e^{-\frac{\left(y^r-(x^d+b^r)\right)^2}{2(v^r)^2}} \tag{4.4}$$

$$p(x^d|\eta,\mathbf{z}^d,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\,e^{-\frac{(x^d-\eta^T\bar{z})^2}{2\sigma^2}} \tag{4.5}$$

$$p(\beta_i|\tau) = \frac{\Gamma(\sum_{k=1}^{V}\tau_k)}{\prod_{j=1}^{V}\Gamma(\tau_j)}\prod_{j=1}^{V}(\beta_{i,j})^{(\tau_j-1)}. \tag{4.6}$$

$$\tag{4.7}$$

4.2 Approximate inference

In this section, the process responsible for estimating the latent variables and the parameters of the model is demonstrated. Since the regression version of the MA-sLDA is slightly different from the classification one, its joint distribution, given by:

$$p(\boldsymbol{\theta},\mathbf{z}_{1:D},\boldsymbol{x},\mathbf{w}_{1:D},\mathbf{y}_{1:D},\boldsymbol{\beta}|\Theta) = \left(\prod_{i=1}^{K}p(\beta_i|\tau)\right)\prod_{d=1}^{D}p(\theta^d|\alpha)$$
$$\times\left(\prod_{n=1}^{N^d}p(z_n^d|\theta^d)p(w_n^d|z_n^d,\boldsymbol{\beta})\right)p(x^d|\boldsymbol{z}^d,\eta,\sigma)\prod_{r=1}^{R}p(y^{d,r}|x^d,b^r,v^r), \tag{4.8}$$

leads to the posterior:

$$p(\boldsymbol{\theta},\mathbf{z}_{1:D},\mathbf{x},\boldsymbol{\beta}|\mathbf{w}_{1:D},\mathbf{y}_{1:D}) = \frac{p(\boldsymbol{\theta},\mathbf{z}_{1:D},\boldsymbol{x},\mathbf{w}_{1:D},\mathbf{y}_{1:D},\boldsymbol{\beta}|\Theta)}{p(\mathbf{w}_{1:D},\mathbf{y}_{1:D}|\Theta)}$$

$$= \frac{\left(\prod_{i=1}^{K} p(\beta_i|\tau)\right) \prod_{d=1}^{D} p(\theta^d|\alpha) \left(\prod_{n=1}^{N^d} p(z_n^d|\theta^d) p(w_n^d|z_n^d, \boldsymbol{\beta}) \right)}{\int_{\beta_i} \left(\prod_{i=1}^{K} p(\beta_i|\tau)\right) \int_{\theta^d} \prod_{d=1}^{D} p(\theta^d|\alpha) \sum_{\mathbf{z}} \left(\prod_{n=1}^{N^d} p(z_n^d|\theta) p(w_n^d|z_n^d, \boldsymbol{\beta}) \right)}$$
$$\times \frac{\prod_{d=1}^{D} p(x^d|\boldsymbol{z}^d, \eta, \sigma) \prod_{r=1}^{R} p(y^{d,r}|x^d, b^r, v^r)}{\int_{\theta^d} \prod_{d=1}^{D} \int_x p(x^d|\mathbf{z}^d, \eta, \sigma) \prod_{r=1}^{R} p(y^{d,r}|x^d, b^r, v^r)}, \tag{4.9}$$

which is also infeasible to compute. Thus, again, it is applied variational Bayesian inference to approximate this posterior distribution.

Let $q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta})$ denote a variational distribution of the latent variables. Since a fully-factorized (mean-field) approximation is used, we have that:

$$q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta}) = \left(\prod_{i=1}^{K} q(\beta_i|\zeta_i) \right) \prod_{d=1}^{D} q(\theta^d|\gamma^d) \left(\prod_{n=1}^{N^d} q(z_n^d|\phi_n^d) \right) q(x^d|m^d, \nu^d), \tag{4.10}$$

where $\boldsymbol{\zeta}, \boldsymbol{\gamma}, \boldsymbol{\phi}_{1:D}$, $\boldsymbol{m}$ and $\boldsymbol{\nu}$ are the variational parameters.

The the evidence lower bound of MA-sLDA for regression is, then, given by:

$$\begin{aligned}
&\log p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\alpha, \eta, \tau, \sigma, \boldsymbol{b}, \boldsymbol{v}) \\
&= \log \int_{\beta} \int_{\theta} \sum_{\mathbf{z}} \int_{x} \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}|\Theta) q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta})}{q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta})} \qquad (4.11) \\
&\geqslant \mathcal{L}(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta) \\
&= \mathbb{E}_q[\log p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}|\Theta)] - \mathbb{E}_q[\log q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta})] \qquad (4.12)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{i=1}^{K} \mathbb{E}_q[\log p(\beta_i|\tau)] - \sum_{i=1}^{K} \mathbb{E}_q[\log q(\beta_i|\zeta_i)] + \sum_{d=1}^{D} \Bigg(\mathbb{E}_q[\log p(\theta^d|\alpha)] \\
&- \mathbb{E}_q[\log q(\theta^d|\gamma^d)] + \sum_{n=1}^{N^d} \mathbb{E}_q[\log p(z_n^d|\theta^d)] - \sum_{n=1}^{N^d} \mathbb{E}_q[\log q(z_n^d|\phi_n^d)] \\
&+ \sum_{n=1}^{N^d} \mathbb{E}_q[\log p(w_n^d|z_n^d, \boldsymbol{\beta})] + \sum_{r=1}^{R} \mathbb{E}_q[\log p(y^{d,r}|x^d, b^r, v^r)] + \mathbb{E}_q[\log p(x^d|\bar{z}^d, \eta, \sigma)] \\
&- \mathbb{E}_q[\log q(x^d|m^d, \nu^d)] \Bigg).
\end{aligned} \tag{4.13}$$

In Chapter 3, we already derived $\mathbb{E}_q[\log p(\beta_i|\tau)]$, $\mathbb{E}_q[\log p(\theta^d|\alpha)$, $\mathbb{E}_q[\log p(z_n^d|\theta^d)]$ and $\mathbb{E}_q[\log p(w_n^d|z_n^d, \boldsymbol{\beta})]$. Thus, here, only the derivations of the non-common terms of the evidence lower bound are performed.

$$\begin{aligned}
\mathbb{E}_q[\log p(y^{d,r}|x^d, b^r, v^r)] &= \mathbb{E}_q\bigg[\log \frac{1}{\sqrt{2\pi}v^r} \exp\bigg(-\frac{\big(y^r - (x^d + b^r)\big)^2}{2(v^r)^2}\bigg)\bigg] \\
&= -\frac{\big(y^r - m^d - b^r\big)^2}{2v^r} - \frac{1}{2}\log\big(2\pi v^r\big) \qquad (4.14) \\
\mathbb{E}_q[\log p(x^d|\bar{z}^d, \eta, \sigma)] &= \mathbb{E}_q\bigg[\log\bigg(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\bigg(-\frac{(x^d - \eta^T\bar{z}^d)^2}{2\sigma^2}\bigg)\bigg)\bigg] \\
&= \frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\bigg(\mathbb{E}_q\big[(x^d)^2\big] - 2\mathbb{E}_q[x^d]\eta^T\mathbb{E}_q[\bar{z}^d] \\
&\quad + \eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\bigg) \qquad (4.15)
\end{aligned}$$

where:

$$\mathbb{E}_q[x^d] = m^d \qquad (4.16)$$

$$\mathbb{E}_q\big[(x^d)^2\big] = \nu^d + (m^d)^2 \qquad (4.17)$$

$$\mathbb{E}_q[\bar{z}^d] = \bar{\phi}^d = \frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d \qquad (4.18)$$

$$\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big] = \frac{1}{(N^d)^2}\bigg(\sum_{n=1}^{N^d}\sum_{m\neq n}^{N^d}\phi_n^d(\phi_m^d)^T + \sum_{n=1}^{N^d} diag(\phi_n^d)\bigg) \qquad (4.19)$$

Similarly, the only corresponding term of the variational distribution that is not common with the MA-sLDA for classification is derived as:

$$\begin{aligned}
\mathbb{E}_q[\log q(x^d|m^d, \nu^d)] &= \mathbb{E}_q\bigg[\log\bigg(\frac{1}{\sqrt{2\pi\nu^d}} \exp\bigg(-\frac{(x^d - m^d)^2}{2\nu^d}\bigg)\bigg)\bigg] \\
&= \frac{1}{2} - \frac{1}{2}\log(2\pi\nu^d) \qquad (4.20)
\end{aligned}$$

Again, for a detailed version of these equations, see Appendix B.

Using these results, the evidence lower bound is then given by:

$$
\begin{aligned}
&\mathcal{L}(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta) \\
&= \sum_{i=1}^{K}\left(\log\Gamma\Big(\sum_{k=1}^{V}\tau_k\Big) - \sum_{j=1}^{V}\log\Gamma(\tau_j) + \sum_{j=1}^{V}(\tau_j-1)\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right)\right) \\
&- \sum_{i=1}^{K}\left(\log\Gamma\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) - \sum_{j=1}^{V}\log\Gamma(\zeta_{i,j}) + \sum_{j=1}^{V}(\zeta_{i,j}-1)\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right)\right) \\
&+ \sum_{d=1}^{D}\left(\log\Gamma\Big(\sum_{j=1}^{K}\alpha_j\Big) - \sum_{i=1}^{K}\log\Gamma(\alpha_i) + \sum_{i=1}^{K}(\alpha_i-1)\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right)\right) \\
&- \sum_{d=1}^{D}\left(\log\Gamma\Big(\sum_{j=1}^{K}\gamma_j^d\Big) - \sum_{i=1}^{K}\log\Gamma(\gamma_i^d) + \sum_{i=1}^{K}(\gamma_i^d-1)\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right)\right) \\
&- \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{i=1}^{K}\phi_{n,i}^d\log\phi_{n,i}^d + \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{i=1}^{K}\phi_{n,i}^d\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right) \\
&+ \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{j=1}^{V}\sum_{i=1}^{K}w_{n,j}^d\phi_{n,i}^d\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right) \\
&- \sum_{d=1}^{D}\sum_{r=1}^{R} -\frac{\left(y^r - m^d - b^r\right)}{2v^r} - \frac{\log\left(2\pi v^r\right)}{2} \\
&+ \frac{D}{2}\log(2\pi\sigma^2) - \sum_{d=1}^{D}\frac{1}{2\sigma^2}\left(\nu^d + (m^d)^2 - 2m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d + \eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right) \\
&+ \frac{D}{2} - \sum_{d=1}^{D} -\frac{1}{2}\log(2\pi\nu^d) \qquad (4.21)
\end{aligned}
$$

We already demonstrated that the γ and ζ updates obtained by the variational inference algorithm are:

$$\gamma_i^d = \alpha_i + \sum_{n=1}^{N^d} \phi_{n,i}^d, \tag{4.22}$$

$$\zeta_{i,j} = \tau_j + \sum_{d=1}^{D} \sum_{n=1}^{N_d} w_{n,j}^d \phi_{n,i}^d. \tag{4.23}$$

For ϕ, we have a similar update to the one in sLDA:

$$\begin{aligned}\phi_{n,i}^d \propto \exp\left(\Psi(\gamma_i) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right) + \frac{m^d}{N^d \sigma^2}\eta\right) \\ \times \exp\left(-\frac{\eta^T N^d \sum_{m \neq n}^{N^d} \phi_m^d \eta + \eta^T \eta}{2(N^d)^2 \sigma^2}\right).\end{aligned} \tag{4.24}$$

To achieve the updates of the variational parameters of the latent variable x^d: m^d and ν^d, we have the same procedure. Starting by collecting only the terms in the bound that contain m^d, it yields:

$$\mathcal{L}_{[m^d]} = -\sum_{d=1}^{D}\sum_{r=1}^{R} -\frac{\left(y^r - m^d - b^r\right)}{2v^r} + \sum_{d=1}^{D} -\frac{(m^d)^2 - 2m^d \eta^T \frac{1}{N^d} \sum_{n=1}^{N^d} \phi_n^d}{2\sigma^2}. \tag{4.25}$$

Taking derivatives w.r.t. m^d gives:

$$\frac{\partial \mathcal{L}_{[m^d]}}{\partial m^d} = \sum_{r=1}^{R}\left(-\frac{-y^{d,r} + (m^d) + b^r}{v^r}\right) - \frac{-m^d - \eta^T \frac{1}{N^d} \sum_{n=1}^{N^d} \phi_n^d}{\sigma^2}. \tag{4.26}$$

By setting this derivative to zero and solving for m^d, we have that:

$$\frac{\partial \mathcal{L}_{[m^d]}}{\partial m^d} = \sum_{r=1}^{R}\left(-\frac{-y^{d,r} + (m^d) + b^r}{v^r}\right) - \frac{-m^d - \eta^T \frac{1}{N^d} \sum_{n=1}^{N^d} \phi_n^d}{\sigma^2} = 0 \tag{4.27}$$

$$\Leftrightarrow m^d = \sum_{r=1}^{R} \frac{y^{d,r}\sigma^2 - b^r\sigma^2 + v^r\eta^T \frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d}{\sigma^2 + v^r}. \tag{4.28}$$

Collecting the terms in the bound that contain ν gives:

$$\mathcal{L}_{[\nu]} = \sum_{d=1}^{D}\sum_{r=1}^{R} -\frac{1}{2v^r}(\nu^d) - \frac{1}{2\sigma^2}\nu^d + \frac{1}{2}\log(2\pi\nu^d). \tag{4.29}$$

Taking its derivatives, setting them to zero and solving for ν^d yields:

$$\nu^d = \sigma^2 + \sum_{r=1}^{R} v^r. \tag{4.30}$$

By using these updates, we can minimize the Kullback-Leibler divergence between the true distribution and the approximate posterior $q(\boldsymbol{\theta}, \boldsymbol{z}_{1:D}, \boldsymbol{x}, \boldsymbol{\beta})$.

4.3 Parameter estimation

In this version of MA-sLDA, the model parameters are: $\boldsymbol{b}, \boldsymbol{v}$, η and σ. For the sake of simplicity, σ is assumed to be fixed. This section explains the parameter estimation of MA-sLDA for regression, which corresponds to the M-step of the variational EM used for the training of the model. Like the method described in Section 4.2 for optimizing the variational parameters (E-step), we will gather the terms in the log-likelihood that contain each parameter, take their derivatives and equal them to zero, in order to find their maximum likelihood estimates. For b^r, the objective function is:

$$\mathcal{L}_{[b^r]} = \sum_{d=1}^{D}\sum_{r=1}^{R} -\frac{1}{2v^r}\Big(- 2y^{d,r}b^r + 2m^d b^r + (b^r)^2\Big). \tag{4.31}$$

Taking derivatives w.r.t. b^r gives:

$$\frac{\partial\mathcal{L}_{[b^r]}}{\partial b^r} = \sum_{d=1}^{D} \frac{y^{d,r} - m^d - b^r}{v^r}. \tag{4.32}$$

Setting it to zero and solving for b^r results in:

$$b^r = \frac{\sum_{d=1}^{D} y^{d,r} - m^d}{D}. \tag{4.33}$$

For v^r, with the same process we obtain:

$$v^r = \frac{1}{D}\sum_{d=1}^{D}\Big((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2\Big). \tag{4.34}$$

Finally, to estimate $\eta_{l,i}$, the objective function is given by:

$$\mathcal{L}_{[\eta]} = \sum_{d=1}^{D}\left(\frac{1}{\sigma^2}m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d - \frac{1}{2\sigma^2}\eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right). \tag{4.35}$$

Taking derivatives, setting them to zero and solving for η gives:

$$\begin{aligned}
&\sum_{d=1}^{D}\left(\frac{1}{\sigma^2}m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d - \frac{1}{\sigma^2}\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right) = 0 \\
&\Leftrightarrow \eta^T = \sum_{d=1}^{D}\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]^{-1}m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d
\end{aligned} \tag{4.36}$$

When the variational EM algorithm meets a global converge criterion, the model is trained and, therefore, able to predict the targets for new (unobserved) documents. This prediction process is the same as the method described in Section 3.5.

4.4 Stochastic variational inference

As we did for the classification model from Chapter 3, we can envision developing a stochastic variational inference for the proposed regression model. In this case, the only global latent variables are the per-topic distributions over words β_k. As for the local latent variables, instead of a single variable λ^d, we now have two variables per-document: m^d and ν^d. The stochastic variational inference can then be summarized as shown in Algorithm 4. For added efficiency, one can also perform stochastic updates of the annotators

biases b^r and variances v^r, by taking a step in the direction of the gradient of the noisy evidence lower bound scaled by the step-size ρ^t.

Input : Corpus, variational parameters
Output: Updated variational parameters

Initialize $\boldsymbol{\zeta}$ randomly and $t = 0$
repeat
  Set $t = t + 1$
  Subsample one or more data points from the corpus $\{\boldsymbol{w}^d\}$
  repeat
    for $\boldsymbol{w}^d \in \{\boldsymbol{w}^d\}$ **do**
      for $n \in \{1...N^d\}$ **do**
        Update ϕ_n^d using equation 3.41
      end
      Update γ^d using equation 3.36
      Update m^d using equation 4.28
    end
    Update ν^d using equation 4.30
  until ϕ_n^d, γ^d *and* λ^d *converge*;
until *convergence*;
Compute the step-size schedule $\rho^t = (t + delay)^{-\kappa}$
for $k \in \{1...K\}$ **do**
  for $n \in \{1...N^d\}$ **do**
    Update $\zeta_{i,j}^{(t)} = (1 - \rho^t)\zeta_{i,j}^{(t-1)} + \rho^t(\tau_j + D\sum_{n=1}^{N^d} w_{n,j}^d \phi_{n,i}^d)$
  end
end

Algorithm 4: Stochastic coordinate ascent variational algorithm.

Chapter 5

Experimental evaluation of multi-annotator supervised latent Dirichlet allocation

In this section, the proposed multi-annotator supervised LDA models for classification (MA-sLDAc) and regression (MA-sLDAr) are validated using simulated annotators on popular corpora and using real multiple-annotator labels obtained from Amazon Mechanical Turk. Namely, we shall consider the following real-world problems:

1. classifying posts and news stories;
2. classifying images according to their content;
3. predicting number of stars of a given user gave to a restaurant based on the review;

We will start by evaluating the classification model proposed in Chapter 3 in the first two problems (see 5.1) and use the last regression problem for evaluating the model proposed in Chapter 4 (see Section 5.2).

5.1 Classification

5.1.1 Data

MA-sLDA was tested in three different well-known labeled datasets: Reuters-21578 (Lewis, 1997), 20-Newsgroups and LabelMe (Russell et al., 2008).

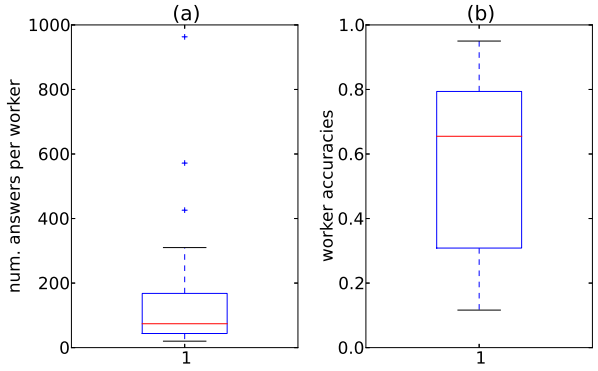


Figure 5.1: Boxplot of the number of answers per annotator (a) and their respective accuracies (b) for the Reuters dataset.

	Train instances	**Test instances**
Earnings	982	2731
Acquisitions	519	1536
Crude-oil	70	251
Trade	69	229
Money effects	55	190
Interest rates	47	150
Shipping	39	103
Grain	18	27
Total	1799	5217

Table 5.1: Class distribution of Reuters-21578.

Reuters-21578 is a group of manually categorized newswire stories with labels such as *Acquisitions*, *Crude-oil*, *Earnings* or *Grain*. It is characterized by having a very skewed distribution of classes over documents, as it can be seen in Table 5.1. Only the documents belonging to the Modified Apte (*ModApte*) split [1] were considered. *ModApte* split is a standard division of the Reuters collection into train and a test sets. However, we had to filter the documents to obtain just single-labeled ones. Of these, 1800 documents were submitted to Amazon Mechanical Turk for multiple annotators to label, giving an average of 3.007 answers per document. Since this train set of 1800 documents was verified to ensure good prediction performances, the remaining 5216 documents were used for testing.

[1] http://www.daviddlewis.com/resources/testcollections/reuters21578/

The collected answers yield an average annotator accuracy of 56.8%. Applying majority voting to these answers reveals a ground truth accuracy of 71.0%. Figure 5.1 shows the boxplots of the number of answers per annotator and the workers accuracies. Observe how applying majority voting yields a higher accuracy than the median accuracy of the workers.

	Train instances	**Test instances**
Computers	3586	1268
Recreative	2980	986
Science	2979	963
Politics	1991	629
Total	11536	3846

Table 5.2: Class distribution of 20-Newsgroups.

20-Newsgroups consists of twenty thousand messages taken from twenty newsgroups. It is a single-label dataset that, comparing to Reuters-21578, has a much more uniform class distribution, which is shown in Table 5.2. This corpus is divided in six super-classes, which are, in turn, partitioned in several sub-classes. Yet, only the four most populated super-classes were used: *Computers*, *Science*, *Politics* and *Recreative*.

To process the natural language of the documents of both Reuters-21578 and 20-Newsgroups corpora, stemming was applied, in order to map related words to the same stem, and stop words were removed, since they are irrelevant for the classification task.

	Train instances	**Test instances**
Open country	154	256
Forest	138	190
Coast	134	226
Tall building	131	225
Mountain	128	246
Inside city	116	192
Street	110	182
Highway	89	171
Total	1000	1688

Table 5.3: Class distribution of LabelMe.

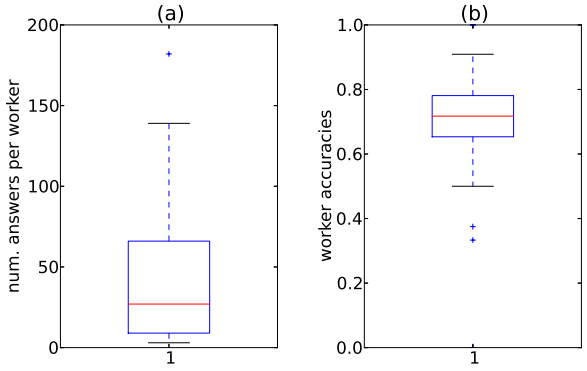


Figure 5.2: Boxplot of the number of answers per annotator (a) and their respective accuracies (b) for the LabelMe dataset.

In contrast to the Reuters and Newsgroups corpora, LabelMe is an open online tool to annotate images, hence, it allows to evaluate the model in non-textual data. For images to be handled by our algorithm, they have to be encoded the same way text is: each image is seen as a document formed by fragments that represent words. Therefore, they were processed following the setting in Fei-Fei & Perona (2005): using a 128-dimensional SIFT (Lowe, 1999) region descriptors given by a sliding grid spaced at 16×16 pixels. This sliding grid extracts local regions of the image of sizes randomly sampled between 16×16 and 32×32 pixels, which are, then, assigned to one of the 200 clusters obtained by a K-Means algorithm (Kadir & Brady, 2001) previously performed in the image collection. This means that there is a vocabulary of 200 different "visual words" that constitute the images. Table 5.3 shows this dataset's class population. Of the total of 2688 labeled images, 1000 images were given to Amazon Mechanical Turk workers to classify with one of the classes above. The train set is smaller than the test set for the same reason of the train-test split considered for the Reuters dataset: as long as the train set size allows the good learning of the model, there is no reason to submit more documents in Amazon Mechanical Turk.

Each image was labeled by an average of 2.547 workers, with a mean accuracy of 69.2%. When majority voting is applied to the collected answers, a ground truth accuracy of 76.9% is obtained. Figure 5.2 shows the boxplots of the number of answers per annotator and the workers accuracies. Interestingly, the worker accuracies are much higher and their distribution is much more concentrated than on the Reuters-21578 data (see Figure 5.1), which suggests that this is an easier task for the Amazon Mechanical Turk workers.

Dataset	Annot. source	Num. ans. per inst. (± stddev.)	Mean annotators accuracy (± stddev.)	Maj. vot. accuracy
20 Newsgroups	Simulated	1.000 ± 0.000	0.405 ± 0.182	0.405
Reuters-21578	Mech. Turk	3.007 ± 1.019	0.568 ± 0.262	0.710
LabelMe	Mech. Turk	2.547 ± 0.576	0.692 ± 0.181	0.769

Table 5.4: Overall statistics of the classification datasets used in the experiments.

As it can be noted from Table 5.4, for the 20-Newsgroups corpus, MA-sLDA was validated in a slightly more controlled environment, by simulating multiple annotators with different levels of expertise. While the process required to gather real annotators' labels consisted only in submitting a task in Amazon Mechanical Turk, to obtain the artificial annotators it took more steps: firstly, a mean desirable accuracy was assigned to each one of the annotators; secondly, each annotator's confusion matrix was randomly simulated based on those accuracies and, finally, the annotations were randomly generated according to both the real label of the instance being annotated and the annotators' confusion matrices. In other words, the line c of the r^{th} annotator's confusion matrix $\boldsymbol{\pi}^r$ can be perceived as a set of multinomial parameters, so that one could sample the r^{th}-annotator's answer $y^r = l$ with some probability $\pi^r_{c,l}$.

5.1.2 Experimental procedure

To assess the implemented model, the following methods were compared with it:

- LDA + LogReg (mv): LDA was employed to extract topics from the data and, then, a logistic regression was applied to classify the instances based on their topic distributions. In order to perform classification, the most voted class by the annotators for each instance acted as its label.

- sLDA (mv): sLDA model was used with the label set obtained by performing the majority voting (mv) method on the annotations.

- LDA + Raykar's: Again, LDA was applied to infer the topic distributions per instance. Then, the framework of (Raykar et al., 2009) was used to perform classification based on the annotators' answers.

- LDA + Rodrigues's: This approach is identical to the aforementioned method, but the framework from (Rodrigues et al., 2013) was applied alternatively to the Raykar's learning algorithm.

Since these algorithms initialize their variables randomly, each one of these methods was run 30 times. Thus, the values presented in the images of the following Section (5.1.3) are the means of the 30 accuracies obtained for each test. We chose accuracy as the metric to compare our model with the remaining approaches, as it is the metric commonly applied in this sort of work. This way, we make it easy to confront the presented results with related ones.

5.1.3 Results

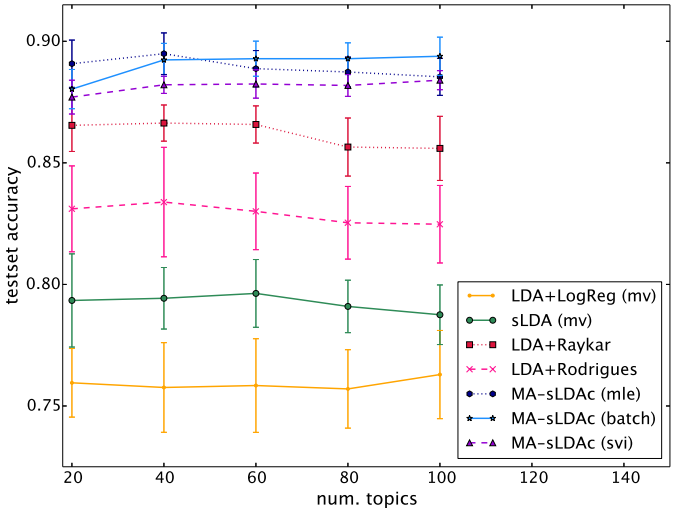


Figure 5.3: Average test set accuracy (over 30 runs; stddev.) of the different approaches on the Reuters data.

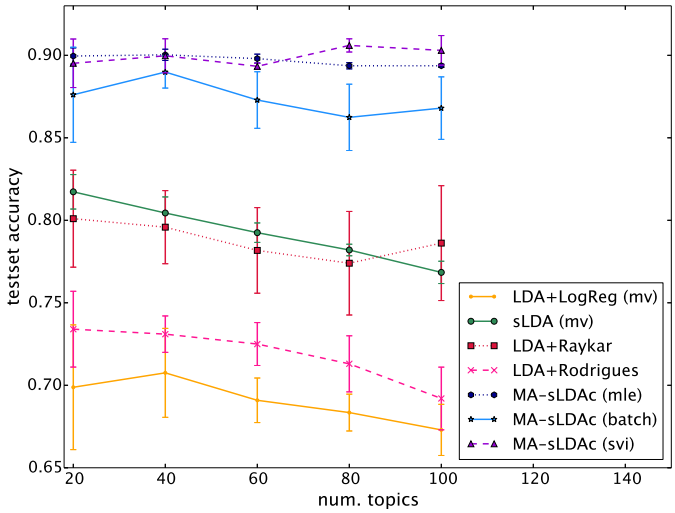


Figure 5.4: Average test set accuracy (over 30 runs; stddev.) of the different approaches on the 20-Newsgroups data.

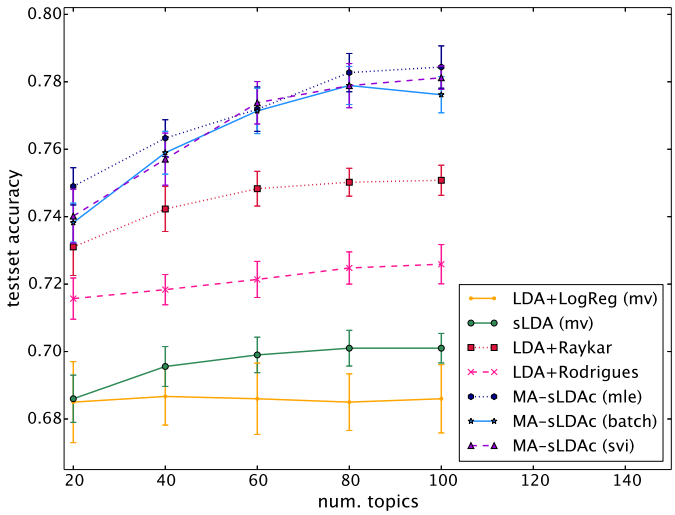


Figure 5.5: Average test set accuracy (over 30 runs; stddev.) of the different approaches on the LabelMe data.

The experimental results are presented in Figures 5.3, 5.4 and 5.5. As it can be seen, MA-sLDA outperforms the remaining methods in all the experiments conducted.

Furthermore, it can be perceived that the best results are obtained in the 20-Newsgroups dataset. Even with the worst quality annotators, the model achieves the highest accuracies comparing with the other datasets, especially comparing to the LabelMe dataset. This can be explained by the fact that this corpora has only four very distinct classes, which, probably, makes it an easier dataset to perform classification.

On the other hand, the results show that the worst accuracies occur in the LabelMe set of images, which have the best annotators. However, there are only 3 annotations per instance, which means that there is less information about the ground truth labels. Yet, in this dataset the difference between the MA-sLDA and the other approaches mean accuracy is superior to the other datasets outcomes.

Other conclusions that can be derived from this set of experiments is that the annotator-aware models always outperform the classifiers that learn on the most voted labels, which evinces the vulnerability of the majority voting method. Similarly, like it was expected, sLDA ensures better inferred labels than the separated approaches for extracting topics and classifying the instances.

Also, to be noticed that all the approaches with the exception of MA-sLDA require multiple disconnected methods: LDA to extract topics, classification to infer labels for the unobserved instances and majority voting to obtain labels from the annotations. Therefore, MA-sLDA is the only approach that combines all the procedures and, for that reason, its modeling takes advantage of all the fullness of the information contained in the data.

All of these are conclusions readable from the figures. Nevertheless, we are interested in assessing the statistical significance of the results obtained. In order to do it, we selected the different models' accuracies for 40 topics to, firstly, use the Kolmorov-Smirnov test to verify if there was statistic facts supporting that the data was drawn from the a normal distribution. For each one of the approaches above compared, except for MA-sLDA with stochastic variational inference, and for each dataset, in Table 5.5, we show the p value resultant from the Kolmogorov-Smirnov test. Since the stochastic variational inference version of MA-sLDA is just a more scalable instance of MA-sLDA, inheriting all its remaining properties, it was not included in this study.

As it can be seen, all the p values are less than 0.05, which means that we can reject the null hypothesis that the data is normally distributed for each one of the cases. We proceed, therefore, to Kruskal-Wallis tests: we

	Reuters	**LabelMe**	**20-Newsgroups**
LDA+LogReg (mv)	9.681e-14	1.366e-14	2.776e-15
sLDA (mv)	2.22e-16	9.57e-14	1.024e-12
LDA + Raykar's	2.2e-16	3.197e-14	6.994e-15
LDA + Rodrigues's	5.44e-15	1.665e-15	2.2e-16
MA-sLDA (mle)	2.2e-16	1.887e-15	6.062e-14
MA-sLDA (batch)	2.2e-16	1.288e-14	4.441e-16

Table 5.5: Kolmogorov-Smirnov test's p values.

have more than two categories to compare, the runs are not paired and the data is not parametric.

	Reuters	**LabelMe**	**20-Newsgroups**
p value	<2.2e-16	<2.2e-16	<2.2e-16

Table 5.6: Kruskal-Wallis tests' p values.

The results shown in Table 5.6 allows us to believe that there is significant difference in the six methods used in the three datasets. Hence, Mann-Whitney tests were then used. For the sake of simplicity, only the pairs formed by MA-sLDA and the most accurate baseline were tested. In this way, we can measure the improvements of the proposed model compared to the best state of the art approach in the classification problems considered.

	Reuters	**LabelMe**	**20-Newsgroups**
p value	2.183e-07	<2.2e-16	0.001
z-score	-6.611	-6.257	-3.062

Table 5.7: Mann-Whitney tests' p values (one-tailed) and z-scores.

Comparing MA-sLDA ($b$atch) with LDA+Raykar's method for all the three corpora, the Mann-Whitney tests outputs one-tailed p values that are less than $\frac{0.05}{15} \approx 0.0033$, which is the significance threshold with the proper Bonferroni correction (Bonferroni, 1935). This means that the accuracies of MA-sLDA are significantly superior from the ones obtained by the LDA+Raykar's approach. Moreover, the z-scores reveal the effect size r of this differences. In the case of 20-Newsgroups dataset, $r_{20Newsgroups} = 0.228$, meaning that the effect size is small. However, for the Reuters and LabelMe dataset, $r_{Reuters} = 0.493$ and $r_{LabelMe} = 0.466$, thus, proving a medium effect size of the difference between the proposed model and the best state of the art approach tested.

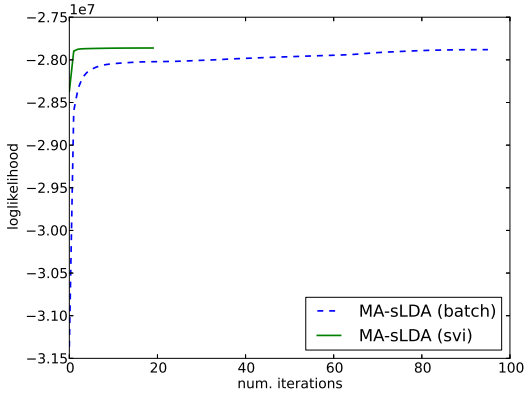
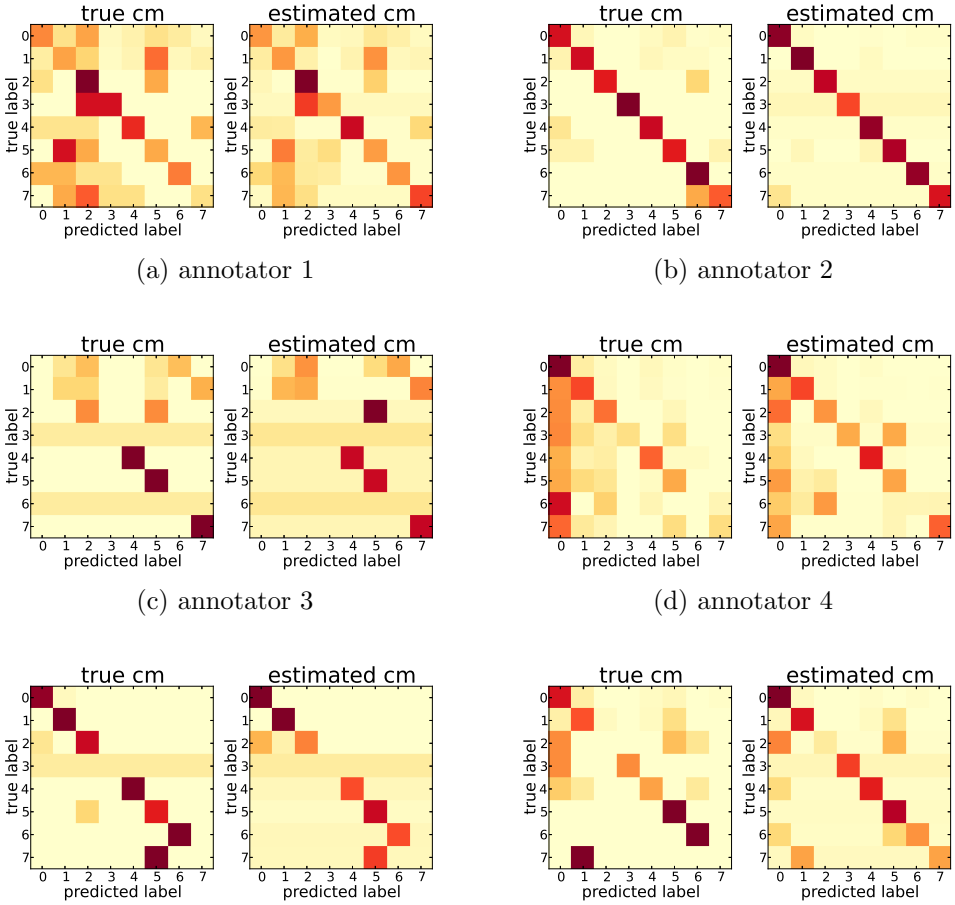


Figure 5.6: Comparison of the log marginal likelihood between the *batch* and the stochastic variational inference (*svi*) algorithms on the 20-Newsgroups corpus.

Concerning the evaluation of the computational advantages of the stochastic variational inference (*svi*) over the *batch* algorithm, the log marginal likelihood (or log evidence) was plotted against the number of iterations. Figure 5.6 shows this comparison. Not surprisingly, the *svi* version converges much faster to higher values of the log marginal likelihood when compared to the *batch* version, which reflects the efficiency of the *svi* algorithm. However, it is important to note that this increased efficiency does not necessarily translates in higher predictive accuracies, as the results of Figure 5.3 demonstrate.

In order to verify that the proposed model was estimating the (normalized) confusion matrices $\boldsymbol{\pi}^r$ of the different workers correctly, a random sample of them was plotted against the true confusion matrices (i.e. the normalized confusion matrices evaluated against the true labels). Figures and 5.7 5.8 show the obtained results, where the colour intensity of the cells increases with the magnitude of the value of $p(y^{d,r} = l|c^d) = \pi^r_{c,l}$. Using this visualization we can verify that the Amazon Mechanical Turk workers are quite heterogeneous in their labeling styles and in the kind of mistakes they make, with several workers showing clear biases (e.g. workers 3 and 4 in Figure 5.7, and workers 1 and 5 in Figure 5.8), while others made mistakes more randomly (e.g. worker 1 in Figure 5.7, and worker 6 in Figure 5.8). Nevertheless, the proposed is able to capture these patterns correctly and account for effect.



(a) annotator 1

(b) annotator 2

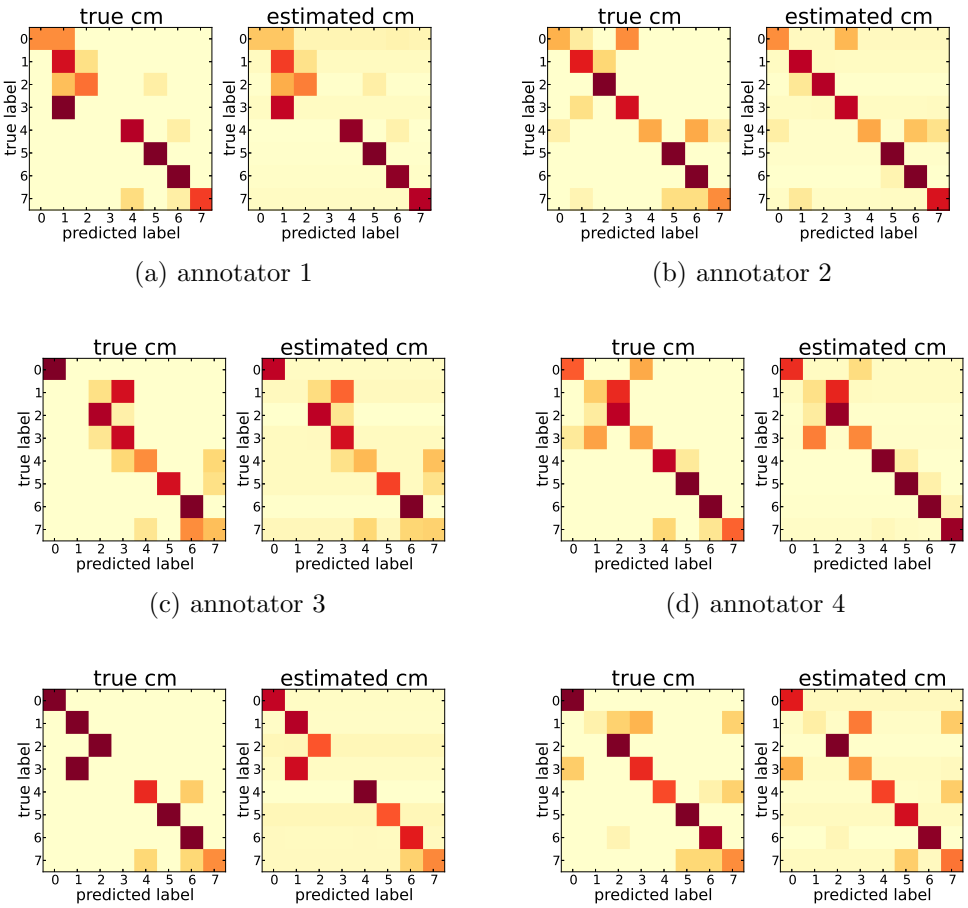
(c) annotator 3

(d) annotator 4

(e) annotator 5

(f) annotator 6

Figure 5.7: True vs. estimated confusion matrix (cm) of 6 different workers of the Reuters-21578 dataset.



(a) annotator 1

(b) annotator 2

(c) annotator 3

(d) annotator 4

(e) annotator 5

(f) annotator 6

Figure 5.8: True vs. estimated confusion matrix (cm) of 6 different workers of the LabelMe dataset.

5.2 Regression

5.2.1 Data

To validate MA-sLDAr, the we8there dataset, consisting of user-submitted restaurant reviews from the website we8there.com, was considered. The we8there corpus was originally presented in Mauá & Cozman (2009) and it contains 6260 reviews. For each review, there is a five-star rating on four specific aspects of quality (food, service, value, and atmosphere) as well as the overall experience. The goal is, then, to predict the overall experience of

Dataset	Train/test sizes	Ann. source	Num. ans./inst. (± stddev.)	Mean annotators R^2 (± stddev.)	Mean answer R^2
we8there	4624/1542	Simulated	2 ± 0.000	0.680 ± 0.260	0.969

Table 5.8: Overall statistics of the regression dataset used in the experiments.

the user based on his comments in the review. We apply the same preprocessing as in Taddy (2013), which consists in tokenizing the text into bigrams and discarding those that appear in less than ten reviews. The preprocessing of the documents consisted of stemming and stop-words removal. After that, 75% of the documents were randomly selected for training and the remaining 25% for testing.

For this dataset, artificial annotations were generated. As with the classification model, we seek to simulate an heterogeneous set of annotators in terms of reliability and bias. Hence, in order to simulate an annotator r, we proceed as follows: let x^d be the true review of the restaurant; we start by assigning the reviewers a given bias b^r and variance v^r, depending on what type of annotator we wish to simulate (see Figure 4.1); we then sample a simulated answer as $y^{d,r} \sim Normal(x^d + b^r, v^r)$. Using this procedure, we simulated 5 annotators with the following (bias, variance) pairs: (0.1, 0.1), (-0.1, 0.1), (1, 0.1), (-1, 0.1) and (0.01, 1). The goal is to have 2 good annotators (low bias, low variance), 2 biased annotators and 1 imprecise. The coefficients of determination (R^2) of the simulated annotators are: [0.939, 0.940, 0.402, 0.392, 0.438]. Computing the mean of the answers of the different annotators yields a R^2 of 0.969. Table 5.8 gives an overview on the statistics of datasets used in the regression experiments.

5.2.2 Experimental procedure

We compare the proposed model (MA-sLDAr) with the two following baselines:

- LDA + LinReg (mean): This baseline corresponds to applying unsupervised LDA to the data and learning a linear regression model on the inferred topics distributions of the documents. The answers from the different annotators were aggregated computing the mean.

- sLDA (mean): This corresponds to using the regression version of sLDA Blei & McAuliffe (2007) with the target variables obtained by computing the mean of the annotators' answers.

5.2.3 Results

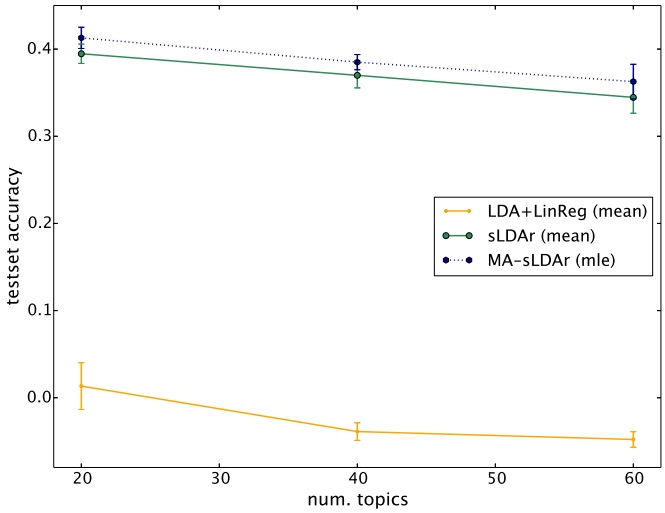


Figure 5.9: Average test set accuracy (over 30 runs; stddev.) of the different approaches on the we8there's data.

Figure 5.9 shows the results obtained for different numbers of topics. Due to the stochastic nature of both the annotators simulation procedure and the initialization of the variational Bayesian EM algorithm, we repeated each experiment 30 times and report the average R^2 obtained with the corresponding standard deviation. The results obtained show the improved performance of MA-sLDAr *mle* (the MA-sLDAr version that uses maximum likelihood estimates described in Appendix E) over the other methods. Unfortunately, the experiments for the *batch* and *svi* versions did not finished in time for their results to be presented. These two versions of our model require the optimization of more parameters and, for this reason, their experimental procedure takes more time.

Besides the contrast among the results of the three approaches that favors the proposed model, the figure also makes it clear the benefit of the use of integrated approaches against two-stage procedures.

	we8there
LDA+LinReg (mean)	1.236e-07
sLDA (mean)	6.244e-11
MA-sLDA (mle)	3.32e-11

Table 5.9: Kolmogorov-Smirnov test's p values.

In order to assess the statistical significance of the results obtained, the same statistical evaluation of Section 5.1.3 was conducted. The different models' R^2 values for 40 topics were compared to the normal distribution through the Kolmorov-Smirnov test. The resultant p values can be seen in Table 5.9.

We can verify that all p values are less than 0.05, thus, in neither case we can assume parametric data. For this reason, because we have more than two categories and because the runs are not paired, the Kruskal-Wallis test was applied to measure the differences between the three methods.

	we8there
p value	<3.799e-15

Table 5.10: Kruskal-Wallis tests' p values.

The results shown in Table 5.10 make us believe that there is significant difference among the three sets of R^2 values. Hence, the Mann-Whitney test was used to compare MA-sLDA with the sLDA (mean) method.

	we8there
p value	7.347e-05
z-score	-4.033

Table 5.11: Mann-Whitney tests' p values (one-tailed) and z-scores.

The output of the Mann-Whitney test exhibited in Table 5.11 shows that the one-tailed p value is less than the significance threshold corrected as in Bonferroni (1935)): $\frac{0.05}{3} \approx 0.0166$, proving the statistical significance of the improvements obtained by MA-sLDA when compared to the sLDA (mean) approach. Moreover, from the z-scores, we get $r_{we8there} = 0.301$, meaning that the effect is medium.

We also studied if the proposed model was, indeed, estimating the biases and variances of the different workers correctly. Figure 5.10 shows the true values against the estimates of MA-sLDAr for our 5 simulated workers, where the higher colour intensities indicate higher values. Ideally, the colour of two horizontally-adjacent squares would then be of similar shades and this

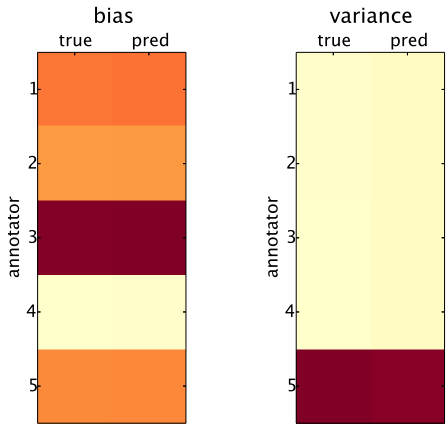


Figure 5.10: Average test set accuracy (over 30 runs; stddev.) of the different approaches on the we8there's data.

is indeed what happens, as Figure 5.10 clearly demonstrates. Interestingly, the fact that the biases are being correctly estimated justifies the inclusion of a bias parameter in the proposed model, which contrasts with previous works (Raykar et al., 2009; Groot et al., 2011).

Chapter 6

Case study: supervised topic models for human mobility prediction

In the previous chapters we developed a supervised topic model that is able to learn from multiple annotators with different levels of expertise. In this chapter, we present a case study of the use of supervised topic models for human mobility prediction.

While our transport systems are generally designed for habitual behaviour, the dynamics of large and mega cities systematically push it to its limits. Particularly, transport planning and operations in large events are well known to be a challenge. Not only they imply stress to the system on an irregular basis, their associated mobility behavior is also difficult to predict.

With this problem as motivation, we present iOracle, an approach that, given minimal data (event title, date and location) obtained from a feed of event listings and a set of automated search queries, predicts the occurrence of public transport overcrowding hotspots. Since iOracle works on textual data, a supervised topic model is used as classification mechanism.

This work was developed in the Singapore-MIT Alliance for Research and Technology centre (SMART) in collaboration with four other researchers: Professor Francisco Câmara Pereira, Filipe Rodrigues, Manuel Frutuoso and Stanislav Borysov, who contributed with guidance, suggestions, valuable discussions, data preparation and preprocessing.

6.1 Proposed approach

The goal of the proposed approach is to predict overcrowding hotspots, by using a historical smartcard dataset, of 5 months of data, together with a dataset of event information captured from the internet. A hotspot is defined as an observation of number of public transport arrivals exceeding the 80th percentile for at least 30 minutes, in practice, representing a continuous series of bus/subway standing trips. This research was developed in collaboration with the Singapore Land Transport Authority (LTA) and aims to solve the concrete objective of having a weekly feed with potential upcoming overcrowding alarms.

iOracle combines MedLDA (Zhu et al., 2009), a maximum margin classifier topic modeling algorithm, with textual data obtained from automatically generated queries. These queries are constructed from basic event information (title, location, time) obtained from event listing websites.

This approach is novel in two ways: web search query content is explicitly included in the model (in fact, it is the only input for the model); we apply a supervised topic model, which guarantees to search for the query content that is more relevant for hotspot prediction. By using such a simple input structure, we also make our model easily portable from city to city and even domain (e.g. stock market).

6.2 Classification mechanism

MedLDA builds on the max-margin principle to train for classification as well as regression. As it is explained in Section 2.3.2, MedLDA looks for the topics that enable the maximum possible margin. Namely, in this case, topics will be preferred that either strongly support, or strongly oppose, the likelihood of a hotspot. The result is a combination of the SVM's principles and LDA that are trained jointly. As a consequence, MedLDA inherits the robustness properties of the SVM and it demonstrated to outperform several state-of-the-art approaches, such as sLDA (Blei & McAuliffe, 2007) and DiscLDA (Lacoste-julien et al., 2009).

In fact, the choice of MedLDA for iOracle was subsequent to the analysis of the system behaviour when other classification mechanisms were employed. sLDA, as well as two-step methods including LDA and a logistic regression and LDA and a SVM classifier were tested. Not surprisingly, sLDA and MedLDA proved to have better classification results than the separated approaches. Also, MedLDA showed to be the most accurate and efficient

model. The observed difference between sLDA and MedLDA training times was the strongest reason why MedLDA was chosen. Especially in this practical application context, the efficiency is a crucial factor and sLDA did not exhibit to be as suitable as MedLDA.

The MedLDA's classification rule for the document d is determined by:

$$c^* = \arg\max_c \mathbb{E}[\eta_c^T \bar{z}|\alpha, \boldsymbol{\beta}], \tag{6.1}$$

where the notation is the same used in the previous chapters: $(c^*)^d$ is the predicted class, η_c is a class-specific set of weights, $\bar{z}^d$ the topic proportions of the document, α is the Dirichlet prior for the distribution of documents over topics and β the distribution of words for each topic k (see Section 2.3.2 for further details).

For this particular case, we are interested in both η_c and $\bar{z}^d$, as the former can help us understand the relative weight of each topic in the determination of a hotspot, while the latter describes, for each event, the proportion of each topic. We will also analyse the word distributions, β_k for each topic.

6.3 Methodology

Figure 6.1 summarizes the architecture of the *internet Oracle* (iOracle). It works in two different modes: training and prediction. The training mode consists of collecting a set of events that share the same time window as the smartcard dataset available. For each event, a set of queries is generated and a binary label that identifies whether a hotspot has occurred or not is created. The query results are aggregated into a single query document, as explained below, becoming the input vector, x, while the label becomes the target variable, y. As a result of this process, a set of K topics are estimated, together with the per class max-margin η parameters, which are saved for the prediction mode. As happens in (Zhu et al., 2009), the variational approximation for η, which is represented as $q(\eta)$, follows a normal distribution, so we only need to save the sufficient statistics (mean and variance).

The objective is to classify potential hotspots for future events. In prediction mode, for each candidate event (retrieved from the list of announced events), the corresponding queries are run, in order to obtain the query document d. The iOracle algorithm performs inference on the MedLDA model, by applying equation 6.1, using $q(\eta)$ and the topic proportions $\bar{z}_d$ of the query document.

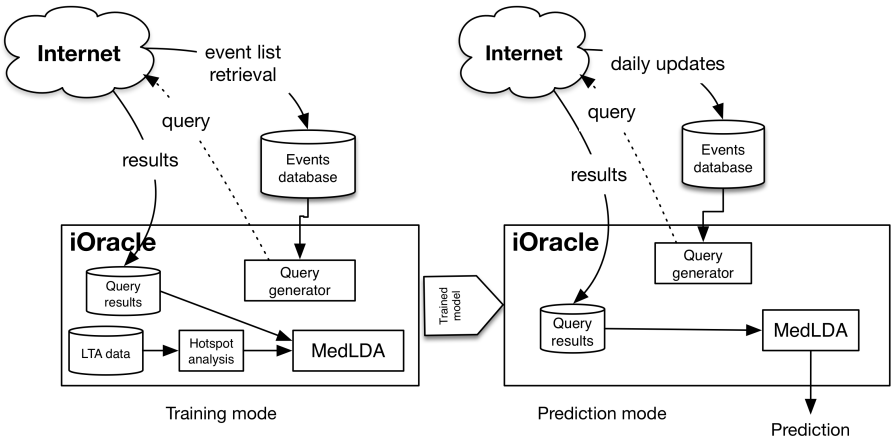


Figure 6.1: Architecture of iOracle.

6.3.1 Sources of data

As it can be seen in Figure 6.1, iOracle handles two kinds of data: the data from LTA and the internet-sourced data (in Figure 6.1: "Events database" and "Query results"). LTA's data comes from the Singaporean fully integrated smartcard transit system, called EZLink. In the EZLink system, all public transport modes (bus, Mass Rapid Transit, or MRT, and light rail, or LR) have a distance-based fare system, calculated by tap-in and tap-out in each ride. Fare is, thus, calculated based on the distance between tap-in and tap-out GPS data. As a consequence, each individual trip is recorded in the system with high spatial and temporal precision.

After selecting 5 different areas (see Figure 6.2) jointly with LTA, we were provided with 5 months of data, from November 2012 to February 2012. In Figure 6.3, it is shown the study areas in detail, in shaded polygons. The Stadium area comprises two venues: Singapore Indoor Stadium (SIS) and Kallang theatre. This area is generally isolated from other attractions or major shopping malls. Orchard and Somerset form part of the "Orchard road" district, famous for its shopping malls. It also holds a significant number of events. Although individually they may have low significance, when put together they can attract a large part of trips.

Harbourfront is the transportation gateway that gives access to Sentosa Island, the largest entertainment area in Singapore. Finally, Singapore Expo regularly hosts exhibition fairs and festivals.

Besides the EZLink data, it is fundamental for this approach to have an events database with the lists of event titles, locations, start times (and

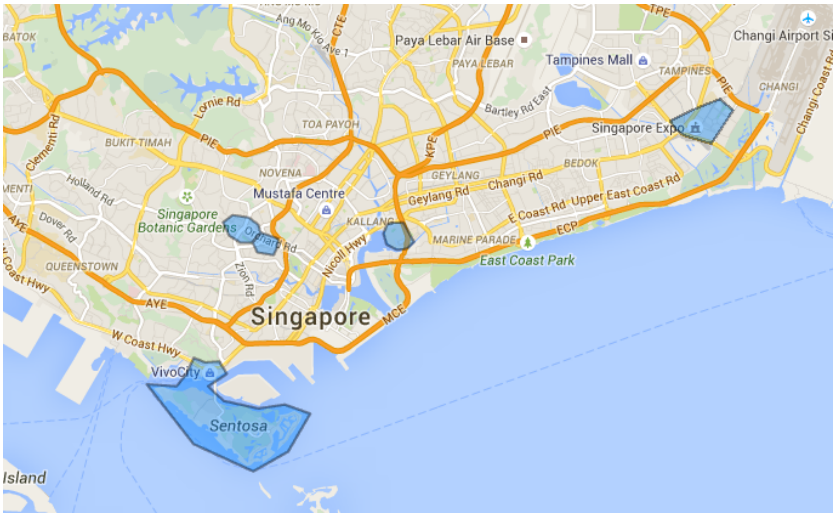


Figure 6.2: Map of Singapore, with the study areas

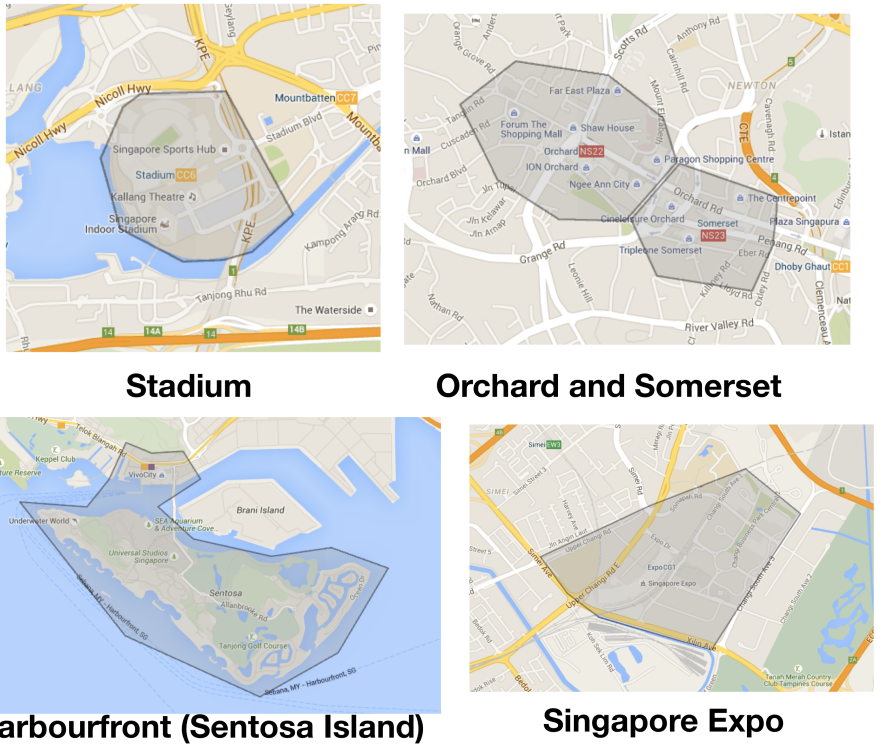


Figure 6.3: Map of Singapore, with the study areas

ending times when available) and short descriptions. For the current experiment, Eventful.org API was used to collect all the data. For training,

data from the same time window as our EZLink dataset was collected. This dataset is continuously increased on a daily basis.

This events database is, then, used for the query generation step. Web search engines provide opportunity to ask questions in form of queries to hopefully get some relevant information. Therefore, for each event of the database, it is important to decide what questions to ask (information retrieval) and how to interpret the answers (information extraction from the search results) in order to maximize prediction power of extracted features. In this case study, sophisticated information retrieval techniques were not touched, only two simple queries were used: event title and event venue name. Also, only basic information from the search results like titles, snippets (the short descriptions that appear below the title of every search result) and URLs was extracted. Finally, for each event, the two used queries were run in Bing.com search engine and their resultant titles and text snippets were aggregated into a single document to train MedLDA.

6.3.2 Data preparation

In order to get the best out of the collected data, it had to be prepared in multiple ways. There were choices to be made regarding the problem of having simultaneous events, the definition of hotspot and, finally, events lists and textual data need to go through cleaning procedures.

Simultaneous events, i.e., events sufficiently close in time and space, can create complex interactions, as discussed in Pereira et al. (2015). From a data modeling perspective, there is the challenge of putting together different features from each event (e.g. categories, start times, Facebook likes). In Pereira et al. (2015), an additive model was created, where each event is separately represented, allowing to separate the total demand into individual demand sources, including individual events and the routine component. In this work, we decided to take a simpler, fully non-parametric, approach where we stack together the query results of multiple events.

Similarly, we had to determine under what circumstances a particular period in a specific area would be declared a hotspot. Thus, for each event in the database, the EZLink arrivals time series for the bus and subway stops that were (manually) associated to that venue were identified. We focused on the period of 2 half-hours before the start time and the half-hour after. If, at any of such period, the total demand exceeded the 80th percentile, it would be defined as a hotspot. Figure 6.4 illustrates this concept.

Expectedly, using this definition there are much more non-hotspot instances than positive ones, which could lead to the generation of flawed

classifications. An unbalanced label set could cause the model's tendency to always predict the most common class, which, in this case, means generating many false negative cases. Since, for a transit agency, a false negative is a much more costly error than a false positive case, we adjusted the dataset by repeating every positive input vector 7 times in the training set, so that both positive and negative instances would be in the same proportion. This way, the model is able to learn to distinguish the two classes fairly. Notice that, at the test set side, we keep the original proportions, so the metrics we use will still reflect the quality of the model on a realistic setting.

Another problem of our definition of hotspot is that events that happen more than once may represent a hotspot one day but not in the other days. The existence of multiple instances with the same textual input vector but different target values could "confuse" the model. Hence, all of these contradictory instances were discarded, since we can not define their ground truth labels.

Table 6.1 summarizes the statistics for study areas, event venues and hotspots.

	Expo	**Harbourfront**	**Orchard**	**Somerset**	**Stadium**
Venues	5	27	54	26	7
Events	266	265	116	77	26
Hotspots	39	16	8	3	19
Mean number of hotspots per day ± stddev. *	0.41± 1.00	0.13 ± 0.64	0.17± 0.56	0.067± 0.25	0.95± 0.80

Table 6.1: Descriptive statistics for events and hotspots database.

∗ Only counting with days with events.

As for the events lists, even though they are generally well structured and clean, there was a verification of the possible errors in venue spatial coordinates as well as in temporal tags, particularly the start/end times. These errors were, then, manually corrected. This approach is scalable for spatial corrections (only once per venue), but harder for temporal tags. In prediction mode at a transit agency, an approach may be to spot suspicious temporal mistakes and have a simple interface to correct them.

Finally, in order to proceed with topic modeling, preprocessing of available textual information was the first crucial step. For titles and snippets (from the web search results), we followed the standard natural language

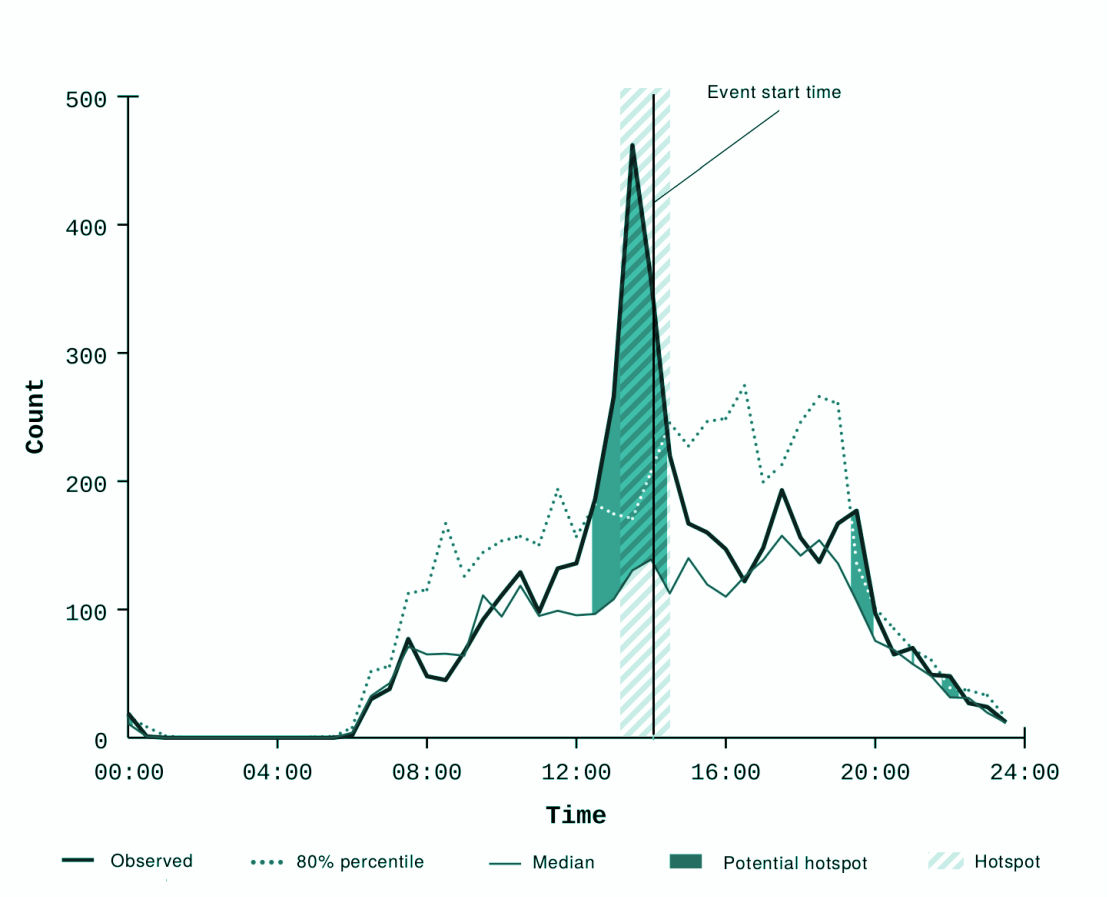


Figure 6.4: Arrivals data for a specific day.

processing procedure: kept only alphabetic symbols, removed stop and short words (less that 3 symbols), snowball stemming, removed words that appear only once. We also extended this approach using Wikipedia API calls for particular lexical patterns (Noun + Noun, etc).

6.4 Experimental design

The results of our model depended on the following decisions: the selection of query templates has a direct implication on the documents for the topic model, since it is the only input in the model, and the classification mechanism parameters. MedLDA has several parameters, namely α, l (penalty term for misclassifications) and C (penalty for soft margin slack variables). These are user given parameters and there is no obvious intuition for their

values, which means that an experiment evaluation was needed to choose their values.

For choosing the queries, we first tested the model with just the search results of the query with the title of the event. Since the type of events is not uniform across all venues, we then compared the prediction accuracies of using just that query and the results combining it with the query related to the venues' names. We verified that using the aggregation of the two queries' outputs, we achieved better predictions.

The choice of MedLDA parameters followed a grid search methodology, essentially by selecting a discrete set of values to try and choosing the best combination. We tested with $\alpha \in \{0.0001/K, 0.001/K, 0.01/K, 0.1/K\}$, $l \in \{1, 5, 8, 10, 12, 15, 20\}$ and $C \in \{1, 10, 20, 30, 50\}$. The best resulting combination was $\alpha = 0.001/K$, $l = 10$ and $C = 30$.

We will first analyse our results in terms of their interpretability. We will discuss the topics extracted, particularly, their word distributions, β_k, and their individual influence in the classification task, as represented by the η vectors. Then, we will compare with a baseline model that does not apply supervised topic models. It is essentially the typical two-stage model where we determine the topics through LDA Blei et al. (2003b) and then train a classifier (we chose and SVM for better comparability with MedLDA). The comparison will be based on three measures: accuracy, F1-score and κ statistic. Since accuracy is just the percentage of correctly classified instances (Equation 6.4), it could be misleading as the test set is disproportionate in the volume of the two classes. Therefore, the predictions' F1-score and κ statistics were analysed. F1-score is given by:

$$F1 = \frac{2TP}{2TP + FP + FN}, \tag{6.2}$$

where TP is the number of positive instances correctly classified, FP is the number of false alarms, in this case, the number of non-hotspots classified as hotspots and FN is the number of positive instances predicted as negative.

The Kappa statistic is defined as:

$$\kappa = \frac{OA - RA}{1 - RA}, \tag{6.3}$$

where OA is the observed accuracy calculated as:

$$OA = \frac{TP - TN}{T}, \tag{6.4}$$

and RA is the random accuracy defined as:

$$RA = \frac{(TN + FP)(TN + FN)(FN + TP)(FP + TP)}{T^2}, \qquad (6.5)$$

where TP, FP and FN follow the same notation used in F1-score, TN are the right classified negative cases and T is the total number of instances.

F1-score is an important metric since it relates the number of well classified positive instances with the quantity of the missed ones. The Kappa statistic, in turn, gives us a comparison between the performances of our classification system and a random one. In other words, it confronts the observed accuracy obtained by the model with the accuracy that any random classifier would be expected to achieve based on the confusion matrix. That is, if there are 20% of hotspots, a random classifier would probably predict 20% of positive instances and 80% of negative. We will see in the next section that these two metrics reveal clearly the quality of the tested approaches.

We applied 10-fold cross validation and we used all events and hotspots mentioned in Table 6.1.

6.5 Results

The topics generated by MedLDA are shown in Figure 6.5 along with their assigned η values (etas). It can be seen that there are two dominant topics: the first and the eighth. These are clearly the topics that have the largest contribution in the classification of the events as hotspots or not. The remaining seven topics are, still, helpful in this discrimination in the decision surface's boundaries, as our analysis revealed.

We evaluated the model using 2, 6, 7, 8, 10, 20, 30 and 50 topics in terms of its prediction quality as well as its topics interpretability. We started by favoring accuracy, by trying big values of K. However, a large K would make it difficult to comprehend the outcome of the model. Since we want to study the relationship between the word distributions of the topics and their individual contribution for the classification process, this would be a drawback. Our goal is to understand the topics underlying explanation of what makes an event a hotspot or not.

Therefore, K was decreased until a good trade-off between the classification performance and the topics understandability was achieved. We observed that, for K values higher than 10, the prediction improvements were marginal and that small values of K resulted in low F1-scores and κ statistics values.

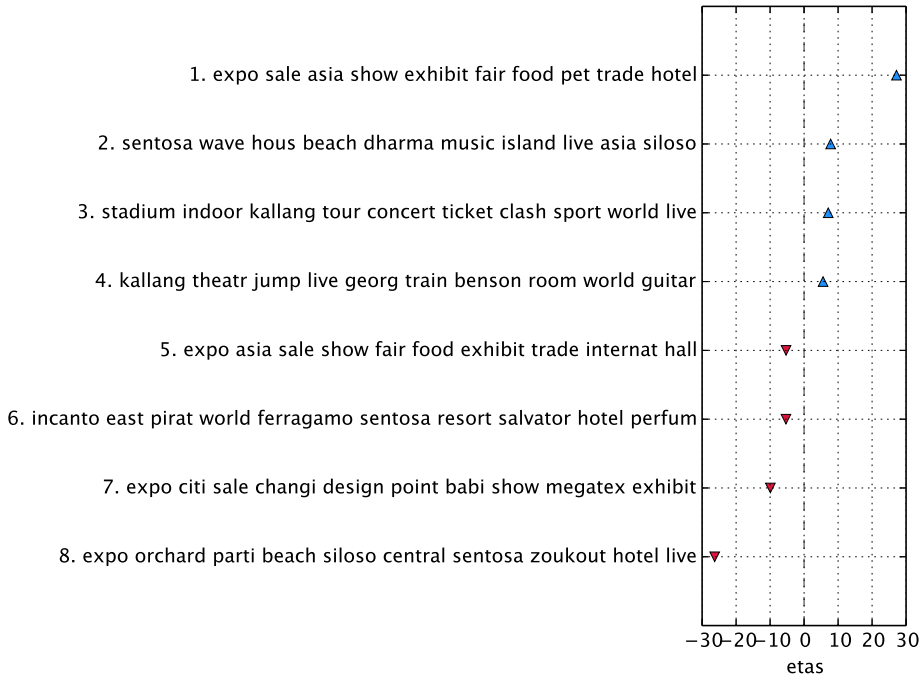
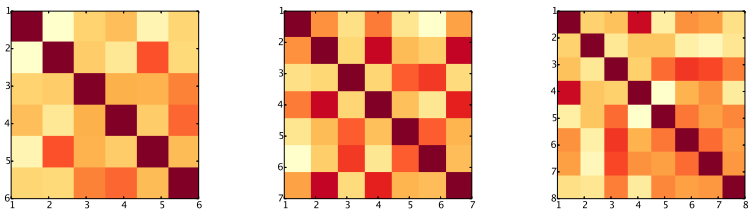


Figure 6.5: Topics and their assigned η's.

Also, we verified that using more than 6 topics resulted in groups of topics highly correlated. Figure 6.6 exhibits this correlation using $K \in \{6, 7, 8\}$, in which the darkest colors represent the highest correlations. Ideally, there would not be correlated topics, since a high similarity between groups of topics could mean that there is multicollinearity. Multicollinearity, in this case, would be having correlated values of η (our predictors), which would affect our interpretation of the model's outcome. Nevertheless, the difference between the F1-scores using 7 (F1-score = 0.6369) or 8 topics (F1-score = 0.6597) justified our choice for $K = 8$.



(a) K = 6 (b) K = 7 (c) K = 8

Figure 6.6: Correlation matrices resultant of applying different values of K.

Figure 6.7: Word clouds of topics.

Figure 6.7 illustrate the topics' words and their strength inside the topic. The biggest words are the words with higher probability in that topic. Moreover, the word clouds with more vivid colors represent the topics more "predictive" and, while the blue topics on the left match the positive η's, the others correspond to the negative ones (this relation is also shown in Figure 6.5).

It stands out that the word "expo" is a heavy word in four topics, a phenomenon that can be associated with the correlations between the topics that are depicted in Figure 6.6c. However, this also can be explained by the fact that Expo is a very heterogeneous area, with a high percentage of hotspots per day (see Table 6.1). In fact, topics reveal that "expo" can be related to very distinct contexts: food, design, sale, pet and Megatex, which is an electronics exposition.

We can also see that there are words clearly discriminative: events in Sentosa, in the Indoor Stadium or in the Kallang Theatre are quite likely to represent a hotspot. In fact, Stadium and Kallang Theatre belong to

the Stadium zone, which have an average of 95% hotspots per day. On the other hand, events in Ochard, which is mainly a shopping area, are more representative of non-hotpots.

	Mean accuracy	**Mean F1-score**	**Mean κ statistic**
LDA + SVM	0.9040	0.2583	0.2427
MedLDA	0.9030	0.6597	0.6053

Table 6.2: Comparison between LDA+SVM and MedLDA results.

Finally, by analysing the prediction results of both MedLDA's and LDA+SVM's methods, it becomes evident that our classifier outperforms the LDA+SVM approach, proving that MedLDA was a better option than the two-step procedure. Even though the accuracies are very similar, the contrast between the two is well manifested by the κ statistic values. While our approach is approximately 61% better than a random classifier, the LDA+SVM's method demonstrates to have a high tendency to predict the most popular class: non-hotspot. These are results for just a single run, since the effect of stochasticity of both models was verified to be insignificant. That is, the F1-scores obtained in 30 runs were the same, probably due to the binary nature of this problem and the small size of the testsets (10% of the data is just 75 events).

These results show that iOracle can predict the correct hotspots with a F1-score of 65.97% using only web search query content as input for the model. Notice that one can possibly further improve the quality of this classifier with other features, like type of day (e.g. weekend/weekday), time of day, event ticket price when available or any other information that is available. The unique advantage of our contribution is that it provides topics that are much more discriminative than a typical two-stage process.

Chapter 7

Work plan

This chapter describes the first and second semester activities. The initial objectives outlined are compared to the ones achieved and the differences between the two are justified. The first section is related to the first semester and the second one to the second semester.

7.1 First semester

The first semester started with the study of the state of the art topics covered in this thesis. Themes such as variational inference, topic models and learning from crowds were explored so that the necessary knowledge to execute the required tasks of this thesis was acquired. This included a discussion about the existent topic models, especially focusing on the supervised approaches, and learning from crowds methods, from which can be highlighted the supervised latent Dirichlet allocation (sLDA) algorithm proposed by Blei & McAuliffe (2007) and the framework described in Raykar et al. (2009).

Beyond the study of the state of the art, the objectives to this semester were to develop the variational inference algorithm for the model, to implement it and, of course, the writing of the intermediate report. All of them were accomplished. Moreover, the implemented model was validated and evaluated in a set of diversified experiments that included real and simulated annotators and text and image datasets. Since these tests and experiences involved data with annotations, it was necessary to collect labeled textual data, to process it using natural language processing techniques, to simulate annotations based on previously defined annotators' accuracies and to use Amazon Mechanical Turk to obtain real annotations. These real annotations were important, not only to assess the behavior of the model with truthful data, but also to evaluate the real labelers' expertise and biases.

The first version of the multi-annotator supervised latent Dirichlet allocation implemented (described in Appendix C) computed the topics and the annotators reliability parameters $\boldsymbol{\beta}$ and $\boldsymbol{\pi}$ using maximum likelihood estimates. After this simpler variant of the model have been validated, in the last weeks of the first semester, a fully Bayesian approach was designed. By "fully Bayesian", we mean that there are, now, two new priors: one over the topics $\boldsymbol{\beta}$ and another over the annotators' confusion matrices $\boldsymbol{\pi}$. In other words, instead of finding just the optimal values for these parameters, such that the likelihood is maximized, a distribution for each of them is obtained. In this way, it is performed Bayesian inference in the topics and the annotators quality parameters to produce smooth posteriors and to control sparsity.

Besides this conceptual improvement, a more scalable variational inference algorithm (using stochastic variational inference) was started to be developed. It was further implemented and tested in the beginning of the second semester.

Finally, in the first semester, the writing of a scientific article intended to be submitted in the International Conference on Machine Learning (ICML) [1] was also started.

7.2 Second semester

Beyond the continuation of the unfinished three tasks started in the first semester: the development and implementation of the stochastic variational inference algorithm, the model redesigning to include two new priors and the writing of scientific article, some main challenges were considered to the second semester. Those were:

- **Application of the developed model to the problem of event classification**: Using the description of major social events such as music concerts or football matches extracted from online event sources, the idea was to study the relation between those descriptions and the impact of the events, in order to predict valuable information like popularity indicators, attendance or even to distinguish relevant events of spam. The main goal was to identify future demand disruptions related to social events in public transports;

- **Generalization of the developed model to regression problems**: The first implemented model was adapted to classification problems,

[1] http://icml.cc/2015

however, it could be useful to predict continuous response variables, instead of a discrete set of classes. Therefore, it was defined as a challenge for the second semester the extension of the model to regression problems;

- **Generalization of the developed model to multi-label problems**: MA-sLDA was initially developed for single-label problems, hence, another generalization that was intended to be made was to take into account data with multiple labels. Multi-label data is not equivalent to multi-annotator data, since multi-label means having more than one label per instance, while in multi-annotation problems there are annotations given by several labelers and the knowledge about who provided the labels is crucial;

- **Writing of a second scientific article**;

- **Writing of the thesis**.

Naturally, the unconcluded tasks of the first semester started the work of the second one. The new version of the MA-sLDA and the new variational inference algorithm were implemented, validated with the set of experiments demonstrated in Chapter 5 and the results were analysed. Then, a paper intended to be submitted to ICML was written presenting the new fully Bayesian version of MA-sLDA including stochastic variational inference and reporting the improvements obtained comparing to the related state of the art approaches. This article was rejected in ICML, but latter accepted in AAAI HCOMP2015 conference [2] and it is presented in Appendix E.

The next step was to design MA-sLDA for regression problems. The development phase was assumed to be similar to the same process for the classification model: to study the code that would be the base for our model and, after that, to start extending it to account for multiple annotators. Unfortunately, the sLDA code for regression did not performed as well as it was supposed to. The experiences specified in Blei & McAuliffe (2007) were replicated, yet, the prediction outcomes were far worse than the ones showed in the paper. Since the author of the code did not give us any hint of what could be the problem, the solution was to modify the classification variant of sLDA code already studied in the first semester in order to adapt it for prediction of continuous values. This was followed by the unavoidable validation of the model.

[2] http://www.humancomputation.com/2015/

When we managed to reproduce the accuracy results of sLDA for regression documented in Blei & McAuliffe (2007), we finally initiated the multi-annotator generalization. This was planned to be started earlier that it actually did, consequence of the unexpected difficulty on getting the sLDA for regression working.

At the time our proposed regression model got ready to be tested and when I was already working as an invited research scientist at Singapore-MIT Alliance for Research and Technology (SMART), some preparation of the Singapore Land Transport Authority (LTA)'s data was also initiated. The goal was to begin another planned task for the second semester: the application of the developed model to the problem of event classification. However, this plan also had a deviation from what it was outlined: since the LTA's data comprises real values of people's arrivals at bus/subway stops, i.e., the ground truth of our target variables, the generalization we proposed that is able to learn from crowds would not be fundamental. Hence, it was chosen the supervised topic model that empirically showed the best prediction performances and efficiency in that context: MedLDA.

Although this turned out to be a case study about MedLDA for human mobility prediction, MA-sLDA would fit very well in this problem if, instead of the data of Singapore LTA, annotated data was available. In fact, to extract crowdsourced data about public transportation travelers is the purpose of the Future Mobility Survey (FMS) (Pereira et al., 2013a). FMS is a smartphone application developed in the Singaporean context also as a collaborative project between the SMART and the Singapore LTA, but, it is still in a field testing stage and that was the reason why we used LTA's data instead.

The primary objective of this part of the work was, then, to build a model able to predict demand in special events by correlating internet search query data with real measurements of transport usage. It was a project of a five people's team in which the classification mechanism of the system (MedLDA) and the textual data preprocessing were my main responsibilities. Therefore, until the end of the second semester, I was focused on it alongside with the experimental evaluation of the MA-sLDA for regression.

With the contributions resultant of the work done in both semesters, two new articles were started to be written: "Learning Supervised Topic Models from Crowds", presenting both MA-sLDA for classification and regression, and "iOracle" introducing the method for human mobility prediction characterized in Chapter 6.

In conclusion, apart from the generalization of the developed model to multi-label problems defined as a goal for the second semester and the

application of MA-sLDA to the problem of event classification, all the tasks were accomplished as they were outlined.

Chapter 8

Conclusion

This thesis proposed a supervised topic model that is able to learn from multiple annotators and crowds, by accounting for their biases and different levels of expertise. Given the large sizes of modern datasets and considering that the majority of the tasks for which crowdsourcing and multiple annotators are desirable candidates, generally involve complex high-dimensional data such as text and images, the proposed model constitutes a strong contribution for the multi-annotator paradigm. Furthermore, an efficient stochastic variational inference algorithm was described, which gives the proposed models the ability to scale to large datasets.

Two distinct variants of multi-annotator supervised topic model proposed were developed, one for classification and another for regression. These models share similar intuitions but they inevitably differ due the nature of the target variables. Both of them are capable of jointly modeling the words in documents as arising from a mixture of topics, as well as the latent true target variables and the (noisy) answers of the multiple annotators. We empirically showed, using both simulated and real annotators from Amazon Mechanical Turk that the proposed model is able to outperform state-of-the-art approaches in several real-world problems, such as classifying posts, news stories and images or predicting the number of stars of a restaurant based on its reviews. For this, we used various popular datasets from the state of the art, that are commonly used for benchmarking machine learning algorithms.

Also, we included a case study about a real-world application of supervised topic models, by proposing iOracle. iOracle predicts the occurrence of public transport overcrowding hotspots using minimal data (event title, date and location) obtained from a feed of event listings and a set of automated search queries. This work was developed in collaboration with the Singapore Land Transport Authority and aims to answer the concrete chal-

lenge of transport planning. The presented approach is novel in the sense that the only input for the model is search query content and, since a supervised topic model is applied, we can study the query content that is more relevant for hotspot prediction. By using such a simple input structure, the proposed method is easily portable from city to city and even domain (e.g. stock market).

The experimental evaluation of iOracle revealed that our method successfully classifies the overcrowded events with a κ statistic value of 60.53% and a F1-score of 65.97%. Furthermore, the experiments' results confirmed the power of supervised topic models when compared to separated topic modeling and classifier procedures.

With the contribution of the work performed in this thesis, one publication was accepted in AAAI HCOMP2015[1] (Appendix E) and another two publications are about to be submitted. One, describing both multi-annotator supervised LDA for classification and for regression, will be submitted to IEEE T-PAMI [2] and an article presenting iOracle has as target Nature Scientic Reports [3].

Future work will explore the extension of the multi-annotator supervised topic model proposed to multi-label classification problems and the enrichment of iOracle with other features.

[1] http://www.humancomputation.com/2015
[2] http://www.computer.org/web/tpami
[3] http://www.nature.com/srep/index.html

References

Ben-Akiva, M. and Lerman, S. *Discrete Choice Analysis: Theory and Application to Travel Demand*. The MIT Press, 1987.

Bernardo, J., Bayarri, M., Berger, J., Dawid, A., Heckerman, D., Smith, A., West, M., et al. The variational bayesian em algorithm for incomplete data: with application to scoring graphical model structures. *Bayesian statistics*, 7:453–464, 2003.

Bishop, C. Variational learning in graphical models and neural networks. In Niklasson, Lars, Bodn, Mikael, and Ziemke, Tom (eds.), *ICANN 98*, Perspectives in Neural Computing, pp. 13–22. Springer London, 1998. ISBN 978-3-540-76263-8.

Bishop, C. et al. *Pattern recognition and machine learning*, volume 1. Springer New York, 2006.

Blei, D. Probabilistic topic models. *Communications of the ACM*, 55(4): 77–84, 2012.

Blei, D. and McAuliffe, J. Supervised topic models. *Neural Information Processing Systems*, 2007.

Blei, D., Ng, A., and Jordan, M. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003a. ISSN 1532-4435.

Blei, D., Ng, A., and Jordan, M. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003b. ISSN 1532–4435.

Bonferroni, C. *Il calcolo delle assicurazioni su gruppi di teste*. Tipografia del Senato, 1935.

Born, K., Yasmin, S., You, D., Eluru, N., Bhat, C., and Pendyala, R. Joint model of weekend discretionary activity participation and episode duration. *Transportation Research Record: Journal of the Transportation Research Board*, (2413):34–44, 2014.

Calabrese, F, Pereira, F., Lorenzo, G., Liu, L., and Ratti, C. The geography of taste: analyzing cell-phone mobility and social events. In Floréen, Patrik, Krüger, Antonio, and Spasojevic, Mirjana (eds.), *Pervasive Computing*, volume 6030, pp. 22–37. LNCS, Springer, 2010.

Chang, M. and Lu, P. A multinomial logit model of mode and arrival time choices for planned special events. *J. of the Eastern Asia Society for Transportation Studies*, 10:710–727, 2013.

Dawid, A. and Skene, A. Maximum likelihood estimation of observer errorrates using the em algorithm. *Applied statistics*, pp. 20–28, 1979.

Dekel, O. and Shamir, O. Good learners for evil teachers. In *Proceedings of the 26th annual international conference on machine learning*, pp. 233–240. ACM, 2009.

Dempster, A., Laird, N., and Rubin, D. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38, 1977.

Donmez, P. and Carbonell, J. Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pp. 619–628. ACM, 2008.

Dredze, M., Talukdar, P., and Crammer, K. Sequence learning from data with multiple labels. In *Workshop Co-Chairs*, pp. 39. Citeseer, 2009.

Fei-Fei, L. and Perona, P. A bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pp. 524–531. IEEE, 2005.

Groot, P., Birlutiu, A., and Heskes, T. Learning from multiple annotators with Gaussian processes. In *Proc. of the 21st Int. Conf. on Artificial Neural Networks*, volume 6792, pp. 159–164, 2011.

Hoffman, M., Blei, D., Wang, C., and Paisley, J. Stochastic variational inference. *J. Mach. Learn. Res.*, 14(1):1303–1347, May 2013. ISSN 1532-4435.

Huang, Y. and Suen, C. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(1):90–94, Jan 1995. ISSN 0162-8828.

Jingbo, Y., Tiexin, C., and Miaomiao, T. Demand forecasting of parking lot based on discrete choice model in planned special events. In *Management and Service Science, 2009. MASS '09. International Conference on*, pp. 1–4, Sept 2009. doi: 10.1109/ICMSS.2009.5304335.

Kadir, T. and Brady, M. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

Kuppam, A., Copperman, R., Rossi, T., Livshits, V., Vallabhaneni, L., Brown, T., and DeBoer, K. Innovative methods for collecting data and for modeling travel related to special events. *Transp. Res. Record: J. of the Transportation Research Board*, 2246(1):24–31, 2011.

Kwoczek, S., Martino, S., and Nejdl, W. Predicting and visualizing traffic congestion in the presence of planned special events. *Journal of Visual Languages and Computing*, 25(6):973 – 980, 2014. ISSN 1045-926X. doi: http://dx.doi.org/10.1016/j.jvlc.2014.10.028. URL `http://www.sciencedirect.com/science/article/pii/S1045926X14001219`. Distributed Multimedia Systems {DMS2014} Part I.

Lacoste-julien, S., Sha, F., and Jordan, M. Disclda: Discriminative learning for dimensionality reduction and classification. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L. (eds.), *Advances in Neural Information Processing Systems 21*, pp. 897–904. Curran Associates, Inc., 2009.

Lei-Lei, D., Jin-Gang, G., Zheng-Liang, S., and Hong-Tong, O. Study on traffic organization and management strategies for large special events. In *System Science and Engineering (ICSSE), 2012 International Conference on*, pp. 432–436, June 2012. doi: 10.1109/ICSSE.2012.6257222.

Lewis, D. Reuters-21578 text categorization test collection, distribution 1.0. 1997.

Li, L., Roth, B., and Sporleder, C. Topic models for word sense disambiguation and token-based idiom detection. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pp. 1138–1147, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

Lowe, D. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pp. 1150–1157. Ieee, 1999.

Mauá, D. and Cozman, F. Representing and classifying user reviews, 2009.

Murphy, K. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Passonneau, R., Salleb-Aouissi, A., Bhardwaj, V, and Ide, N. Word sense annotation of polysemous words by multiple annotators. In *LREC*, 2010.

Pereira, F., Carrion, C., Zhao, F., Cottrill, C., Zegras, C., and Ben-Akiva, M. The future mobility survey: Overview and preliminary evaluation. In *Proceedings of the Eastern Asia Society for Transportation Studies*, volume 9, 2013a.

Pereira, F., Rodrigues, F., and Ben-Akiva, M. Using data from the web to predict public transport arrivals under special events scenarios. *J. of Intelligent Transportation Systems*, 2013b.

Pereira, F., Rodrigues, F., Polisciuc, E., and Ben-Akiva, M. Why so many people? explaining nonhabitual transport overcrowding with internet data. *Intelligent Transportation Systems, IEEE Transactions on*, 16(3):1370–1379, June 2015. ISSN 1524-9050. doi: 10.1109/TITS.2014.2368119.

Putthividhy, D., Attias, T., and Nagarajan, S. Topic regression multi-modal latent dirichlet allocation for image annotation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 3408–3415, June 2010.

Ramage, D., Hall, D., Nallapati, R., and Manning, C. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pp. 248–256, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-59-6.

Raykar, V., Yu, S., Zhao, L., Jerebko, A., Florin, C., Valadez, G., Bogoni, L., and Moy, L. Supervised learning from multiple experts: whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual international conference on machine learning*, pp. 889–896. ACM, 2009.

Robbins, H. and Monro, S. A stochastic approximation method. *The annals of mathematical statistics*, pp. 400–407, 1951.

Rodrigues, F., Pereira, F., and Ribeiro, B. Learning from multiple annotators: Distinguishing good from random labelers. *Pattern Recognition Letters*, 34(12):1428–1436, 2013.

Rubin, T., Chambers, A., Smyth, P., and Steyvers, M. Statistical topic models for multi-label document classification. *Machine Learning*, 88(1-2):157–208, 2012. ISSN 0885-6125.

Russell, B., Torralba, A., Murphy, K., and Freeman, W. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008. ISSN 0920-5691.

Sall, E. and Bhat, C. An analysis of weekend work activity patterns in the san francisco bay area. *Transportation*, 34(2):161–175, 2007. ISSN 0049-4488. doi: 10.1007/s11116-006-0008-2. URL `http://dx.doi.org/10.1007/s11116-006-0008-2`.

Shahin, S., Hseyin, T., and Kemal, O. Evaluating transportation preferences for special events: A case study for a megacity, istanbul. *Procedia - Social and Behavioral Sciences*, 111(0):98 – 106, 2014. ISSN 1877-0428. doi: http://dx.doi.org/10.1016/j.sbspro.2014.01.042. URL `http://www.sciencedirect.com/science/article/pii/S1877042814000433`. Transportation: Can we do more with less resources? 16th Meeting of the Euro Working Group on Transportation Porto 2013.

Sheng, V., Provost, F., and Ipeirotis, P. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 614–622. ACM, 2008.

Smyth, P., Fayyad, U., Burl, M., Perona, P., and Baldi, P. Inferring ground truth from subjective labelling of venus images. *Advances in neural information processing systems*, pp. 1085–1092, 1995.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pp. 254–263. Association for Computational Linguistics, 2008.

Spiegelhalter, D. and Stovin, P. An analysis of repeated biopsies following cardiac transplantation. *Statistics in medicine*, 2(1):33–40, 1983.

Steyvers, M. and Griffiths, T. Probabilistic topic models. *Handbook of latent semantic analysis*, 427(7):424–440, 2007.

Taddy, M. Multinomial inverse regression for text analysis. *Journal of the American Statistical Association*, 108(503):755–770, 2013.

Wang, C., Blei, D., and Li, F-F. Simultaneous image classification and annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1903–1910, June 2009.

Wright, S. and Nocedal, J. *Numerical optimization*, volume 2. Springer New York, 1999.

Yan, Y., Rosales, R., Fung, G., Schmidt, M., Valadez, G., Bogoni, L., Moy, L., and Dy, J. Modeling annotator expertise: Learning when everybody knows a bit of something. In *International conference on artificial intelligence and statistics*, pp. 932–939, 2010.

Zeno, S., Duvvuri, R., and Millard, R. *The educator's word frequency guide*. Touchstone Applied Science Associates Brewster, NY, 1995.

Zhu, J., Ahmed, A., and Xing, E. Medlda: Maximum margin supervised topic models for regression and classification. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 1257–1264, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1.

Appendices

Appendix A

Inference and parameter estimation in the classification model

In this appendix, a detailed version of Sections 3.2 and 3.3 is presented.

A.1 Derivation of the terms in the lower bound

$$\begin{aligned}\mathbb{E}_q[\log p(\beta_i|\tau)] &= \mathbb{E}_q\left[\log \frac{\Gamma(\sum_{k=1}^{V}\tau_k)}{\prod_{j=1}^{V}\Gamma(\tau_j)}\prod_{j=1}^{V}\beta_{i,j}^{(\tau_j-1)}\right] \\ &= \log\Gamma\left(\sum_{k=1}^{V}\tau_k\right) - \sum_{j=1}^{V}\log\Gamma(\tau_j) + \sum_{j=1}^{V}(\tau_j-1)\mathbb{E}_q[\log\beta_{i,j}].\end{aligned} \tag{A.1}$$

$$\begin{aligned}\mathbb{E}_q[\log p(\pi_c^r|\omega)] &= \mathbb{E}_q\left[\log \frac{\Gamma(\sum_{t=1}^{C}\omega_t)}{\prod_{l=1}^{C}\Gamma(\omega_l)}\prod_{l=1}^{C}\left(\pi_{c,l}^r\right)^{(\omega_l-1)}\right] \\ &= \log\Gamma\left(\sum_{t=1}^{C}\omega_t\right) - \sum_{l=1}^{C}\log\Gamma(\omega_l) + \sum_{l=1}^{C}(\omega_l-1)\mathbb{E}_q[\log\pi_{c,l}^r].\end{aligned} \tag{A.2}$$

$$\mathbb{E}_q[\log p(\theta^d|\alpha)] = \mathbb{E}_q\bigg[\log \frac{\Gamma(\sum_{j=1}^K \alpha_j)}{\prod_{i=1}^K \Gamma(\alpha_i)} \prod_{i=1}^K (\theta_i^d)^{(\alpha_i-1)}\bigg]$$
$$= \log\Gamma\bigg(\sum_{j=1}^K \alpha_j\bigg) - \sum_{i=1}^K \log\Gamma(\alpha_i) + \sum_{i=1}^K (\alpha_i - 1)\mathbb{E}_q[\log\theta_i^d]. \tag{A.3}$$

$$\mathbb{E}_q[\log p(z_n^d|\theta^d)] = \mathbb{E}_q\bigg[\log \prod_{i=1}^K \left(\theta_i^d\right)^{z_{n,i}^d}\bigg] = \sum_{i=1}^K \mathbb{E}_q[z_{n,i}^d]\mathbb{E}_q[\log\theta_i^d]$$
$$= \sum_{i=1}^K \phi_{n,i}^d \mathbb{E}_q[\log\theta_i^d]. \tag{A.4}$$

$$\mathbb{E}_q[\log p(w_n^d|z_n^d,\boldsymbol{\beta})] = \mathbb{E}_q\bigg[\log \prod_{j=1}^V (\beta_{z_n^d,j})^{w_{n,j}^d}\bigg] = \sum_{j=1}^V \mathbb{E}_q[w_{n,j}^d]\mathbb{E}_q[\log\beta_{z_n^d,j}]$$
$$= \sum_{j=1}^V w_{n,j}^d \mathbb{E}_q[\log\beta_{z_n^d,j}] = \sum_{j=1}^V w_{n,j}^d \mathbb{E}_q\bigg[\sum_{i=1}^K z_{n,i}^d \log\beta_{i,j}\bigg]$$
$$= \sum_{j=1}^V \sum_{i=1}^K w_{n,j}^d \phi_{n,i}^d \mathbb{E}_q[\log\beta_{i,j}]. \tag{A.5}$$

$$\mathbb{E}_q[\log p(y^{d,r}|c^d,\boldsymbol{\pi}^r)] = \mathbb{E}_q\bigg[\log \prod_{l=1}^C (\pi_{c^d,l}^r)^{y_l^{d,r}}\bigg] = \mathbb{E}_q\bigg[\sum_{l=1}^C y_l^{d,r}\log\pi_{c^d,l}^r\bigg]$$
$$= \sum_{c=1}^C \lambda_c^d \sum_{l=1}^C \mathbb{E}_q[y_l^{d,r}]\mathbb{E}_q[\log\pi_{c,l}^r] = \sum_{c=1}^C\sum_{l=1}^C \lambda_c^d y_l^{d,r}\mathbb{E}_q[\log\pi_{c,l}^r]. \tag{A.6}$$

$$\mathbb{E}_q[\log q(\pi_c^r|\xi_c^r)] = \mathbb{E}_q\bigg[\log \frac{\Gamma(\sum_{t=1}^C \xi_{c,t}^r)}{\prod_{l=1}^C \Gamma(\xi_{c,l}^r)} \prod_{l=1}^C \left(\pi_{c,l}^r\right)^{(\xi_{c,l}^r-1)}\bigg]$$
$$= \log\Gamma\bigg(\sum_{t=1}^C \xi_{c,t}^r\bigg) - \sum_{l=1}^C \log\Gamma(\xi_{c,l}^r) + \sum_{l=1}^C (\xi_{c,l}^r - 1)\mathbb{E}_q[\log\pi_{c,l}^r]. \tag{A.7}$$

$$\mathbb{E}_q[\log q(\beta_i|\zeta_i)] = \mathbb{E}_q\bigg[\log \frac{\Gamma(\sum_{k=1}^{V}\zeta_{i,k})}{\prod_{j=1}^{V}\Gamma(\zeta_{i,j})}\prod_{j=1}^{V}\left(\beta_{i,j}\right)^{(\zeta_{i,j}-1)}\bigg]$$

$$= \log\Gamma\bigg(\sum_{k=1}^{V}\zeta_{i,k}\bigg) - \sum_{j=1}^{V}\log\Gamma(\zeta_{i,j}) + \sum_{j=1}^{V}(\zeta_{i,j}-1)\mathbb{E}_q[\log\beta_{i,j}]. \tag{A.8}$$

$$\mathbb{E}_q[\log q(\theta^d|\gamma^d)] = \mathbb{E}_q\bigg[\log \frac{\Gamma(\sum_{j=1}^{K}\gamma_j^d)}{\prod_{i=1}^{K}\Gamma(\gamma_i^d)}\prod_{i=1}^{K}\left(\theta_i^d\right)^{(\gamma_i^d-1)}\bigg]$$

$$= \log\Gamma\bigg(\sum_{j=1}^{K}\gamma_j^d\bigg) - \sum_{i=1}^{K}\log\Gamma(\gamma_i^d) + \sum_{i=1}^{K}(\gamma_i^d-1)\mathbb{E}_q[\log\theta_i^d]. \tag{A.9}$$

$$\mathbb{E}_q[\log q(z_n^d|\phi_n^d)] = \mathbb{E}_q\bigg[\log\prod_{i=1}^{K}\left(\phi_{n,i}^d\right)^{z_{n,i}^d}\bigg] = \sum_{i=1}^{K}\mathbb{E}_q[z_{n,i}^d]\mathbb{E}_q[\log\phi_{n,i}^d]$$

$$= \sum_{i=1}^{K}\phi_{n,i}^d\log\phi_{n,i}^d. \tag{A.10}$$

$$\mathbb{E}_q[\log q(c^d|\lambda^d)] = \mathbb{E}_q\bigg[\log\prod_{l=1}^{C}\left(\lambda_l^d\right)^{c_l^d}\bigg] = \sum_{l=1}^{C}\mathbb{E}_q[c_l^d]\mathbb{E}_q[\log\lambda_l^d] = \sum_{l=1}^{C}\lambda_l^d\log\lambda_l^d. \tag{A.11}$$

$$\mathbb{E}_q[\log p(c^d|\bar{z}^d,\eta)] = \mathbb{E}_q\bigg[\log\frac{\exp(\eta_{c^d}^T\bar{z}^d)}{\sum_{l=1}^{C}\exp(\eta_l^T\bar{z}^d)}\bigg]$$

$$= \bigg(\mathbb{E}_q[\eta_{c^d}^T\bar{z}^d] - \mathbb{E}_q\bigg[\log\sum_{l=1}^{C}\exp(\eta_l^T\bar{z}^d)\bigg]\bigg). \tag{A.12}$$

The first term can be easily computed as:

$$\mathbb{E}_q[\eta_{c^d}^T\bar{z}^d] = \mathbb{E}_q\bigg[\sum_{j=1}^{K}\eta_{c^d,j}\bar{z}_j^d\bigg] = \sum_{l=1}^{C}\lambda_l^d\sum_{j=1}^{K}\eta_{l,j}\mathbb{E}_q[\bar{z}_j^d]$$

$$= \sum_{l=1}^{C}\lambda_l^d\sum_{j=1}^{K}\eta_{l,j}\frac{1}{N_d}\sum_{n=1}^{N_d}\mathbb{E}_q[z_{n,j}^d] = \frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\sum_{n=1}^{N_d}\sum_{j=1}^{K}\eta_{l,j}\phi_{n,j}^d$$

$$= \frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\sum_{n=1}^{N_d}\eta_l^T\phi_n^d. \tag{A.13}$$

Appealing to Jensens inequality, the second term is given by:

$$
\begin{aligned}
&- \mathbb{E}_q\Big[\log \sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d)\Big] \geqslant - \log \sum_{l=1}^{C} \mathbb{E}_q[\exp(\eta_l^T \bar{z}^d)] \\
&= - \log \sum_{l=1}^{C} \mathbb{E}_q\Big[\exp(\eta_l^T \frac{1}{N_d} \sum_{n=1}^{N_d} z_n^d)\Big] = - \log \sum_{l=1}^{C} \mathbb{E}_q\Big[\prod_{n=1}^{N_d} \exp(\eta_l^T \frac{1}{N_d} z_n^d)\Big] \\
&= - \log \sum_{l=1}^{C} \prod_{n=1}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right) \\
&= - \log \sum_{l=1}^{C} \left(\phi_j^d\right)^T \exp(\eta_l \frac{1}{N_d}) \prod_{n=1, n\neq j}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right) \\
&= - \log \left(\phi_j^d\right)^T \underbrace{\sum_{l=1}^{C} \exp(\eta_l \frac{1}{N_d}) \prod_{n=1, n\neq j}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right)}_{=h} \\
&= - \log \left(\phi_j^d\right)^T h = - \log h^T \phi_j^d. \qquad \text{(A.14)}
\end{aligned}
$$

where we defined $h = \sum_{l=1}^{C} \exp(\eta_l \frac{1}{N_d}) \prod_{n=1, n\neq j}^{N_d} \left(\phi_n^d\right)^T \exp\left(\eta_l \frac{1}{N_d}\right)$.

A.2 Optimizing the lower bound

A.2.1 Optimizing w.r.t. γ_i^d

Collecting only the terms in the bound that contain γ gives:

$$
\begin{aligned}
\mathcal{L}_{[\gamma]} = &\sum_{d=1}^{D} \sum_{i=1}^{K} (\alpha_i - 1) \left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K} \gamma_j^d\Big)\right) \\
&+ \sum_{d=1}^{D} \sum_{n=1}^{N_d} \sum_{i=1}^{K} \phi_{n,i}^d \left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K} \gamma_j^d\Big)\right) \\
&- \sum_{d=1}^{D} \left(\log \Gamma\Big(\sum_{j=1}^{K} \gamma_j^d\Big) - \sum_{i=1}^{K} \log \Gamma(\gamma_i^d)\right) \\
&- \sum_{d=1}^{D} \left(\sum_{i=1}^{K} (\gamma_i^d - 1) \left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K} \gamma_j^d\Big)\right)\right) \qquad \text{(A.15)}
\end{aligned}
$$

$$
\begin{aligned}
&= \sum_{d=1}^{D}\sum_{i=1}^{K}\left(\Psi(\gamma_i^d) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\right)\left(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\right) \\
&- \sum_{d=1}^{D}\log\Gamma\Big(\sum_{j=1}^{K}\gamma_j^d\Big) + \sum_{d=1}^{D}\sum_{i=1}^{K}\log\Gamma(\gamma_i^d) \\
&= \sum_{d=1}^{D}\sum_{i=1}^{K}\Psi(\gamma_i^d)\left(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\right) - \sum_{d=1}^{D}\sum_{i=1}^{K}\Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\left(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\right) \\
&- \sum_{d=1}^{D}\log\Gamma\Big(\sum_{j=1}^{K}\gamma_j^d\Big) + \sum_{d=1}^{D}\sum_{i=1}^{K}\log\Gamma(\gamma_i^d).
\end{aligned}
\tag{A.16}
$$

Taking derivatives w.r.t. γ_i^d gives:

$$
\begin{aligned}
\frac{\partial\mathcal{L}_{[\gamma]}}{\partial\gamma_i^d} &= \Psi'(\gamma_i^d)\Big(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\Big) - \Psi(\gamma_i^d) \\
&\quad - \Psi'\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\sum_{j=1}^{K}\Big(\alpha_j + \sum_{n=1}^{N_d}\phi_{n,j}^d - \gamma_j^d\Big) + \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big) - \Psi\Big(\sum_{j=1}^{K}\gamma_j^d\Big) + \Psi(\gamma_i^d) \\
&= \Psi'(\gamma_i^d)\Big(\alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d - \gamma_i^d\Big) - \Psi'\Big(\sum_{j=1}^{K}\gamma_j^d\Big)\sum_{j=1}^{K}\Big(\alpha_j + \sum_{n=1}^{N_d}\phi_{n,j}^d - \gamma_j^d\Big).
\end{aligned}
\tag{A.17}
$$

Setting this derivative to zero in order to get a maximum (notice that the solutions for the different γ_i^d are coupled, hence they have to be solved as a system of linear equations), we get the solution:

$$
\gamma_i^d = \alpha_i + \sum_{n=1}^{N_d}\phi_{n,i}^d, \tag{A.18}
$$

which can be easily verified by submitting the value for γ_i^d above in the expression for the partial derivatives.

A.2.2 Optimizing w.r.t. $\phi^d_{n,i}$

Collecting only the terms in the bound that contain $\phi^d_{n,i}$ and adding the necessary Lagrange multipliers gives:

$$\begin{aligned}
\mathcal{L}_{[\phi^d_{n,i}]} &= \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi^d_{n,i}\left(\Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right)\right) \\
&+ \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{j=1}^{V}\sum_{i=1}^{K} w^d_{n,j}\phi^d_{n,i}\left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right)\right) \\
&+ \sum_{d=1}^{D}\left(\frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\sum_{n=1}^{N_d}\eta_l^T\phi^d_n - \sum_{n=1}^{N_d}(h^T(\phi^d_n)^{old})^{-1}(h^T\phi^d_n)\right) \\
&\sum_{d=1}^{D}\left(-\sum_{n=1}^{N_d}\log(h^T(\phi^d_n)^{old}) + N_d\right) \\
&- \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{i=1}^{K}\phi^d_{n,i}\log\phi^d_{n,i} + \mu\left(\sum_{k=1}^{K}\phi^d_{n,k} - 1\right).
\end{aligned} \tag{A.19}$$

Taking derivatives w.r.t. $\phi^d_{n,i}$ gives:

$$\begin{aligned}
\frac{\partial\mathcal{L}_{[\phi^d_{n,i}]}}{\partial\phi^d_{n,i}} &= \Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right) + \sum_{j=1}^{V}w^d_{n,j}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V}w^d_{n,j}\Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right) \\
&+ \frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\eta_{l,i} - (h^T(\phi^d_n)^{old})^{-1}h_i - \log\phi^d_{n,i} - 1 + \mu.
\end{aligned} \tag{A.20}$$

Setting this derivative to zero and solving for $\phi^d_{n,i}$ gives:

$$\begin{aligned}
&\Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K}\gamma^d_j\right) + \sum_{j=1}^{V}w^d_{n,j}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V}w^d_{n,j}\Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right) \\
&+ \frac{1}{N_d}\sum_{l=1}^{C}\lambda^d_l\eta_{l,i} - (h^T(\phi^d_n)^{old})^{-1}h_i - \log\phi^d_{n,i} - 1 + \mu = 0
\end{aligned}$$

$$\Leftrightarrow \log \phi_{n,i}^d = \Psi(\gamma_i) - \Psi\Big(\sum_{j=1}^{K} \gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k}\Big)$$
$$+ \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,i} - (h^T (\phi_n^d)^{old})^{-1} h_i - 1 + \mu$$
$$\Leftrightarrow \phi_{n,i}^d = \exp\left(\Psi(\gamma_i) - \Psi\Big(\sum_{j=1}^{K} \gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k}\Big)\right.$$
$$\left. + \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,i} - (h^T (\phi_n^d)^{old})^{-1} h_i - 1 + \mu\right)$$
$$\Leftrightarrow \phi_{n,i}^d = \exp\left(\Psi(\gamma_i) - \Psi\Big(\sum_{j=1}^{K} \gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k}\Big)\right.$$
$$\left. + \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,i} - (h^T (\phi_n^d)^{old})^{-1} h_i - 1\right) \exp(\mu). \tag{A.21}$$

Plugging this expression in the constraint and solving for μ (or $\exp(\mu)$) gives:

$$\sum_{k=1}^{K} \phi_{n,k}^d = 1$$
$$\Leftrightarrow \sum_{k=1}^{K} \exp\left(\Psi(\gamma_k) - \Psi\Big(\sum_{j=1}^{K} \gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k}\Big)\right.$$
$$\left. + \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,k} - (h^T (\phi_n^d)^{old})^{-1} h_k - 1 + \mu\right) = 1$$
$$\Leftrightarrow \sum_{k=1}^{K} \exp\left(\Psi(\gamma_k) - \Psi\Big(\sum_{j=1}^{K} \gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k}\Big)\right.$$
$$\left. + \frac{1}{N_d} \sum_{l=1}^{C} \lambda_l^d \eta_{l,k} - (h^T (\phi_n^d)^{old})^{-1} h_k - 1\right) \exp(\mu) = 1$$

$$\Leftrightarrow \exp(\mu) = \frac{1}{\sum_{k=1}^{K} \exp\Big(\Psi(\gamma_k) - \Psi\big(\sum_{j=1}^{K}\gamma_j\big) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j})\Big)}$$
$$\times \frac{1}{-\sum_{j=1}^{V} w_{n,j}^{d}\Psi(\sum_{k=1}^{V}\zeta_{i,k}\Big) + \sum_{k=1}^{K}\exp\Big(\frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\eta_{l,k} - (h^T(\phi_n^d)^{old})^{-1}h_k - 1\Big)}. \tag{A.22}$$

Plugging this expression back in the expression for $\phi_{n,i}^d$ gives the solution:

$$\begin{aligned}
\phi_{n,i}^d &= \frac{\exp\Bigg(\Psi(\gamma_i) - \Psi\Big(\sum_{j=1}^{K}\gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j})\Bigg)}{\sum_{k=1}^{K}\exp\Big(\Psi(\gamma_k) - \Psi\big(\sum_{j=1}^{K}\gamma_j\big) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j})\Big)} \\
&\times \frac{\exp\Bigg(\frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\eta_{l,i} - (h^T(\phi_n^d)^{old})^{-1}h_i - 1 - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\Bigg)}{\sum_{k=1}^{K}\exp\Bigg(\frac{\sum_{l=1}^{C}\lambda_l^d\eta_{l,k}}{N_d} - (h^T(\phi_n^d)^{old})^{-1}h_k - 1 - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\Bigg)} \\
&\propto \exp\Bigg(\Psi(\gamma_i) - \Psi\Big(\sum_{j=1}^{K}\gamma_j\Big) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\Bigg) \\
&\times \exp\Bigg(\frac{\sum_{l=1}^{C}\lambda_l^d\eta_{l,i}}{N_d} - (h^T(\phi_n^d)^{old})^{-1}h_i\Bigg) \\
&\propto \exp\Bigg(\Psi(\gamma_i) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) + \frac{\sum_{l=1}^{C}\lambda_l^d\eta_{l,i}}{N_d}\Bigg) \\
&\exp\Bigg(-(h^T(\phi_n^d)^{old})^{-1}h_i\Bigg).
\end{aligned} \tag{A.23}$$

A.2.3 Optimizing w.r.t. λ_l^d

By collecting only the terms in the bound that contain λ_l^d and adding the necessary Lagrange multipliers, we have that:

$$\begin{aligned}
\mathcal{L}_{[\lambda_l^d]} &= \sum_{d=1}^{D}\left(\frac{1}{N_d}\sum_{l=1}^{C}\lambda_l^d\sum_{n=1}^{N_d}\eta_l^T\phi_n^d\right) \\
&+ \sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{l=1}^{C}\sum_{c=1}^{C}\lambda_l^d y_c^{d,r}\left(\Psi(\xi_{l,c}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big)\right) \\
&- \sum_{d=1}^{D}\sum_{l=1}^{C}\lambda_l^d\log\lambda_l^d + \mu\Big(\sum_{k=1}^{C}\lambda_k^d - 1\Big) \\
&= \sum_{d=1}^{D}\sum_{l=1}^{C}\lambda_l^d\eta_l^T\overline{\phi^d} + \sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{l=1}^{C}\sum_{c=1}^{C}\lambda_l^d y_c^{d,r}\left(\Psi(\xi_{l,c}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big)\right) \\
&- \sum_{l=1}^{C}\lambda_l^d\log\lambda_l^d + \mu\Big(\sum_{k=1}^{C}\lambda_k^d - 1\Big). \qquad \text{(A.24)}
\end{aligned}$$

where we defined $\overline{\phi^d} = \frac{1}{N_d}\sum_{n=1}^{N_d}\phi_n^d$.
Taking derivatives w.r.t. λ_l^d gives:

$$\frac{\partial\mathcal{L}_{[\lambda_l^d]}}{\partial\lambda_l^d} = \eta_l^T\overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big) - \log\lambda_l^d - 1 + \mu. \qquad \text{(A.25)}$$

Setting this derivative to zero and solving for λ_l^d gives:

$$\begin{aligned}
&\eta_l^T\overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big) - \log\lambda_l^d - 1 + \mu = 0 \\
&\Leftrightarrow \log\lambda_l^d = \eta_l^T\overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big) - 1 + \mu \\
&\Leftrightarrow \lambda_l^d = \exp\left(\eta_l^T\overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C}y_c^{d,r}\Psi\Big(\sum_{t=1}^{C}\xi_{l,t}^r\Big) - 1\right)\exp(\mu). \qquad \text{(A.26)}
\end{aligned}$$

Plugging this expression in the constraint and solving for μ (or $\exp(\mu)$) gives:

$$\begin{aligned}
&\sum_{k=1}^{K} \lambda_k^d = 1 \\
&\Leftrightarrow \sum_{k=1}^{K} \exp\left(\eta_k^T \overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right) - 1\right) \exp(\mu) = 1 \\
&\Leftrightarrow \exp(\mu) = \frac{1}{\sum_{k=1}^{K} \exp\left(\eta_k^T \overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r)\right)} \\
&\times \frac{1}{\sum_{k=1}^{K} \exp\left(-\sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right) - 1\right)}
\end{aligned} \tag{A.27}$$

Plugging this expression back in the expression for λ_l^d gives the solution:

$$\begin{aligned}
\lambda_l^d &= \frac{\exp\left(\eta_l^T \overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right)\right)}{\sum_{k=1}^{K} \exp\left(\eta_k^T \overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right)\right)} \\
&\propto \exp\left(\eta_l^T \overline{\phi^d} + \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi(\xi_{l,c}^r) - \sum_{r=1}^{R}\sum_{c=1}^{C} y_c^{d,r} \Psi\left(\sum_{t=1}^{C} \xi_{l,t}^r\right)\right).
\end{aligned} \tag{A.28}$$

Optimizing w.r.t. $\zeta_{i,j}$

Collecting only the terms in the log-likelihood that contain ζ yields:

$$\begin{aligned}
\mathcal{L}_{[\zeta]} = &\sum_{i=1}^{K}\sum_{j=1}^{V} (\tau_j - 1)\left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right)\right) \\
&+ \sum_{d=1}^{D}\sum_{n=1}^{N_d}\sum_{j=1}^{V} w_{n,j}^d \sum_{i=1}^{K} \phi_{n,i}^d \left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right)\right) \\
&- \sum_{i=1}^{K} \log\Gamma\left(\sum_{k=1}^{V} \zeta_{i,k}\right) + \sum_{i=1}^{K}\sum_{j=1}^{V} \log\Gamma(\zeta_{i,j}) \\
&- \sum_{i=1}^{K}\sum_{j=1}^{V} (\zeta_{i,j} - 1)\left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right)\right)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{K}\sum_{j=1}^{V}\left(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\right)\left(\tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d} - \zeta_{i,j}\right)\\
&- \sum_{i=1}^{K}\log\Gamma\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) + \sum_{i=1}^{K}\sum_{j=1}^{V}\log\Gamma(\zeta_{i,j})\\
&= \sum_{i=1}^{K}\sum_{j=1}^{V}\Psi(\zeta_{i,j})\left(\tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d} - \zeta_{i,j}\right)\\
&- \sum_{i=1}^{K}\sum_{j=1}^{V}\Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\left(\tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d} - \zeta_{i,j}\right)\\
&- \sum_{i=1}^{K}\log\Gamma\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) + \sum_{i=1}^{K}\sum_{j=1}^{V}\log\Gamma(\zeta_{i,j}). \qquad \text{(A.29)}
\end{aligned}
$$

Taking derivatives w.r.t. $\zeta_{i,j}$ gives:

$$
\begin{aligned}
\frac{\partial\mathcal{L}_{[\zeta]}}{\partial\zeta_{i,j}} &= \Psi'(\zeta_{i,j})\left(\tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d} - \zeta_{i,j}\right) - \Psi(\zeta_{i,j})\\
&- \Psi'\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\sum_{k=1}^{V}\left(\tau_k + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,k}^{d}\phi_{n,i}^{d} - \zeta_{i,k}\right) + \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\\
&- \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big) + \Psi(\zeta_{i,j})\\
&= \Psi'(\zeta_{i,j})\left(\tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d} - \zeta_{i,j}\right)\\
&- \Psi'\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\sum_{k=1}^{V}\left(\tau_k + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,k}^{d}\phi_{n,i}^{d} - \zeta_{i,k}\right). \qquad \text{(A.30)}
\end{aligned}
$$

Setting this derivative to zero in order to get a maximum (notice that the solutions for the different $\zeta_{i,j}$ are coupled, hence they have to be solved as a system of linear equations), we get the solution:

$$
\zeta_{i,j} = \tau_j + \sum_{d=1}^{D}\sum_{n=1}^{N_d} w_{n,j}^{d}\phi_{n,i}^{d}. \qquad \text{(A.31)}
$$

which can be easily verified by submitting the value for $\zeta_{i,j}$ above in the expression for the partial derivatives.

A.2.4 Optimizing w.r.t. $\xi_{c,l}^r$

Collecting only the terms in the log-likelihood that contain ξ gives:

$$\begin{aligned}
\mathcal{L}_{[\xi]} &= \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}(\omega_l - 1)\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right) \\
&+ \sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\lambda_c^d y_l^{d,r}\left((\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right) \\
&- \sum_{r=1}^{R}\sum_{c=1}^{C}\left(\log\Gamma\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) - \sum_{l=1}^{C}\log\Gamma(\xi_{c,l}^r)\right. \\
&+ \sum_{l=1}^{C}(\xi_{c,l}^r - 1)\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right) \\
&= \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\left(\Psi(\xi_{c,l}^r) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\right)\left(\omega_l + \sum_{d=1}^{D}\lambda_c^d y_l^{d,r} - \xi_{c,l}^r\right) \\
&- \sum_{r=1}^{R}\sum_{c=1}^{C}\log\Gamma\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) + \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\log\Gamma(\xi_{c,l}^r) \\
&= \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\Psi(\xi_{c,l}^r)\left(\omega_l + \sum_{d=1}^{D}\lambda_c^d y_l^{d,r} - \xi_{c,l}^r\right) \\
&- \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\left(\omega_l + \sum_{d=1}^{D}\lambda_c^d y_l^{d,r} - \xi_{c,l}^r\right) \\
&- \sum_{r=1}^{R}\sum_{c=1}^{C}\log\Gamma\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) + \sum_{r=1}^{R}\sum_{c=1}^{C}\sum_{l=1}^{C}\log\Gamma(\xi_{c,l}^r). \qquad \text{(A.32)}
\end{aligned}$$

Taking derivatives w.r.t. $\xi_{c,l}^r$ gives:

$$\begin{aligned}
\frac{\partial\mathcal{L}_{[\xi]}}{\partial\xi_{c,l}^r} &= \Psi'(\xi_{c,l}^r)\left(\omega_l + \sum_{d=1}^{D}\lambda_c^d y_l^{d,r} - \xi_{c,l}^r\right) - \Psi(\xi_{c,l}^r) \\
&- \Psi'\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big)\sum_{t=1}^{C}\left(\omega_t + \sum_{d=1}^{D}\lambda_c^d y_t^{d,r} - \xi_{c,t}^r\right) \\
&+ \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) - \Psi\Big(\sum_{t=1}^{C}\xi_{c,t}^r\Big) + \Psi(\xi_{c,l}^r)
\end{aligned}$$

$$= \Psi'(\xi^r_{c,l}) \left(\omega_l + \sum_{d=1}^{D} \lambda^d_c y^{d,r}_l - \xi^r_{c,l} \right)$$
$$- \Psi' \left(\sum_{t=1}^{C} \xi^r_{c,t} \right) \sum_{t=1}^{C} \left(\omega_t + \sum_{d=1}^{D} \lambda^d_c y^{d,r}_t - \xi^r_{c,t} \right). \tag{A.33}$$

Setting this derivative to zero in order to get a maximum (notice that the solutions for the different $\xi^r_{c,l}$ are coupled, hence they have to be solved as a system of linear equations), we get the solution:

$$\xi^r_{c,t} = \omega_t + \sum_{d=1}^{D} \lambda^d_c y^{d,r}_t. \tag{A.34}$$

which can be easily verified by submitting the value for $\xi^r_{c,l}$ above in the expression for the partial derivatives.

A.3 Parameter estimation

A.3.1 Estimating $\eta_{l,i}$

Collecting only the terms in the log-likelihood that contain η_l yields:

$$\mathcal{L}_{[\eta_{l,i}]} = \sum_{d=1}^{D} \frac{1}{N_d} \sum_{l=1}^{C} \lambda^d_l \sum_{n=1}^{N_d} \eta_l^T \phi^d_n - \log \sum_{l=1}^{C} \lambda^d_l \prod_{n=1}^{N_d} \left(\sum_{i=1}^{K} \phi^d_{n,i} \exp\left(\eta_{l,i} \frac{1}{N_d} \right) \right). \tag{A.35}$$

Taking derivatives w.r.t. $\eta_{l,i}$ gives:

$$\frac{\partial \mathcal{L}_{[\eta_{l,i}]}}{\partial \eta_{l,i}} = \sum_{d=1}^{D} \lambda^d_l \overline{\boldsymbol{\phi}_n}^d$$
$$- \sum_{d=1}^{D} \frac{\lambda^d_l \sum_{n=1}^{N_d} \left[\frac{1}{N_d} \phi^d_{n,i} \exp\left(\frac{1}{N_d} \eta_{c,i} \right) \right] \prod_{j=1, j \neq n}^{N_d} \left[\sum_{i=1}^{K} \phi^d_{j,i} \exp\left(\frac{1}{N_d} \eta_{c,i} \right) \right.}{\sum_{l=1}^{C} \lambda^d_l \prod_{n=1}^{N_d} \left(\sum_{i=1}^{K} \phi^d_{n.i} \exp\left(\eta_{l,i} \frac{1}{N_d} \right) \right)}$$

$$= \lambda_l^d \sum_{d=1}^{D} \overline{\boldsymbol{\phi}_n}^d \times \lambda_l^d \sum_{d=1}^{D} \left(- \frac{\sum_{n=1}^{N^d} \left[\frac{1}{N^d} \phi_{n,i}^d \exp\left(\frac{1}{N^d} \eta_{c,i} \right) \right]}{\sum_{l=1}^{C} \lambda_l^d \prod_{n=1}^{N^d} \left(\sum_{i=1}^{K} \phi_{n,i}^d \exp\left(\eta_{l,i} \frac{1}{N^d} \right) \right)} \right)$$

$$\times \lambda_l^d \sum_{d=1}^{D} \left(\frac{\prod_{j=1}^{N^d} \left[\sum_{i=1}^{K} \phi_{j,i}^d \exp\left(\frac{1}{N^d} \eta_{c,i} \right) \right]}{\sum_{i=1}^{K} \phi_{n,i}^d - \exp\left(\frac{1}{N^d} \eta_{c,i} \right)} \right). \tag{A.36}$$

Setting this derivative to zero does not lead to a closed-form solution.

Appendix B

Inference and parameter estimation in the regression model

In this appendix, a detailed version of Sections 4.2 and 4.3 is presented. Recall that in the Chapter 3, we already derived $\mathbb{E}_q[\log p(\beta_i|\tau)], \mathbb{E}_q[\log p(\theta^d|\alpha)$, $\mathbb{E}_q[\log p(z_n^d|\theta^d)]$ and $\mathbb{E}_q[\log p(w_n^d|z_n^d, \boldsymbol{\beta})]$. Since the regression model differs slightly from the classification one, here, only the derivations of the non-common terms of the evidence lower bound and the optimization of the new variables are obtained.

B.1 Derivation of the terms in the lower bound

$$\begin{aligned}
\mathbb{E}_q[\log p(y^{d,r}|x^d, b^r, v^r)] &= \mathbb{E}_q[\log Normal(y^{d,r}|x^d + b^r, v^r)] \\
&= \mathbb{E}_q\left[\log \frac{1}{\sqrt{2\pi}v^r} \exp\left(-\frac{\left(y^r - (x^d + b^r)\right)^2}{2(v^r)^2}\right)\right] \\
&= \mathbb{E}_q\left[-\frac{(y^r - x^d - b^r)^2}{2(v^r)^2}\right] - \mathbb{E}_q\left[\log\left(v^r\sqrt{2\pi}\right)\right] \\
&= -\frac{\left(y^r - m^d - b^r\right)^2}{2v^r} - \frac{1}{2}\log\left(2\pi v^r\right) \qquad \text{(B.1)}
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}_q[\log p(x^d|\bar{z}^d,\eta,\sigma)] &= \mathbb{E}_q\bigg[\log\bigg(\frac{1}{\sqrt{2\pi\sigma^2}}\exp\bigg(-\frac{(x^d-\eta^T\bar{z}^d)^2}{2\sigma^2}\bigg)\bigg)\bigg] \\
&= \mathbb{E}_q\bigg[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(x^d)^2 - 2x^d\eta^T\bar{z}^d + \eta^T\bar{z}^d(\bar{z}^d)^T\eta}{2\sigma^2}\bigg] \\
&= \frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\bigg(\mathbb{E}_q\big[(x^d)^2\big] \\
&\quad - 2\mathbb{E}_q[x^d]\eta^T\mathbb{E}_q[\bar{z}^d] + \eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\bigg) \qquad \text{(B.2)}
\end{aligned}$$

B.2 Optimizing the lower bound

B.2.1 Optimizing w.r.t. $\phi^d_{n,i}$

By collecting only the terms in the bound that contain $\phi^d_{n,i}$ and adding the necessary Lagrange multipliers, we have that:

$$\begin{aligned}
\mathcal{L}_{[\phi^d_{n,i}]} &= \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{i=1}^{K}\phi^d_{n,i}\bigg(\Psi(\gamma^d_i) - \Psi\Big(\sum_{j=1}^{K}\gamma^d_j\Big)\bigg) \\
&\quad + \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{j=1}^{V}\sum_{i=1}^{K} w^d_{n,j}\phi^d_{n,i}\bigg(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V}\zeta_{i,k}\Big)\bigg) \\
&\quad - \sum_{d=1}^{D}\frac{1}{2\sigma^2}\bigg(-2m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi^d_n\bigg) \\
&\quad - \sum_{d=1}^{D}\frac{1}{2\sigma^2}\bigg(\eta^T\frac{1}{(N^d)^2}\bigg(\sum_{n=1}^{N^d}\sum_{m\neq n}^{N^d}\phi^d_n(\phi^d_m)^T + \sum_{n=1}^{N^d} diag(\phi^d_n)\bigg)\eta\bigg) \\
&\quad - \sum_{d=1}^{D}\sum_{n=1}^{N^d}\sum_{i=1}^{K}\phi^d_{n,i}\log\phi^d_{n,i} + \mu\Big(\sum_{k=1}^{K}\phi^d_{n,k} - 1\Big), \qquad \text{(B.3)}
\end{aligned}$$

where we defined $\overline{\phi}^d = \frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d$. Taking derivatives w.r.t. $\phi_{n,i}^d$ gives:

$$\frac{\partial \mathcal{L}_{[\phi_{n,i}^d]}}{\partial \phi_{n,i}^d} = \Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg)$$
$$- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta_i\frac{1}{N^d} + \eta_i\frac{1}{(N^d)^2}\Bigg(\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta_i\Bigg) - \log\phi_{n,i}^d - 1 + \mu. \tag{B.4}$$

Setting this derivative to zero and solving for $\phi_{n,i}^d$ gives:

$$\Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg)$$
$$- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta_i\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\Bigg(\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta_i\Bigg) - \log\phi_{n,i}^d - 1 + \mu = 0$$
$$\Leftrightarrow \log\phi_{n,i}^d = \Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg)$$
$$- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta_i\frac{1}{N^d} + \eta_i\frac{1}{(N^d)^2}\Bigg(\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta_i\Bigg) - 1 + \mu$$
$$\Leftrightarrow \phi_{n,i}^d = \exp\Bigg(\Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^d \Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg)$$
$$- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta_i\frac{1}{N^d} + \eta_i\frac{1}{(N^d)^2}\Bigg(\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta_i\Bigg) - 1 + \mu\Bigg). \tag{B.5}$$

Plugging this expression in the constraint and solving for μ (or $\exp(\mu)$) yields:

$$
\begin{aligned}
&\sum_{k=1}^{K} \phi_{n,k}^{d} = 1 \\
&\Leftrightarrow \sum_{k=1}^{K} \exp\Bigg(\Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg) \\
&- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta^T\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\Bigg(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta\Bigg) - 1 + \mu\Bigg) = 1 \\
&\Leftrightarrow \sum_{k=1}^{K} \exp\Bigg(\Psi(\gamma_i^d) - \Psi\Bigg(\sum_{j=1}^{K}\gamma_j^d\Bigg) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\Bigg(\sum_{k=1}^{V}\zeta_{i,k}\Bigg) \\
&- \frac{1}{2\sigma^2}\Bigg(- 2m^d\eta\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\Bigg(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\Bigg)\eta\Bigg) - 1\Bigg)\exp(\mu) = 1 \\
&\Leftrightarrow \exp(\mu) = \\
&\frac{1}{\sum_{k=1}^{K}\exp\Bigg(\Psi(\gamma_i^d) - \Psi\big(\sum_{j=1}^{K}\gamma_j^d\big) + \sum_{j=1}^{V} w_{n,j}^{d}\Psi(\zeta_{i,j}) - \sum_{j=1}^{V} w_{n,j}^{d}\Psi\big(\sum_{k=1}^{V}\zeta_{i,k}\big)\Bigg)} \\
&\times \frac{1}{\sum_{k=1}^{K}\exp\Bigg(-\frac{1}{2\sigma^2}\Bigg(- 2m^d\eta\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\big(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\big)\eta\Bigg) - 1\Bigg)}.
\end{aligned}
\tag{B.6}
$$

Plugging this expression back in the expression for $\phi_{n,i}^d$ gives the solution:

$$\begin{aligned}
\phi_{n,i}^d &= \frac{\exp\Big(\Psi(\gamma_i^d) - \Psi\big(\sum_{j=1}^K \gamma_j^d\big) + \sum_{j=1}^V w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^V w_{n,j}^d \Psi\big(\sum_{k=1}^V \zeta_{i,k}\big)\Big)}{\sum_{k=1}^K \exp\Big(\Psi(\gamma_i^d) - \Psi\big(\sum_{j=1}^K \gamma_j^d\big) + \sum_{j=1}^V w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^V w_{n,j}^d \Psi\big(\sum_{k=1}^V \zeta_{i,k}\big)\Big)} \\
&\times \frac{\exp\Big(-\frac{1}{2\sigma^2}\Big(- 2m^d\eta\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\big(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\big)\eta\Big) - 1\Big)}{\sum_{k=1}^K \exp\Big(\sum_{k=1}^K \exp\Big(-\frac{1}{2\sigma^2}\Big(- 2m^d\eta\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\big(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\big)\eta\Big) - 1\Big)\Big)} \\
&\propto \exp\left(\Psi(\gamma_i) + \sum_{j=1}^V w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^V w_{n,j}^d \Psi\left(\sum_{k=1}^V \zeta_{i,k}\right)\right) \\
&\times \exp\left(-\frac{1}{2\sigma^2}\Big(- 2m^d\eta\frac{1}{N^d} + \eta^T\frac{1}{(N^d)^2}\big(N^d\sum_{m\neq n}^{N^d}\phi_m^d + 1\big)\eta\Big)\right) \\
&\propto \exp\left(\Psi(\gamma_i) + \sum_{j=1}^V w_{n,j}^d \Psi(\zeta_{i,j}) - \sum_{j=1}^V w_{n,j}^d \Psi\left(\sum_{k=1}^V \zeta_{i,k}\right) + \frac{m^d}{N^d\sigma^2}\eta\right) \\
&\times \exp\left(-\frac{\eta^T N^d \sum_{m\neq n}^{N^d}\phi_m^d\eta + \eta^T\eta}{2(N^d)^2\sigma^2}\right).
\end{aligned} \tag{B.7}$$

B.2.2 Optimizing w.r.t. m^d

We again start by collecting only the terms in the bound that contain m^d.

$$\begin{aligned}
\mathcal{L}_{[m^d]} = &-\sum_{d=1}^D\sum_{r=1}^R -\frac{\big(y^r - m^d - b^r\big)}{2v^r} \\
&+\sum_{d=1}^D -\frac{1}{2\sigma^2}\left((m^d)^2 - 2m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d\right).
\end{aligned} \tag{B.8}$$

Taking derivatives w.r.t. m^d gives:

$$\frac{\partial\mathcal{L}_{[m^d]}}{\partial m^d} = \sum_{r=1}^R\left(-\frac{1}{2v^r}\Big(- 2y^{d,r} + 2(m^d) + 2b^r\Big)\right) - \frac{1}{2\sigma^2}\Big(- 2m^d - 2\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d\Big). \tag{B.9}$$

Setting this derivative to zero and solving for m^d yields:

$$\begin{aligned}
\frac{\partial \mathcal{L}_{[m^d]}}{\partial m^d} &= \sum_{r=1}^{R}\left(-\frac{1}{2v^r}\left(-2y^{d,r}+2(m^d)+2b^r\right)\right)- \\
&\frac{1}{2\sigma^2}\left(-2m^d-2\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d\right)=0 \\
&\Leftrightarrow \sum_{r=1}^{R}-\frac{2m^d}{2v^r}-\frac{-2m^d}{2\sigma^2}=-\sum_{r=1}^{R}\left(-\frac{\left(-2y^{d,r}+2b^r\right)}{2v^r}\right) \\
&+\frac{\left(-2\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d\right)}{2\sigma^2} \\
&\Leftrightarrow \sum_{r=1}^{R}\frac{-\sigma^2m^d+v^rm^d}{v^r\sigma^2}=\sum_{r=1}^{R}\frac{-y^{d,r}+b^r}{v^r}-\frac{\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d}{\sigma^2} \\
&\Leftrightarrow m^d=\sum_{r=1}^{R}\frac{y^{d,r}\sigma^2-b^r\sigma^2+v^r\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d}{\sigma^2+v^r}. \qquad \text{(B.10)}
\end{aligned}$$

B.2.3 Optimizing w.r.t. ν^d

Collecting only the terms in the bound that contain ν gives:

$$\mathcal{L}_{[\nu]}=\sum_{d=1}^{D}\sum_{r=1}^{R}-\frac{1}{2v^r}(\nu^d)-\frac{1}{2\sigma^2}\nu^d+\frac{1}{2}\log(2\pi\nu^d). \qquad \text{(B.11)}$$

Taking derivatives w.r.t. ν^d gives:

$$\frac{\partial \mathcal{L}_{[\nu^d]}}{\partial \nu^d}=\sum_{r=1}^{R}-\frac{1}{2v^r}-\frac{1}{2\sigma^2}+\frac{1}{2\nu^d}. \qquad \text{(B.12)}$$

Setting this derivative to zero and solving for ν^d gives:

$$\begin{aligned}
\sum_{r=1}^{R}-\frac{1}{2v^r}-\frac{1}{2\sigma^2}+\frac{1}{2\nu^d}&=0 \\
\Leftrightarrow \nu^d&=\sigma^2+\sum_{r=1}^{R}v^r. \qquad \text{(B.13)}
\end{aligned}$$

B.3 Parameter estimation

B.3.1 Estimating η

By collecting only the terms in the log-likelihood that contain η_l, the objective function becomes:

$$\begin{aligned}\mathcal{L}_{[\eta]} &= \sum_{d=1}^{D} -\frac{1}{2\sigma^2}\Bigg(-2m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d + \eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\Bigg) \\ &= \sum_{d=1}^{D}\left(\frac{1}{\sigma^2}m^d\eta^T\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d - \frac{1}{2\sigma^2}\eta^T\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right). \end{aligned} \tag{B.14}$$

Taking derivatives w.r.t. η gives:

$$\sum_{d=1}^{D}\left(\frac{1}{\sigma^2}m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d - \frac{1}{\sigma^2}\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right). \tag{B.15}$$

Setting this derivative to zero and solving for η yields:

$$\begin{aligned}&\sum_{d=1}^{D}\left(\frac{1}{\sigma^2}m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d - \frac{1}{\sigma^2}\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right) = 0 \\ &\qquad\Leftrightarrow \sum_{d=1}^{D}\left(m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d = \mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]\eta\right) \\ &\qquad\qquad\Leftrightarrow \eta^T = \sum_{d=1}^{D}\mathbb{E}_q\big[\bar{z}^d(\bar{z}^d)^T\big]^{-1}m^d\frac{1}{N^d}\sum_{n=1}^{N^d}\phi_n^d. \end{aligned} \tag{B.16}$$

Estimating b

Collecting only the terms in the log-likelihood that contain b^r gives:

$$\mathcal{L}_{[b^r]} = \sum_{d=1}^{D}\sum_{r=1}^{R} -\frac{1}{2v^r}\Bigg(-2y^{d,r}b^r + 2m^d b^r + (b^r)^2\Bigg). \tag{B.17}$$

By taking derivatives w.r.t. b^r we have that:

$$\frac{\partial \mathcal{L}_{[b^r]}}{\partial b^r} = \sum_{d=1}^{D} \frac{y^{d,r} - m^d - b^r}{v^r}. \tag{B.18}$$

Setting this derivative to zero and solving for b^r gives:

$$\begin{aligned} \sum_{d=1}^{D} \sum_{d=1}^{D} \frac{y^{d,r} - m^d - b^r}{v^r} &= 0 \\ \Leftrightarrow b^r &= \frac{\sum_{d=1}^{D} y^{d,r} - m^d}{D}. \end{aligned} \tag{B.19}$$

B.3.2 Estimating v

By gathering only the terms in the log-likelihood that contain $v^r.$, the objective function is given by:

$$\begin{aligned} \mathcal{L}_{[v^r]} = \sum_{d=1}^{D} \sum_{r=1}^{R} -\frac{1}{2v^r} \Big((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2 \Big) \\ - \frac{\log\left(2\pi v^r\right)}{2}. \end{aligned} \tag{B.20}$$

Taking derivatives w.r.t. v^r gives:

$$\begin{aligned} \frac{\partial \mathcal{L}_{[v^r]}}{\partial v^r} = \sum_{d=1}^{D} \frac{1}{2(v^r)^2} \Big((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2 \Big) \\ - \frac{1}{2v^r}. \end{aligned} \tag{B.21}$$

Setting this derivative to zero and solving for v^r yields:

$$\begin{aligned}
&\sum_{d=1}^{D} \frac{1}{2(v^r)^2}\left((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2\right) - \frac{1}{2v^r} \\
&= 0 \\
&\Leftrightarrow \sum_{d=1}^{D} \frac{2(v^r)^2}{2v^r} = \sum_{d=1}^{D}\left((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2\right) \\
&\Leftrightarrow v^r = \frac{1}{D}\sum_{d=1}^{D}\left((y^{d,r})^2 - 2y^{d,r}m^d - 2y^{d,r}b^r + \nu^d + (m^d)^2 + 2m^d b^r + (b^r)^2\right).
\end{aligned} \tag{B.22}$$

Appendix C

Classification model using maximum likelihood estimation

In this appendix a *mle* version of the classification model is presented. The generative process of this simpler variant is similar to the model described in Chapter 3:

1. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$
   (b) For the n^{th} word
      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Multinomial(\beta_{z_n^d})$
   (c) Draw latent (true) class $c^d|\mathbf{z}^d, \boldsymbol{\eta} \sim Softmax(\bar{z}^d, \boldsymbol{\eta})$ where $\bar{z}^d = \frac{1}{N^d}\sum_{n=1}^{N^d} z_n^d$ and

$$p(c^d|\bar{z}^d, \boldsymbol{\eta}) = \frac{\exp(\eta_c^T \bar{z}^d)}{\sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d)} \tag{C.1}$$

   (d) For each annotator r
      i. Draw annotator's answer $y^{d,r}|c^d, \boldsymbol{\pi}^r \sim Multinomial(\pi_{c^d}^r)$.

The differences between this model and the one presented in Chapter 3 are in the $\boldsymbol{\beta}$ and $\boldsymbol{\pi}$ parameters. While in the fully Bayesian MA-sLDA there are two priors connected to these parameters (compare Figures C.1 and 3.1) and, consequently, two new steps in the generative process, the

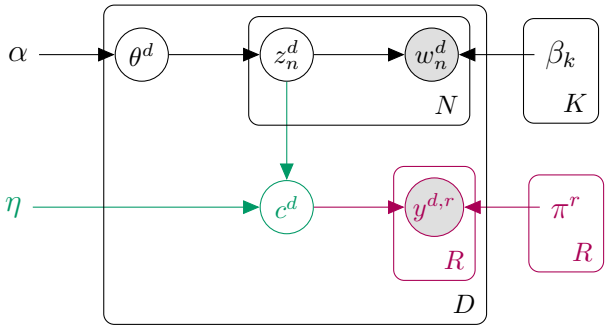


Figure C.1: MA-sLDA graphical model.

model described here is simpler. This implies a different posterior:

$$p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \mathbf{c}|\mathbf{w}_{1:D}, \mathbf{y}_{1:D}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta)}{p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta)} \tag{C.2}$$

$$= \frac{\prod_{d=1}^{D} p(\theta^d|\alpha)\left(\prod_{n=1}^{N^d} p(z_n^d|\theta^d)p(w_n^d|z_n^d, \boldsymbol{\beta})\right) p(c^d|\boldsymbol{z}^d, \boldsymbol{\eta}) \prod_{r=1}^{R} p(y^{d,r}|c^d, \boldsymbol{\pi}^r)}{\int_\theta^d \prod_d^D p(\theta^d|\alpha) \sum_{\mathbf{z}} \left(\prod_{n=1}^{N^d} p(z_n^d|\theta)p(w_n^d|z_n^d, \boldsymbol{\beta})\right) \sum_c p(c^d|\mathbf{z}^d, \boldsymbol{\eta}) \prod_{r=1}^{R} p(y^{d,r}|c^d, \boldsymbol{\pi}^r)} \tag{C.3}$$

and an analogously simpler variational distribution of the latent variables:

$$q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}) = \prod_{d=1}^{D} q(\theta^d|\gamma^d)\left(\prod_{n=1}^{N^d} q(z_n^d|\phi_n^d)\right) q(c^d|\lambda^d), \tag{C.4}$$

where $\boldsymbol{\gamma}, \boldsymbol{\phi}_{1:D}$ and $\boldsymbol{\lambda}$ are only three variational parameters. Therefore, the differences between the evidence lower bound of these two versions are that, in the one from Chapter 3 there are two terms that do not exist in this one ($\sum_{i=1}^{K} \mathbb{E}_q[\log p(\beta_i|\tau)$ and $\sum_{r=1}^{R}\sum_{c=1}^{C} \mathbb{E}_q[\log p(\pi_c^r|\omega)]$) and that, while in Bayesian variant the expectations of the log of the Dirichlet are given by:

$$\mathbb{E}_q[\log \theta_i^d] = \Psi(\gamma_i^d) - \Psi\left(\sum_{j=1}^{K} \gamma_j^d\right) \tag{C.5}$$

$$\mathbb{E}_q[\log \beta_{i,j}] = \Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V} \zeta_{i,k}\right) \tag{C.6}$$

$$\mathbb{E}_q[\log \pi^r_{c,l}] = \Psi(\xi^r_{c,l}) - \Psi\left(\sum_{t=1}^{C} \xi^r_{c,t}\right), \tag{C.7}$$

here, are only:

$$\mathbb{E}_q[\log \theta^d_i] = \Psi(\gamma^d_i) - \Psi\left(\sum_{j=1}^{K} \gamma^d_j\right) \tag{C.8}$$

$$\mathbb{E}_q[\log \beta_{i,j}] = \log \beta_{i,j} \tag{C.9}$$

$$\mathbb{E}_q[\log \pi^r_{c,l}] = \log \pi^r_{c,l}. \tag{C.10}$$

This gets intuitive if we think that, here β and π are just one value and the expectation of a single value is the value itself. Oppositely, in the entirely Bayesian approach, they are Dirichlet distributions and, thus, are calculated in a more complex way.

Also, here, the updates that involve β and π are, naturally, different from the previously presented version of MA-sLDA. Therefore, we follow by showing them:

$$\phi^d_{n,i} \propto \sum_{j=1}^{V} w^d_{n,j} \beta_{k,j} \exp\left(\Psi(\gamma_i) + \frac{\sum_{l=1}^{C} \lambda^d_l \eta_{l,i}}{N^d} - (h^T(\phi^d_n)^{old})^{-1} h_i\right); \tag{C.11}$$

$$\lambda^d_l \propto \left(\prod_{r=1}^{R}\prod_{c=1}^{C} y_c^{d,r} \pi^r_{k,c}\right) \exp\left(\eta_k^T \overline{\phi^d}\right); \tag{C.12}$$

$$\beta_{i,j} \propto \sum_{d=1}^{D}\sum_{n=1}^{N^d} w^d_{n,j} \phi^d_{n,i}; \tag{C.13}$$

$$\pi^r_{c,l} \propto \sum_{d=1}^{D} y^{r,d}_l \lambda_c. \tag{C.14}$$

Notice that $\phi^d_{n,i}$ and λ^d_l are latent variables and $\beta_{i,j}$ and $\pi^r_{c,l}$ are parameters. So, in conclusion, this model is distinguished from the model of Chapter 3 by performing variational Bayesian inference on the first two and maximum likelihood estimation on the last ones.

Appendix D

Regression model using maximum likelihood estimation

In this appendix a *mle* version of the regression model is presented. The generative process of this simpler variant is similar to the model described in Chapter 4:

1. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dirichlet(\alpha)$
   (b) For the n^{th} word
      i. Draw topic assignment $z_n^d|\theta^d \sim Multinomial(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Multinomial(\beta_{z_n^d})$
   (c) Draw latent (true) value $x^d|\mathbf{z}^d, \boldsymbol{\eta}, \sigma \sim Normal(x^d|\boldsymbol{\eta}^T \bar{z}^d, \sigma)$
   (d) For each annotator r
      i. Draw annotator's answer $y^{d,r}|x^d, b^r, v^r \sim Normal(y^{d,r}|x^d + b^r, v^r)$

This model differs from the one from Chapter 4 in the way the per topic word distribution parameter $\boldsymbol{\beta}$ is calculated. While in the fully Bayesian MA-sLDA for regression there is a Dirichlet on this variable (τ), here it acts as a global parameter obtained by maximum likelihood estimation. Hence, the

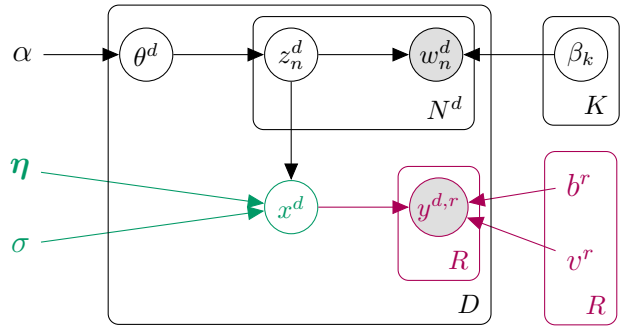


Figure D.1: Graphical model.

posterior becomes:

$$
\begin{aligned}
& p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \mathbf{x} | \mathbf{w}_{1:D}, \mathbf{y}_{1:D}) \\
&= \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D} | \Theta)}{p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D} | \Theta)} \qquad \text{(D.1)} \\
&= \frac{\prod_{d=1}^{D} p(\theta^d|\alpha) \left(\prod_{n=1}^{N^d} p(z_n^d|\theta^d) p(w_n^d|z_n^d, \boldsymbol{\beta}) \right)}{\int_\theta^d \prod_d^D p(\theta^d|\alpha) \sum_{\mathbf{z}} \left(\prod_{n=1}^{N^d} p(z_n^d|\theta) p(w_n^d|z_n^d, \boldsymbol{\beta}) \right)} \\
&\times \frac{\prod_{d=1}^{D} p(x^d|\boldsymbol{z}^d, \boldsymbol{\eta}, \sigma) \prod_{r=1}^{R} p(y^{d,r}|x^d, b^r, v^r)}{\int_\theta^d \prod_d^D \int_x p(x^d|\mathbf{z}^d, \boldsymbol{\eta}, \sigma) \prod_{r=1}^{R} p(y^{d,r}|x^d, b^r, v^r)}, \qquad \text{(D.2)}
\end{aligned}
$$

Similarly, the variational distribution of the latent variables of this *mle* model is:

$$
q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{x}) = \prod_{d=1}^{D} q(\theta^d|\gamma^d) \left(\prod_{n=1}^{N^d} q(z_n^d|\phi_n^d) \right) q(x^d|m^d, \nu^d), \qquad \text{(D.3)}
$$

where $\boldsymbol{\gamma}, \boldsymbol{\phi}_{1:D}$, $\boldsymbol{m}$ and $\boldsymbol{\nu}$ are the variational parameters. Also, the fact that the parameter $\boldsymbol{\beta}$ does not have a prior in this version means that:

$$
\mathbb{E}_q[\log \beta_{i,j}] = \log \beta_{i,j}, \qquad \text{(D.4)}
$$

which is intuitive for the reasons described in the Appendix C.

Therefore, to make the lower bound as close as possible to the true posterior, we update the variational parameter $phi_{n,i}^d$ and the global model

parameter $\beta_{i,j}$ as:

$$\phi_{n,i}^{d} \propto \exp\left(\Psi(\gamma_i) + \sum_{j=1}^{V} w_{n,j}^{d}\left(\Psi(\zeta_{i,j}) - \Psi\left(\sum_{k=1}^{V}\zeta_{i,k}\right)\right)\right.$$
$$\left. + \frac{m^d}{N^d\sigma^2}\boldsymbol{\eta} - \frac{2(\boldsymbol{\eta}^T\phi_{-n}^{d})\boldsymbol{\eta} + (\boldsymbol{\eta}\circ\boldsymbol{\eta})}{2(N^d)^2\sigma^2}\right), \tag{D.5}$$

$$\beta_{i,j} \propto \sum_{d=1}^{D}\sum_{n=1}^{N^d} w_{n,j}^{d}\phi_{n,i}^{d}. \tag{D.6}$$

Appendix E

Submitted publications

This appendix presents the accepted publication on AAAI HCOMP2015 conference.

Learning Supervised Topic Models from Crowds

Abstract

The growing need to analyze large collections of documents has led to great developments in topic modeling. Since documents are frequently associated with other related variables, such as labels or ratings, much interest has been placed on supervised topic models. However, the nature of most annotation tasks, prone to ambiguity and noise, often with high volumes of documents, deem learning under a single-annotator assumption unrealistic or unpractical for most real-world applications. In this paper, we propose a supervised topic model that accounts for the heterogeneity and biases among different annotators that are encountered in practice when learning from crowds. We develop an efficient stochastic variational inference algorithm that is able to scale to very large datasets, and we empirically demonstrate the advantages of the proposed model over state of the art approaches.

Introduction

Topic models, such as latent Dirichlet allocation (LDA), allow us to analyze large collections of documents, by revealing their underlying themes, or topics, and how each document exhibits them (Blei, Ng, and Jordan 2003). Therefore, it is not surprising that topic models have become a standard tool in machine learning, with many applications that transcend their original purpose of modeling textual data, such as analyzing images (Fei-Fei and Perona 2005; Wang, Blei, and Fei-Fei 2009), videos (Niebles, Wang, and Fei-Fei 2008), survey data (Erosheva, Fienberg, and Joutard 2007) or social networks data (Airoldi et al. 2007).

Since documents are frequently associated with other variables such as labels, tags or ratings, much interest has been placed on supervised topic models (Mcauliffe and Blei 2008), which allow the use of that extra information to "guide" the topics discovery. By jointly learning the topics distributions and a regression or classification model, supervised topic models have been shown to outperform the separate use of their unsupervised analogues with an external regression/classification algorithm (Wang, Blei, and Fei-Fei 2009; Zhu, Ahmed, and Xing 2012).

Supervised topics models are then state-of-the-art approaches for predicting target variables associated with complex high-dimensional data, such as documents or images. Unfortunately, the size of modern datasets make the use of a single annotator unrealistic and unpractical for the majority of the real-world applications that involve some form of human labeling. For instance, the popular Reuters-21578 benchmark dataset was categorized by a group of personnel from Reuters Ltd and Carnegie Group, Inc. Similarly, the LabelMe[1] project asks volunteers to annotate images from a large collection using an online tool. Hence, it is seldom the case where a single oracle labels an entire collection.

Furthermore, the Web, through its social nature, also exploits the wisdom of crowds to annotate large collections of documents and images. By categorizing texts, tagging images or rating products, Web users are generating large volumes of labeled content. However, when learning supervised models from crowds, the quality of labels can vary a lot due to task subjectivity and differences in annotator reliability (or bias) (Snow et al. 2008; Rodrigues, Pereira, and Ribeiro 2013). It is therefore essential to account for these issues when learning from this increasingly common type of data. Hence, the interest of researchers on building models that take the reliabilities of different annotators into consideration and mitigate the effect of their biases has spiked during the last few years (e.g. (Welinder et al. 2010; Yan et al. 2014)).

The increasing popularity of crowdsourcing platforms like Amazon Mechanical Turk (AMT) has further contributed to the recent developments in learning from crowds. This kind of platforms offer a fast, scalable and inexpensive solution for labeling large amounts of data. However, their heterogeneous nature in terms of contributors makes their straightforward application prone to many sorts of labeling noise and bias. Hence, a careless use of crowdsourced data as training data risks generating flawed models.

In this paper we propose a fully generative supervised topic model that is able to account for the different reliabilities of multiple annotators and correct their biases. The proposed model is then capable of jointly modeling the words in documents as arising from a mixture of topics, the latent true labels as a result of the empirical distribution over topics

[1] http://labelme.csail.mit.edu

of the documents, and the labels of the multiple annotators as noisy versions of that latent ground truth. Although we focus on multi-class classification problems, the same rationale can be applied to regression problems. Since the majority of the tasks for which multiple annotators are used generally involve complex data such as text, images and video, by developing a multi-annotator supervised topic model we are contributing with a powerful tool for learning predictive models of complex high-dimensional data from crowds.

Given that the increasing sizes of modern datasets can pose a problem for obtaining human labels as well as for Bayesian inference, we propose an efficient stochastic variational inference algorithm (Hoffman et al. 2013) that is able to scale to very large datasets. We empirically show, using both simulated and real multiple-annotator labels obtained from AMT for popular text and image collections, that the proposed model is able to outperform other state-of-the-art approaches. We further show the computational and predictive advantages of the stochastic variational inference algorithm over its batch counterpart.

State of the art

Latent Dirichlet allocation (LDA) soon proved to be a powerful tool for modeling documents (Blei, Ng, and Jordan 2003) and images (Fei-Fei and Perona 2005), by extracting their underlying topics. However, the need to model the relationship between documents and labels quickly gave rise to many supervised variants of LDA. One of the first notable works was that of (Mcauliffe and Blei 2008) in developing supervised LDA (sLDA). By extending LDA through the inclusion of a response variable that is linearly dependent on the mean topic-assignments of the words in a document, sLDA is able to jointly model the documents and their responses, in order to find latent topics that will best predict the response variables for future unlabeled documents. Although initially developed for general continuous response variables, (Wang, Blei, and Fei-Fei 2009) later extended sLDA to classification problems, by modeling the relationship between topic-assignments and labels with a softmax function.

There are several ways in which document classes can be included in LDA. The most natural one in this setting is probably the sLDA approach, since the classes are directly dependent on the empirical topic mixture distributions. This approach is coherent with the generative perspective of LDA but, nevertheless, several discriminative alternatives also exist. For example, DiscLDA (Lacoste-Julien, Sha, and Jordan 2009) introduces a class-dependent linear transformation on the topic mixture proportions, whose parameters are estimated by maximizing the conditional likelihood of response variables. (Ramage et al. 2009) propose Labeled-LDA, a variant of LDA that incorporates supervision by constraining the topic model to use only the topics that correspond to a document's label set. While this has the advantage of allowing multiple labels per document, it is restrictive in the sense that the number of topics needs to be the same as the number of possible labels.

The approaches discussed so far rely on likelihood-based estimation procedures. The work of (Zhu, Ahmed, and Xing 2012) contrasts with these approaches by proposing MedLDA, a supervised topic model that utilizes the max-margin principle for estimation. Despite its margin-based advantages, MedLDA looses the probabilistic interpretation of the document classes given the topic mixture distributions. On the contrary, this paper proposes a fully generative probabilistic model of the labels of multiple annotators and the words in the documents.

Learning from multiple annotators is an increasingly important research topic. Since the early work of (Dawid and Skene 1979), who attempted to obtain point estimates of the error rates of patients given repeated but conflicting responses to various medical questions, many approaches have been proposed. These usually rely on latent variable models. For example, (Smyth et al. 1995) proposed a model to estimate the ground truth from the labels of multiple experts, which is then used to train a classifier.

While earlier works usually focused on estimating the ground truth and the error rates of different annotators, recent works are more focused on the problem of learning a classifier. This idea was explored in (Raykar et al. 2010), who proposed an approach for jointly learning the levels of expertise of different annotators and the parameters of a logistic regression classifier, by modeling the ground truth labels as latent variables. This work was later extended by (Yan et al. 2014) by considering the dependencies of the annotators' labels on the instances they are labeling, and also by (Rodrigues, Pereira, and Ribeiro 2014) through the use of Gaussian process classifiers. The model proposed in this paper shares the same intuition with this line of work, and models the true labels as latent variables. However, it differs significantly by using a fully Bayesian approach for estimating the reliabilities and biases of the different annotators. Furthermore, it considers the problems of learning a low-dimensional representation of the input data (through topic modeling) and modeling the answers of multiple annotators jointly, providing an efficient stochastic variational inference algorithm.

Approach

In this section we develop a multi-class supervised topic model with multiple annotators. We start by deriving a (*batch*) variational inference algorithm for approximating the posterior distribution over the latent variables and an algorithm to estimate the model parameters. We then develop a stochastic variational inference algorithm that gives the model the capability of handling large collections of documents. Finally, we show how to use the learned model to classify new documents.

Proposed model

Let $\mathcal{D} = \{\mathbf{w}^d, \mathbf{y}^d\}_{d=1}^D$ be an annotated corpus of size D, where each document $\mathbf{w}^d$ is given a set of labels $\mathbf{y}^d = \{y_r^d\}_{r=1}^{R_d}$ from R_d distinct annotators. We can take advantage of the inherent topical structure of documents and model their words as arising from a mixture of topics, each being defined as a distribution over the words in a vocabulary, as in LDA. In LDA, the n^{th} word, w_n^d, in a document d

is provided a discrete topic-assignment z_n^d, which is drawn from the documents' distribution over topics θ^d. This allows us to build lower-dimensional representations of documents, which we can explore to build classification models by assigning coefficients $\boldsymbol{\eta}$ to the mean topic-assignment of the words in the document, $\bar{z}^d$, and applying a softmax function in order to obtain a distribution over classes.

Unfortunately, a direct mapping between document classes and the labels provided by the different annotators in a multiple-annotator setting would correspond to assuming that they are all equally reliable, an assumption that is violated in practice, as previous works clearly demonstrate (e.g. (Snow et al. 2008; Rodrigues, Pereira, and Ribeiro 2013)). Hence, we assume the existence of a latent ground truth class, and model the labels from the different annotators using a noise model that states that, given a true class c, each annotator r provides the label l with some probability $\pi_{c,l}^r$. Hence, by modeling $\boldsymbol{\pi}^r$ we are in fact modeling a per-annotator confusion matrix, which allows us to account for their different levels of expertise and correct their potential biases.

The generative process of the proposed model can then be summarized as follows:

1. For each annotator r
   (a) For each class c
      i. Draw reliability parameter $\pi_c^r|\omega \sim Dir(\omega)$
2. For each topic k
   (a) Draw topic distribution $\beta_k|\tau \sim Dir(\tau)$
3. For each document d
   (a) Draw topic proportions $\theta^d|\alpha \sim Dir(\alpha)$
   (b) For the n^{th} word
      i. Draw topic assignment $z_n^d|\theta^d \sim Mult(\theta^d)$
      ii. Draw word $w_n^d|z_n^d, \boldsymbol{\beta} \sim Mult(\beta_{z_n^d})$
   (c) Draw latent (true) class $c^d|\mathbf{z}^d, \boldsymbol{\eta} \sim Softmax(\bar{z}^d, \boldsymbol{\eta})$
   (d) For each annotator $r \in R_d$
      i. Draw annotator's label $y^{d,r}|c^d, \boldsymbol{\pi}^r \sim Mult(\pi_{c^d}^r)$

where R_d denotes the set of annotators that labeled the d^{th} document, $\bar{z}^d = \frac{1}{N_d}\sum_{n=1}^{N_d} z_n^d$, and the softmax is given by:

$$p(c^d|\bar{z}^d, \boldsymbol{\eta}) = \frac{\exp(\eta_c^T \bar{z}^d)}{\sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d)}.$$

Figure 1 shows a graphical model representation of the proposed model, where K denotes the number of topics, C is the number of classes, R is the total number of annotators and N_d is the number of words in the document d. Notice that we included a Dirichlet prior over the topics β_k to produce a smooth posterior and control sparsity. Similarly, instead of computing maximum likelihood or MAP estimates for the annotators reliability parameters π_c^r, we place a Dirichlet prior over these variables and perform (approximate) Bayesian inference. This contrasts with previous works on learning from crowds (Raykar et al. 2010; Yan et al. 2010).

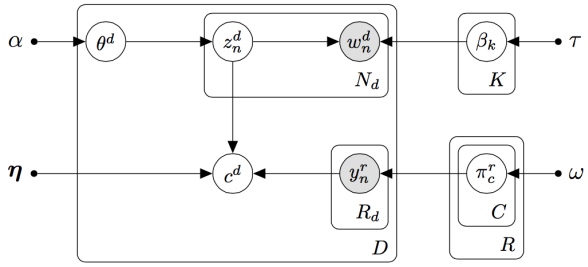


Figure 1: Graphical model representation of the proposed model.

Variational param.	Original param.
ξ_c^r	π_c^r
ζ_k	β_k
γ^d	θ^d
λ^d	c^d
ϕ_n^d	z_n^d

Table 1: Correspondence between variational parameters and the original parameters.

Approximate inference

Given a dataset $\mathcal{D}$, the goal of inference is to compute the posterior distribution of the per-document topic proportions θ^d, the per-word topic assignments z_n^d, the per-topic distribution over words β_k, the per-document latent true class c^d, and the per-annotator confusion parameters $\boldsymbol{\pi}^r$. As with LDA, computing the exact posterior distribution of the latent variables is computationally intractable. Hence, we employ mean-field variational inference to perform approximate Bayesian inference.

Variational inference methods seek to minimize the KL divergence between the variational and the true posterior distribution. We assume a fully-factorized (mean-field) variational distribution of the form:

$$q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}) = \left(\prod_{r=1}^{R}\prod_{c=1}^{C} q(\pi_c^r|\xi_c^r)\right)$$
$$\times \left(\prod_{i=1}^{K} q(\beta_i|\zeta_i)\right)\prod_{d=1}^{D} q(\theta^d|\gamma^d)q(c^d|\lambda^d)\prod_{n=1}^{N_d} q(z_n^d|\phi_n^d),$$

where $\boldsymbol{\xi}_{1:R}$, $\boldsymbol{\zeta}$, $\boldsymbol{\gamma}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\phi}_{1:D}$ are variational parameters. Table 1 shows the correspondence between variational parameters and the original parameters.

Let $\Theta = \{\alpha, \tau, \omega, \boldsymbol{\eta}\}$ denote the model parameters. Following (Jordan et al. 1999), the KL minimization can be equivalently formulated as maximizing the following lower

bound on the log marginal likelihood,

$$\begin{aligned}
&\log p(\mathbf{w}_{1:D}, \mathbf{y}_{1:D}|\Theta) \\
&\quad = \log \int \sum_{\mathbf{z},\boldsymbol{c}} q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}) \\
&\quad \times \frac{p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta)}{q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})} \, d\boldsymbol{\theta} \, d\boldsymbol{\beta} \, d\boldsymbol{\pi}_{1:R} \\
&\quad \geqslant \mathbb{E}_q[\log p(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \mathbf{w}_{1:D}, \mathbf{y}_{1:D}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R}|\Theta)] \\
&\quad + \mathbb{E}_q[\log q(\boldsymbol{\theta}, \mathbf{z}_{1:D}, \boldsymbol{c}, \boldsymbol{\beta}, \boldsymbol{\pi}_{1:R})] \\
&\quad = \mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\phi}_{1:D}, \boldsymbol{\lambda}, \boldsymbol{\zeta}, \boldsymbol{\xi}_{1:R}|\Theta), \qquad (1)
\end{aligned}$$

which we maximize using coordinate ascent.

Optimizing $\mathcal{L}$ w.r.t. γ and ζ gives the same coordinate ascent updates as in (Blei, Ng, and Jordan 2003):

$$\gamma_i^d = \alpha + \sum_{n=1}^{N_d} \phi_{n,i}^d \qquad (2)$$

$$\zeta_{i,j} = \tau + \sum_{d=1}^{D} \sum_{n=1}^{N_d} w_{n,j}^d \phi_{n,i}^d. \qquad (3)$$

The variational Dirichlet parameters ξ can be optimized by collecting only the terms in $\mathcal{L}$ that contain ξ:

$$\begin{aligned}
\mathcal{L}_{[\xi]} = &\sum_{r=1}^{R} \sum_{c=1}^{C} \sum_{l=1}^{C} \mathbb{E}_q[\log \pi_{c,l}^r] \Big(\omega + \sum_{d=1}^{D_r} \lambda_c^d y_l^{d,r} - \xi_{c,l}^r \Big) \\
&- \sum_{r=1}^{R} \sum_{c=1}^{C} \log \Gamma \Big(\sum_{t=1}^{C} \xi_{c,t}^r \Big) + \sum_{r=1}^{R} \sum_{c=1}^{C} \sum_{l=1}^{C} \log \Gamma(\xi_{c,l}^r),
\end{aligned}$$

where D_r denotes the documents labeled by the r^{th} annotator, $\mathbb{E}_q[\log \pi_{c,l}^r] = \Psi(\xi_{c,l}^r) - \Psi(\sum_{t=1}^{C} \xi_{c,t}^r)$, and $\Gamma(\cdot)$ and $\Psi(\cdot)$ are the gamma and digamma functions, respectively. Taking derivatives of $\mathcal{L}_{[\xi]}$ w.r.t. ξ and setting them to zero, yields the following update:

$$\xi_{c,l}^r = \omega + \sum_{d=1}^{D_r} \lambda_c^d y_l^{d,r}. \qquad (4)$$

Similarly, the coordinate ascent updates for the documents distribution over classes λ can be found by considering the terms in $\mathcal{L}$ that contain λ:

$$\begin{aligned}
\mathcal{L}_{[\lambda]} = &\sum_{d=1}^{D} \sum_{l=1}^{C} \lambda_l^d \eta_l^T \bar{\phi}^d - \sum_{l=1}^{C} \lambda_l^d \log \lambda_l^d \\
&+ \sum_{d=1}^{D} \sum_{r=1}^{R_d} \sum_{l=1}^{C} \sum_{c=1}^{C} \lambda_l^d \, y_c^{d,r} \, \mathbb{E}_q[\log \pi_{l,c}^r],
\end{aligned}$$

where $\bar{\phi}^d = \frac{1}{N_d} \sum_{n=1}^{N_d} \phi_n^d$. Adding the necessary Lagrange multipliers to ensure that $\sum_{l=1}^{C} \lambda_l^d = 1$ and setting the derivatives w.r.t. λ_l^d to zero gives the following update:

$$\lambda_l^d \propto \exp\Big(\eta_l^T \bar{\phi}^d + \sum_{r=1}^{R_d} \sum_{c=1}^{C} y_c^{d,r} \, \mathbb{E}_q[\log \pi_{l,c}^r] \Big). \qquad (5)$$

Observe how the variational distribution over the true classes results from a combination between the dot product of the inferred mean topic assignment $\bar{\phi}^d$ with the coefficients $\boldsymbol{\eta}$ and the labels $\mathbf{y}$ from the multiple annotators "weighted" by their expected log probability $\mathbb{E}_q[\log \pi_{l,c}^r]$.

The main difficulty of applying standard variational inference methods to the proposed model is the non-conjugacy between the distribution of the mean topic-assignment $\bar{z}^d$ and the softmax. Namely, in the expectation

$$\mathbb{E}_q[\log p(c^d|\bar{z}^d, \boldsymbol{\eta})] = \mathbb{E}_q[\eta_{c^d}^T \bar{z}^d] - \mathbb{E}_q\Big[\log \sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d) \Big],$$

the second term is intractable to compute. We can make progress by applying Jensen's inequality to bound it as follows:

$$\begin{aligned}
-\mathbb{E}_q\Big[\log \sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d) \Big] &\geqslant -\log \sum_{l=1}^{C} \mathbb{E}_q[\exp(\eta_l^T \bar{z}^d)] \\
&= -\log \sum_{l=1}^{C} \prod_{j=1}^{N_d} \big(\phi_j^d\big)^T \exp\Big(\eta_l \frac{1}{N_d} \Big) \\
&= -\log(a^T \phi_n^d),
\end{aligned}$$

where $a \triangleq \sum_{l=1}^{C} \exp(\frac{\eta_l}{N_d}) \prod_{j=1, j\neq n}^{N_d} \big(\phi_j^d\big)^T \exp\Big(\frac{\eta_l}{N_d}\Big)$, which is constant w.r.t. ϕ_n^d. This local variational bound can be made tight by noticing that $\log(x) \leqslant \epsilon^{-1} x + \log(\epsilon) - 1, \forall x > 0, \epsilon > 0$, where equality holds if and only if $x = \epsilon$. Hence, given the current parameter estimates $(\phi_n^d)^{old}$, if we set $x = a^T \phi_n^d$ and $\epsilon = a^T (\phi_n^d)^{old}$ then, for an individual parameter ϕ_n^d, we have that:

$$\begin{aligned}
&-\mathbb{E}_q\Big[\log \sum_{l=1}^{C} \exp(\eta_l^T \bar{z}^d) \Big] \\
&\qquad \geqslant -(a^T (\phi_n^d)^{old})^{-1} (a^T \phi_n^d) - \log(a^T (\phi_n^d)^{old}) + 1.
\end{aligned}$$

Using this local bound to approximate the expectation of the log-sum-exp term, and taking derivatives of the evidence lower bound w.r.t. ϕ_n with the constraint that $\sum_{i=1}^{K} \phi_{n,i}^d = 1$, yields the following fix-point update:

$$\begin{aligned}
\phi_{n,i}^d \propto \exp\Bigg(&\Psi(\gamma_i) + \sum_{j=1}^{V} w_{n,j}^d \Big(\Psi(\zeta_{i,j}) - \Psi\Big(\sum_{k=1}^{V} \zeta_{i,k} \Big) \Big) \\
&+ \frac{\sum_{l=1}^{C} \lambda_l^d \eta_{l,i}}{N_d} - (a^T (\phi_n^d)^{old})^{-1} a_i \Bigg). \qquad (6)
\end{aligned}$$

where V denotes the size of the vocabulary. Notice how the per-word variational distribution over topics ϕ depends on the variational distribution over the true class label λ.

The variational inference algorithm iterates between equations 2-6 until the evidence lower bound, eq. 1, converges. The supplementary material provides additional details on the derivation of this algorithm[2].

[2]Supplementary material available at:
https://dl.dropboxusercontent.com/u/1566445/supp-mat.pdf

Parameter estimation

The model parameters are $\Theta = \{\alpha, \tau, \omega, \boldsymbol{\eta}\}$. For the sake of simplicity we assume the parameters α, τ and ω of the Dirichlet priors to be fixed, and only estimate the coefficients $\boldsymbol{\eta}$ using a variational EM algorithm. Therefore, in the E-step we use the variational inference algorithm from section to estimate the posterior distribution of the latent variables, and in the M-step we find maximum likelihood estimates of $\boldsymbol{\eta}$ by maximizing the evidence lower bound $\mathcal{L}$. Unfortunately, taking derivatives of $\mathcal{L}$ w.r.t. $\boldsymbol{\eta}$ does not yield a closed-form solution, hence we use a numerical method, namely L-BFGS (Nocedal and Wright 2006), to find an optimum. The objective function and gradients are given by

$$\mathcal{L}_{[\eta]} = \sum_{d=1}^{D} \left(\sum_{l=1}^{C} \lambda_l^d \eta_l^T \bar{\phi}^d - \log \sum_{l=1}^{C} b_l^d \right)$$

$$\nabla_{\eta_{l,i}} = \sum_{d=1}^{D} \left(\lambda_{l,i}^d \bar{\phi}_i^d - \frac{b_l^d}{\sum_{t=1}^{C} b_t^d} \times \sum_{n=1}^{N_d} \frac{\frac{1}{N_d} \phi_{n,i}^d \exp(\frac{1}{N_d} \eta_{l,i})}{\sum_{j=1}^{K} \phi_{n,j}^d \exp(\frac{1}{N_d} \eta_{l,j})} \right),$$

where, for convenience, we defined the following variable: $b_l^d \triangleq \prod_{n=1}^{N_d} \left(\sum_{i=1}^{K} \phi_{n,i}^d \exp\left(\frac{1}{N_d} \eta_{l,i}\right)\right)$.

Stochastic variational inference

In the "approximate inference" section, we proposed a batch coordinate ascent algorithm for doing variational inference in the proposed model. This algorithm iterates between analyzing every document in the corpus to infer the local hidden structure, and estimating the global hidden variables. However, this can be inefficient for large datasets, since it requires a full pass through the data at each iteration before updating the global variables. In this section we develop a stochastic variational inference algorithm (Hoffman et al. 2013), which follows noisy estimates of the gradients of the evidence lower bound $\mathcal{L}$.

Based on the theory of stochastic optimization (Robbins and Monro 1951), we can find unbiased estimates of the gradients by subsampling a document (or a mini-batch of documents) from the corpus, and using it to compute the gradients as if that document was observed D times. Hence, given an uniformly sampled document d, we use the current posterior distributions of the global latent variables, $\boldsymbol{\beta}$ and $\boldsymbol{\pi}_{1:R}$, and the current coefficient estimates $\boldsymbol{\eta}$, to compute the posterior distribution over the local hidden variables θ^d, $\mathbf{z}^d$ and c^d using eqs. 2, 6 and 5 respectively. These posteriors are then used to update the global variational parameters, $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}_{1:R}$ by taking a step of size ρ_t in the direction of the noisy estimates of the natural gradients.

Algorithm 1 describes a stochastic variational inference algorithm for the proposed model. Given an appropriate schedule for the learning rates $\{\rho_t\}$, such that $\sum_t \rho_t$ and $\sum_t \rho_t^2 < \infty$, the stochastic optimization algorithm is guaranteed to converge to a local maximum of the evidence lower bound (Robbins and Monro 1951).

Algorithm 1 Stochastic variational inference

1: Initialize $\boldsymbol{\gamma}^{(0)}$, $\boldsymbol{\phi}_{1:D}^{(0)}$, $\boldsymbol{\lambda}^{(0)}$, $\boldsymbol{\zeta}^{(0)}$, $\boldsymbol{\xi}_{1:R}^{(0)}$, $t = 0$
2: **repeat**
3: Set t = t + 1.
4: Sample a document $\mathbf{w}^d$ uniformly from the corpus.
5: **repeat**
6: Compute ϕ_n^d using eq. 6, for $n \in \{1..N_d\}$.
7: Compute γ^d using eq. 2.
8: Compute λ^d using eq. 5.
9: **until** local parameters ϕ_n^d, γ^d and λ^d converge.
10: Compute step-size $\rho_t = (t + delay)^{-\kappa}$.
11: Update topics variational parameters

$$\zeta_{i,j}{}^{(t)} = (1 - \rho_t)\zeta_{i,j}^{(t-1)} + \rho_t \left(\tau + D \sum_{n=1}^{N_d} w_{n,j}^d \phi_{n,i}^d \right).$$

12: Update annotators confusion parameters

$$\xi_{c,l}^r{}^{(t)} = (1 - \rho_t)\xi_{c,l}^r{}^{(t-1)} + \rho_t \left(\omega + D\, \lambda_c^d\, y_l^{d,r}\right).$$

13: **until** global convergence criterion is met.

Document classification

In order to make predictions for a new (unlabeled) document d, we start by computing the approximate posterior distribution over the latent variables θ^d and $\mathbf{z}^d$. This can be achieved by dropping the terms that involve y, c and π from the model's joint distribution (since, at prediction time, the multi-annotator labels are no longer observed) and averaging over the estimated topics distributions. Letting the topics distribution over words inferred during training be $q(\boldsymbol{\beta}|\boldsymbol{\zeta})$, the joint distribution for a single document is now simply given by

$$p(\theta^d, \mathbf{z}^d) = \int q(\boldsymbol{\beta}|\boldsymbol{\zeta}) p(\theta^d|\alpha) \prod_{n=1}^{N_d} p(z_n^d|\theta^d) p(w_n^d|z_n^d, \boldsymbol{\beta}) d\boldsymbol{\beta}.$$

Deriving a mean-field variational inference algorithm for computing the posterior over $q(\theta^d, \mathbf{z}^d) = q(\theta^d|\gamma^d) \prod_{n=1}^{N_d} q(z_n^d|\phi_n^d)$ results in the same fixed-point updates as in LDA (Blei, Ng, and Jordan 2003) for γ_i^d and $\phi_{n,i}^d$. Using the inferred posteriors and the coefficients $\boldsymbol{\eta}$ estimated during training, we can make predictions as follows:

$$c_*^d = \arg\max_c \eta_c^T \bar{\phi}^d. \tag{7}$$

This is equivalent to making predictions in sLDA (Wang, Blei, and Fei-Fei 2009).

Experiments

In this section, the proposed model, multi-annotator supervised LDA (MA-sLDA), is validated using both simulated annotators on popular corpora and using real multiple-annotator labels obtained from Amazon Mechanical Turk.[3]

[3]Source code and datasets used are available at: ADD URL

Simulated annotators

In order to first validate the proposed model in a slightly more controlled environment, the well-known 20-Newsgroups benchmark corpus (Lang 1995) was used by simulating multiple annotators with different levels of expertise. The 20-Newsgroups consists of twenty thousand messages taken from twenty newsgroups, and is divided in six super-classes, which are, in turn, partitioned in several subclasses. For this first set of experiments, only the four most populated super-classes were used: "computers", "science", "politics" and "recreative". The preprocessing of the documents consisted of stemming and stop-words removal. After that, 75% of the documents were randomly selected for training and the remaining 25% for testing.

The different annotators were simulated by sampling their answers from a multinomial distribution, where the parameters are given by the lines of the annotators' confusion matrices. Hence, for each annotator r, we start by pre-defining a confusion matrix $\boldsymbol{\pi}^r$ with elements $\pi^r_{c,l}$, which correspond to the probability that the annotators' answer is l given that the true label is c, $p(y^r_i = l|c)$. Then, the answers are sampled i.i.d. from $y^r_i \sim Mult(\pi^r_{c,l})$. This procedure was used to simulate 5 different annotators with the following accuracies: 0.737, 0.468, 0.284, 0.278, 0.260. In this experiment, no repeated labelling was used. Hence, each annotator only labels roughly one-fifth of the data. When compared to the ground truth, the simulated answers revealed an accuracy of 0.405. See Table 2 for an overview of the details of the datasets used.

Both the *batch* and the stochastic variational inference (*svi*) versions of the proposed model (MA-sLDA) are compared with the following baselines:

- *LDA + LogReg (mv)*: This baseline corresponds to applying unsupervised LDA to the data, and learning a logistic regression classifier on the inferred topics distributions of the documents. The labels from the different annotators were aggregated using majority voting (mv).
- *LDA + Raykar*: For this baseline, the model of (Raykar et al. 2010) was applied using the documents' topic distributions inferred by LDA as features.
- *LDA + Rodrigues*: This baseline is similar to the previous one, but uses the model of (Rodrigues, Pereira, and Ribeiro 2013) instead.
- *Blei 2003 (mv)*: The idea of this baseline is to replicate a popular state-of-the-art approach for document classification. Hence, the approach of (Blei, Ng, and Jordan 2003) was used. It consists of applying LDA to extract the documents' topics distributions, which are then used to train a SVM. Similarly to the previous approach, the labels from the different annotators were aggregated using majority voting (mv).
- *sLDA (mv)*: This corresponds to using sLDA (Wang, Blei, and Fei-Fei 2009) with the labels obtained by performing majority voting (mv) on the annotators' answers.

For all the experiments the hyper-parameters α, τ and ω were set using a simple grid search in the collection $\{0.01, 0.1, 1.0, 10.0\}$. The same approach was used to optimize the hyper-parameters of the all the baselines. For the *svi* algorithm, different mini-batch sizes and forgetting rates κ were tested. For the 20-Newsgroup dataset, the best results were obtained with a mini-batch size of 500 and $\kappa = 0.6$. The $delay$ was kept at 1. The results are shown in Figure 2 for different numbers of topics, where we can see that the proposed model outperforms all the baselines, being the *svi* version the one that performs best.

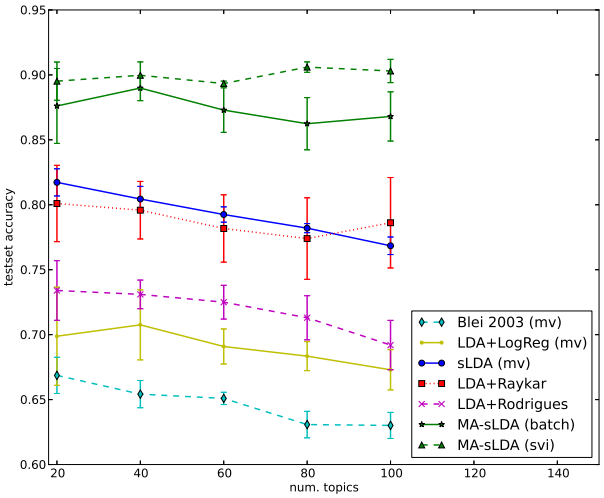


Figure 2: Average testset accuracy (over 5 runs; $\pm$ stddev.) of the different approaches on the 20-Newsgroups data.

In order to assess the computational advantages of the stochastic variational inference (*svi*) over the *batch* algorithm, the log marginal likelihood (or log evidence) was plotted against the number of iterations. Figure 3 shows this comparison. Not surprisingly, the *svi* version converges much faster to higher values of the log marginal likelihood when compared to the *batch* version, which reflects the efficiency of the *svi* algorithm.

Amazon Mechanical Turk

In order to validate the proposed model in a real crowdsourcing setting, Amazon Mechanical Turk (AMT) was used to obtain labels from multiple annotators for two popular datasets: Reuters-21578 (Lewis 1997) and LabelMe (Russell et al. 2008).

Reuters-21578 is a collection of manually categorized newswire stories with labels such as Acquisitions, Crude-oil, Earnings or Grain. For this experiment, only the documents belonging to the ModApte split were considered with the additional constraint that the documents should have no more than one label. This resulted in a total of 7016 documents distributed among 8 classes. Of these, 1800 documents were submitted to AMT for multiple annotators to label, giving an average of 3.007 answers per document (see Table 2 for further details). The remaining 5216 documents were used for testing. The collected answers yield an average annotator ac-

Dataset	Num. classes	Train/test sizes	Annotators source	Num. answers per instance ($\pm$ stddev.)	Mean annotators accuracy ($\pm$ stddev.)	Maj. vot. accuracy
20 Newsgroups	4	11536/3846	Simulated	1.000 ± 0.000	0.405 ± 0.182	0.405
Reuters-21578	8	1800/5216	Mech. Turk	3.007 ± 1.019	0.568 ± 0.262	0.710
LabelMe	8	1000/1688	Mech. Turk	2.547 ± 0.576	0.692 ± 0.181	0.769

Table 2: Overall statistics of the datasets used in the experiments.

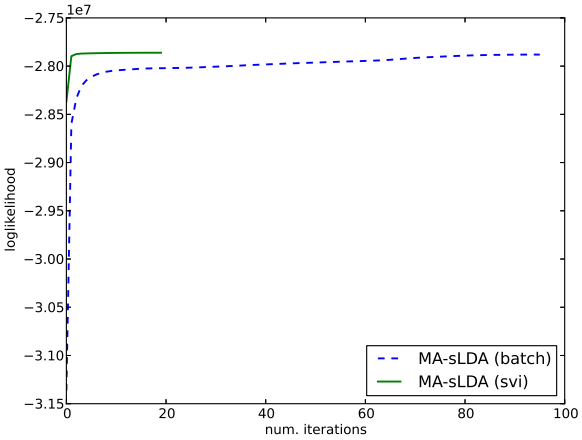


Figure 3: Comparison of the log marginal likelihood between the *batch* and the stochastic variational inference (*svi*) algorithms on the 20-Newsgroups corpus.

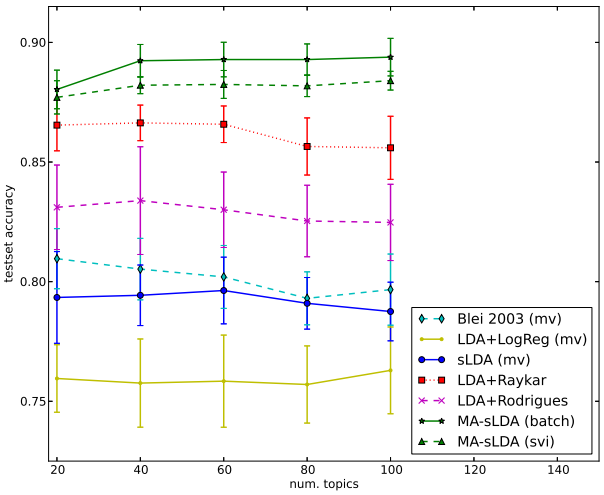


Figure 4: Average testset accuracy (over 30 runs; $\pm$ stddev.) of the different approaches on the Reuters data.

curacy of 56.8%. Applying majority voting to these answers reveals a ground truth accuracy of 71.0%.

The results obtained by the different approaches are given in Figure 4, where it can be seen that the proposed model (MA-sLDA) outperforms all the other approaches. For this dataset, the *svi* algorithm is using mini-batches of 300 documents.

The proposed model is also validated using a dataset from the computer vision domain: LabelMe (Russell et al. 2008). In contrast to the Reuters and Newsgroups corpora, LabelMe is an open online tool to annotate images. Hence, this experiment allows us to see how the proposed model generalises beyond non-textual data. Using the provided Matlab interface, we extracted a subset of the LabelMe data, consisting of all the 256 x 256 images with the categories: "highway", "inside city", "tall building", "street", "forest", "coast", "mountain" or "open country". This allowed us to collect a total of 2688 labeled images. Of these, 1000 images were given to AMT workers to classify with one of the classes above. Each image was labeled by an average of 2.547 workers, with a mean accuracy of 69.2%. When majority voting is applied to the collected answers, a ground truth accuracy of 71.0% is obtained.

The preprocessing of the images used is similar to the approach of (Fei-Fei and Perona 2005). It uses 128-dimensional SIFT (Lowe 1999) region descriptors selected by a sliding grid spaced at one pixel. This sliding grid extracts local regions of the image with sizes uniformly sampled between 16 x 16 and 32 x 32 pixels. The 128-dimensional SIFT descriptors produced by the sliding window are then fed to a k-means algorithm (with k=200) in order construct a vocabulary of 200 "visual words". This allows us to represent the images with a bag of visual words model.

With the purpose of comparing the proposed model with a popular state-of-the-art approach for image classification, for the LabelMe dataset, the following baseline was introduced:

- *Bosch 2006 (mv)*: This baseline is similar to one in (Bosch, Zisserman, and Muñoz 2006). The authors propose the use of pLSA to extract the latent topics, and the use of k-nearest neighbor (kNN) classifier using the documents' topics distributions. For this baseline, unsupervised LDA is used instead of pLSA, and the labels from the different annotators for kNN (with $k = 10$) are aggregated using majority voting (mv).

The results obtained by the different approaches for the LabelMe data are shown in Figure 5, where the *svi* version is using mini-batches of 200 documents.

Analyzing the results for the Reuters-21578 and LabelMe data, we can observe that the proposed model outperforms all the baselines, with slightly better accuracies for the *batch* version, especially in the Reuters data. Interestingly, the second best results are consistently obtained by the multi-

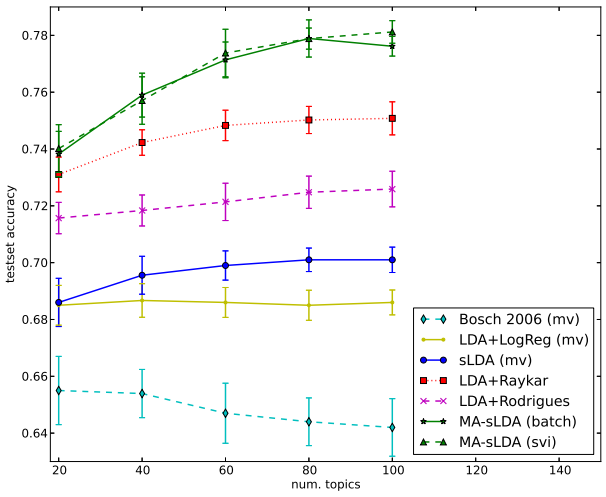


Figure 5: Average testset accuracy (over 30 runs; $\pm$ stddev.) of the different approaches on the LabelMe data.

annotator approaches, which highlights the need for accounting for the noise and biases of the answers of the different annotators.

Conclusion

This paper proposed a supervised topic model that is able to learn from multiple annotators and crowds, by accounting for their biases and different levels of expertise. Given the large sizes of modern datasets, and considering that the majority of the tasks for which crowdsourcing and multiple annotators are desirable candidates, generally involve complex high-dimensional data such as text and images, the proposed model constitutes a strong contribution for the multi-annotator paradigm. This model is then capable of jointly modeling the words in documents as arising from a mixture of topics, as well as the latent true labels and the (noisy) labels of the multiple annotators. We empirically showed, using simulated annotators on the 20-Newsgroups dataset and using real annotators from Amazon Mechanical Turk for Reuters-21578 and LabelMe data, that the proposed model is able to outperform state-of-the-art approaches. Finally, an efficient stochastic variational inference algorithm was described, which gives the proposed model the ability to scale to large datasets.

Given that the target variables associated with documents can be continuous, and also considering that documents can sometimes belong to more than one class, future work will explore the extension of the proposed model to regression and multi-label classification problems.

References

Airoldi, E.; Blei, D.; Fienberg, S.; and Xing, E. 2007. Combining stochastic block models and mixed membership for statistical network analysis. In *Statistical Network Analysis: Models, Issues, and New Directions*. Springer. 57–74.

Blei, D.; Ng, A.; and Jordan, M. I. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.* 3:993–1022.

Bosch, A.; Zisserman, A.; and Muñoz, X. 2006. Scene classification via plsa. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer. 517–530.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society. Series C* 28(1):20–28.

Erosheva, E.; Fienberg, S.; and Joutard, C. 2007. Describing disability through individual-level mixture models for multivariate binary data. *The annals of applied statistics* 1(2):346.

Fei-Fei, L., and Perona, P. 2005. A Bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, 524–531. IEEE.

Hoffman, M.; Blei, D.; Wang, C.; and Paisley, J. 2013. Stochastic variational inference. *J. Mach. Learn. Res.* 14(1):1303–1347.

Jordan, M. I.; Ghahramani, Z.; Jaakkola, T.; and Saul, L. 1999. An introduction to variational methods for graphical models. *Mach. Learn.* 37(2):183–233.

Lacoste-Julien, S.; Sha, F.; and Jordan, M. 2009. Disclda: Discriminative learning for dimensionality reduction and classification. In *Advances in neural information processing systems*, 897–904.

Lang, K. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, 331–339.

Lewis, D. 1997. Reuters-21578 text categorization test collection, distribution 1.0.

Lowe, D. 1999. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, 1150–1157. Ieee.

Mcauliffe, J., and Blei, D. 2008. Supervised topic models. In *Advances in neural information processing systems*, 121–128.

Niebles, J.; Wang, H.; and Fei-Fei, L. 2008. Unsupervised learning of human action categories using spatial-temporal words. *Int. J. Comput. Vision* 79(3):299–318.

Nocedal, J., and Wright, S. 2006. *Numerical Optimization*. World Scientific.

Ramage, D.; Hall, D.; Nallapati, R.; and Manning, C. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 248–256. Association for Computational Linguistics.

Raykar, V.; Yu, S.; Zhao, L.; Valadez, G.; Florin, C.; Bogoni, L.; and Moy, L. 2010. Learning from Crowds. *Journal of Machine Learning Research* 1297–1322.

Robbins, H., and Monro, S. 1951. A stochastic approximation method. *Ann. Math. Statist.* 22(3):400–407.

Rodrigues, F.; Pereira, F.; and Ribeiro, B. 2013. Learning from multiple annotators: distinguishing good from random labelers. *Pattern Recognition Letters* 1428–1436.

Rodrigues, F.; Pereira, F.; and Ribeiro, B. 2014. Gaussian process classification and active learning with multiple annotators. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 433–441.

Russell, B.; Torralba, A.; Murphy, K.; and Freeman, W. 2008. Labelme: a database and web-based tool for image annotation. *International journal of computer vision* 77(1-3):157–173.

Smyth, P.; Fayyad, U.; Burl, M.; Perona, P.; and Baldi, P. 1995. Inferring ground truth from subjective labelling of venus images. In *Advances in Neural Information Processing Systems*, 1085–1092.

Snow, R.; O'Connor, B.; Jurafsky, D.; and Ng, A. 2008. Cheap and fast - but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing*, 254–263.

Wang, C.; Blei, D.; and Fei-Fei, L. 2009. Simultaneous image classification and annotation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 1903–1910. IEEE.

Welinder, P.; Branson, S.; Perona, P.; and Belongie, S. 2010. The multidimensional wisdom of crowds. In *Advances in neural information processing systems*, 2424–2432.

Yan, Y.; Rosales, R.; Fung, G.; Schmidt, M.; Valadez, G.; Bogoni, L.; Moy, L.; and Dy, J. 2010. Modeling annotator expertise: Learning when everybody knows a bit of something. *Journal of Machine Learning Research* 9:932–939.

Yan, Y.; Rosales, R.; Fung, G.; Subramanian, R.; and Dy, J. 2014. Learning from multiple annotators with varying expertise. *Mach. Learn.* 95(3):291–327.

Zhu, J.; Ahmed, A.; and Xing, E. 2012. Medlda: Maximum margin supervised topic models. *J. Mach. Learn. Res.* 13(1):2237–2278.