Author and Topic Modelling in Text Data

Rasmus Troelsgård

Kongens Lyngby 2012
IMM-M.Sc.-2012-67

Technical University of Denmark
Informatics and Mathematical Modelling
Building 321, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk IMM-M.Sc.-2012-67

Summary (English)

This thesis deals with probabilistic modelling of authors, documents, and topics in textual data. The focus is on the Latent Dirichlet Allocation (LDA) model and the Author-Topic (AT) model where Gibbs sampling is used for inferring model parameters from data. Furthermore, a method for optimising hyper parameters in an ML-II setting is described.

Model properties are discussed in connection with applications of the models which include detection of unlikely documents among scientific papers from the NIPS conferences using document perplexity, and the problem of link prediction in the online social network Twitter for which the results are reported as Area Under the ROC curve (AUC) and compared to well known graph-based methods.

Summary (Danish)

Denne afhandling forsøger at give en beskrivelse statistiske modeller for dokumenter, forfattere og emner i tekstdata.
Fokus er på Latent Dirichlet Allocation (LDA) og Author-Topic modellen hvori Gibbs sampling er brugt som inferensmetode. Derudover er en metode til optimering af hyperparametre blevet beskrevet.
Modellernes egenskaber bliver diskuteret i forbindelse med eksempler på brug af modellerne. Disse eksempler omfatter detektion af usandsynlige dokumenter i et korpus bestående af videnskabelige artikler fra NIPS konferencerne, og et venne-anbefalingssystem i forbindelse med det sociale medie Twitter, inklusiv sammenligning med graf-baserede metoder.

Preface

This thesis was prepared at Section for Cognitive Systems at the department of Informatics and Mathematical Modelling at the Technical University of Denmark in fulfilment of the requirements for acquiring the Master of Science degree in Mathematical Modelling and Computation.
The work presented here has been carried out in the period from February 1st to July 2nd 2012.

The objective for the master project was to explore, analyse and experiment with methods for statistical modelling of authors and topics in text data. The work performed during the project period has been a mixture of literature study, model analysis, software implementation, data pre-processing, and experimental work and documentation.

The project was supervised by Prof. Lars Kai Hansen, Head of Section for Cognitive Systems at DTU Informatics. I am very appreciative of his help and of our stimulating discussions. Furthermore, I would like to thank Senior researcher Finn Årup Nielsen for his help with making the Twitter data available.

Lyngby, 02-July-2012

Rasmus Troelsgård

Contents

Chapter 1

Introduction

Thanks to digitisation of old material, registration of new material, sensor data and both governmental and private digitisation strategies in general, the amount of data available of all sorts has been expanding and increasing for the last decade. Simultaneously, the need for automatic data organisation tools and search engines has become obvious. Naturally, this has lead to an increased scientific interest and activity in related areas such as pattern recognition and dimensionality reduction. The Internet has revolutionised the way people are able to communicate and store and share information. Both in private and public contexts.

A lot of the available data was and still is text. A typical text dataset consists of a number of documents, which are basically lists of words. A widely used group of models does not take the order in which the words appear in the documents into account. This assumption is often referred to as "bag-of-words". Many words, especially nouns and verbs that only seldom occur outside a limited number of contexts, have one specific meaning or at least only a few, not depending on the position in the text. To capture the essence and describe the topical aspects of a document collection, the bag-of-words assumption might be acceptable. Of course, vital information about the specific meaning of the text is lost in such a simplification. The main advantages of bag-of-words models are their simplicity in the description and obvious lack of dependency on word order.

The idea of analysing text data by deriving a representation in a lower dimensional space of “topics” or “aspects” was followed by [DFL+88], who in the late 1980s proposed one of the most basic approaches to topic modelling, called LSA or LSI. This method is based on the theory of linear algebra and uses the bag-of-words assumption. The core of the method is to apply SVD to the co-occurrence count matrix of documents and terms (often referred to as the term-document matrix), to obtain a reduced dimensionality representation of the documents.

In 1999 Thomas Hofmann suggested a model called probabilistic Latent Semantic Indexing (pLSI/pLSA) [Hof99] in which the topic distributions over words were still estimated from co-occurrence statistics within the documents, but introduced the use of latent topic variables in the model. pLSI is a probabilistic method, and has shown itself superior to LSA in a number of applications, including Information Retrieval (IR). Since then there have been an increasing focus on using probabilistic modelling as a tool rather than using linear algebra. According to Blei et al.[BNJ03], pLSI has some shortcomings with regard to overfitting and generation of new documents. This was one of the motivating factors to propose Latent Dirichlet Allocation (LDA) [BNJ03], a model that quickly became very popular, and has since been widely used and modified to fit countless specific tasks within areas of IR, Data Mining, Natural Language Processing (NLP) and related topics.

In contrast to the models relying on the bag-of-words data representation, another major class of models based on specifically capturing the word order, exists. These models are primarily applied within the field of natural language processing, and thus focus more on the local structure and flow of language. Examples of such models are traditional language models based on bi- and tri-grams and Hidden Markov Models in various forms. This field of research has been very popular in the latest decades, and numerous extensions and combinations of models have been developed and described. Examples of such combinations of traditional language models and topic models are [HG06] [Wal06] [GH99] [GSBT05].

“Topic model” or “aspect model” are generic terms for models capable of describing data by means of discrete probability distributions over smaller components forming the dataset. Thus it is worth mentioning that topic models are not limited to analysis of textual data; they can and have also been used to describe various other types of data such as images, video, audio [LMD10] [WM09] [RFE+06] [KNS09] [HGX09] ([BNJ03]). In other words, all kinds of data that have an inherent grouping (the documents) of features (the words) having different statistical properties. One of the advantages of topic models over simpler clustering models is the ability to model mixed membership of different classes.

Working with topic models, there is no guarantee that the estimated topics are semantically meaningful to humans, and often this is not a criterion of success. One is likely to be blinded by the desire to make the model "understand" the data in a humanly comprehensible way even though it is most sensible that performance is measured by the task for which the system is built.

Topic models have been applied to a huge variety of areas, and this thesis will uncover some ground in the usage of topic models too. In the present work, I will explore and discuss the properties of the LDA model and one of its derivatives, the Author-Topic (AT) model [RZCG$^+$10]. These models are fully generative, meaning that new documents can be generated from the set of model parameters. Multiple possible methods for parameter inference in the models exist. The most popular are variational Bayes (VB) [BNJ03], collapsed VB [TNW07], collapsed Gibbs sampling [GS04], expectation propagation (EP) [ML02] and belief propagation [ZCL11]. In the recent years, a considerable amount of work has been put into making the parameter estimation algorithms more efficient, and several papers dealing with parallelisation of existing inference techniques, and methods for on-line inference, have been published [SN10, NASW09, YMM09, HBB10].

LDA and the AT model will be examined theoretically and tested using both synthetic data, real world data from the NIPS conference, and data from the on-line social network Twitter. In particular, this study will treat the AT model in a setting of outlier-detection in document collections, i.e. discovering false author attributions by measuring how likely it is that a particular document is written by the stated author. A closely related task is to find authors that are likely to have written a particular document. This can be used in the case of missing author information and use of pseudonyms. Examples of this usage is mentioned in [SSRZG04].

The other application of LDA and AT treated in this thesis is the task of link prediction in the Twitter network. It can easily be realised that using solely topic models for this task is inadequate and inferior to models also including overt information such as the existing graph structure. This is particularly true for huge networks, as there may exist many different communities clearly separated according to the graph structure, but having significant topical similarities. For the task of predicting future links, intra-community links might perform best, whereas in the case of link recommendation systems, inter-community links will help people to find and connect to people with similar interests. The data used in this work consist of rather small sub-networks of the Twitter graph, thus topical similarity might perform acceptably for the link prediction task as well. Other works [PG11, PECX10] have focused on the use of LDA for link prediction in the Twitter graph, and the contribution of this thesis will mainly be an investigation of the usability of the AT model for this task. This is done by augmenting the original tweet with extra author information from the tweet itself, such as "@mention"s, "@reply"s and "retweet".

The analyses performed in this work should be seen as an attempt to explore the influence on prediction performance, of differences between model structures and settings of the models LDA and AT. This will act as a guideline to what features are important when considering using topic models for predicting and recommending new links, which is of considerable importance in many business areas and in particular for on-line service providers. The results presented are based on a quite small sample of data from Twitter, and thus this study will not draw any conclusions regarding Twitter in general, but should merely be seen as a pilot study.
All results presented in this thesis are generated using a basic collapsed Gibbs sampling scheme. Section 4.1 includes exploratory analyses of the behaviour of the Gibbs sampler for LDA under the conditions of varying sizes of training corpora.

1.1 Related Work

As mentioned above, topic models and LDA in particular have been applied in numerous research areas. Other work that deals with problems similar to outlier detection includes usage examples of the AT model mentioned in [RZCG^{+}10] where examples of finding unusual papers for authors in a scientific paper collection, are given. Several papers describe methods for matching peer reviewers to scientific papers by use of topic models. This includes early work using LSI [DN92], and extensions of LDA like the author-persona-topic model [ACM07].
Another task for which topic models have been put to use is link-prediction or network completion, i.e. the task of recovering the network structure from a partly- or non-observed network. This field of research is flourishing, as more and more network data have been collected and also generated by means of the Internet. Several approaches to this task use the observed part of the network to predict the missing parts [CMN08, KL11, YHD11].
Weng et al. [WLJH10] uses LDA to show the existence of topical homophily in Twitter user communities. This is crucial for the success of using topic models for link prediction in the Twitter network.
[PG11] uses a pure topic model approach, using LDA to recommend new links to users in the on-line social network Twitter. This approach is very similar to this work, but is far less thorough.
Similar is also the work by [PECX10], studying the use of LDA for predicting links in both the explicit follower graph in Twitter and a “message graph” where edges are present if personal messages have been sent between two nodes.
[HD10] performs an empirical study of LDA and AT using Twitter data, experimenting with the use of topic models for classification of popular messages.
The Topic-Link LDA [LNMG09] combines information of the network structure

and the content associated with the nodes in a single model used for predicting links. A similar approach is taken by [NC08].

Chapter 2

Datasets

This chapter contains information about the two real-world data sets used in the experiments in this thesis.

2.1 NIPS Data Set

The NIPS dataset used in this work has been extracted from the Matlab data file `nips12raw_str602.mat`, obtained from `http://cs.nyu.edu/ roweis/data/`. It contains 1740 documents written by 2038 different authors, utilising a vocabulary size of 13649 unique words. The total number of word tokens in the corpus is 2,301,375. Sorted in descending order, the number of documents each author has participated in writing, is shown in figure 2.1. The dataset available is already preprocessed including removal of so called stop-words.

As there are more authors than documents the author-document assignment matrix is quite sparse, which could make it hard to infer something about each author.

There are minor errors in the NIPS dataset as also mentioned in [RZCG+10]. For instance, the two authors “Martin I. Sereno” and “Margaret E. Sereno” have been mapped to the same author id “Sereno_M” (see file: `nips03/0320.txt`). This is the only error that has been corrected in the data used for this work.

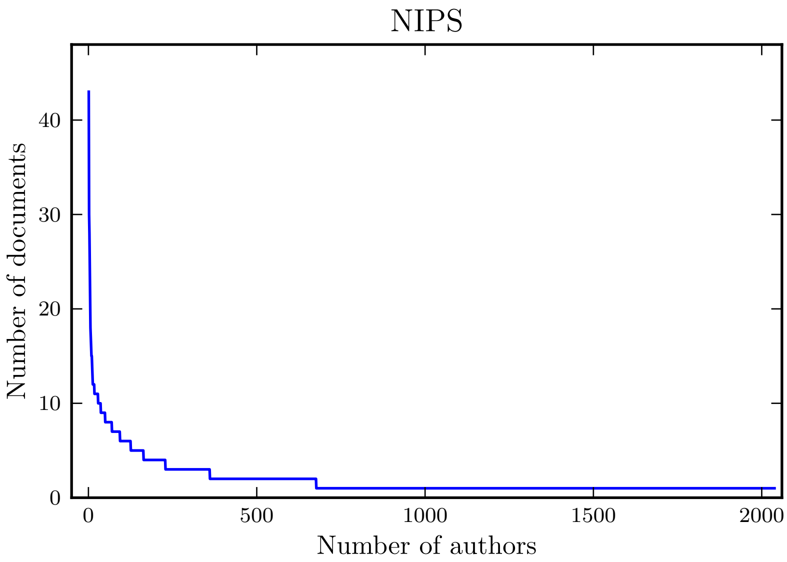


Figure 2.1: Number documents each of the authors have (co-)authored. Only 124 out of the 2038 authors have participated in writing more than 5 papers over a long period.

2.2 Twitter Dataset

The Twitter dataset used here consists of a snapshot of the graph [KLPM10], and tweets collected in the last seven months of 2009 [YL11]. The dataset is estimated to contain 20-30% of all tweets from that period.
Only 9447016 users that have written tweets are also present in the graph. This means that this is the maximum number of nodes for which we have all three types of information; tweets, graph, and userid/screenname correspondence, and hence this is the dataset used in the thesis. The distribution of number of tweets per user is very skewed, meaning that relatively few users have posted thousands of tweets, while the majority have been far less active. See Figure 2.2 illustrating the distribution.
Figures 2.3 and 2.4 show all the users in a “number of followers”/“number of followees”-coordinate system, and their corresponding number of tweets/posted messages is given by the colour. Here the term “followee” denotes a user that is being followed

To be able to handle, and model the data within a foreseeable time frame, several sub-networks are extracted from the full dataset. As the data are going

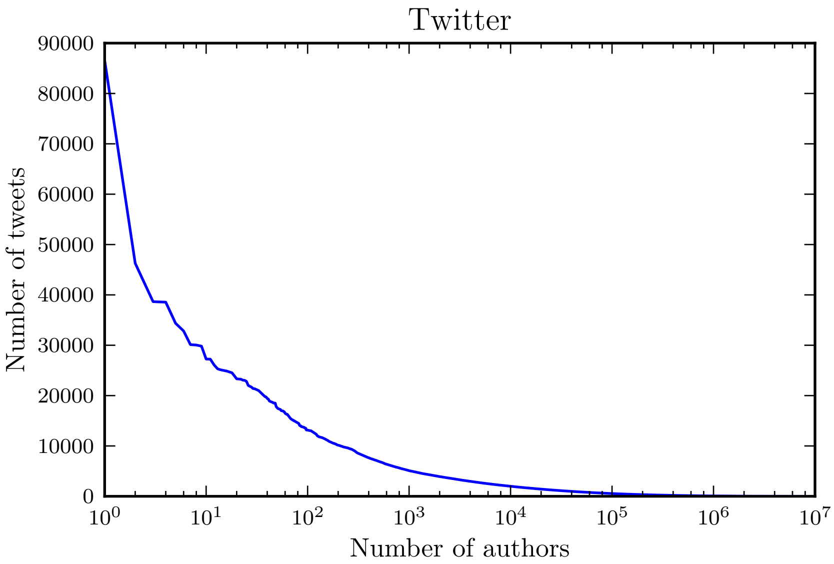


Figure 2.2: Number tweets each of the users have posted. 9078865 out of the 9447016 users have written less than 200 tweets in the time period covered by the dataset.

to be used with topic model, to estimate different authors topical preferences, some minimum amount of data has to be available for each author. See section 2.2.1 detail about the sub graph extraction criteria.

2.2.1 Extracting Sub Graphs

Each sub-network is grown from a seed user. The seed users are limited to the set of users fulfilling the following criteria:

1. $\min(c_{in}, c_{out}) > |c_{in} - c_{out}|$: There has to be a some balance in the number of inbound and outbound connections, denoted c_{in} and c_{out} respectively.

2. $5 \leq c_{in} < 500$ and $5 \leq c_{out} < 500$

3. has written more than 100 tweets.

Each sub-network is grown from to include all nodes with a minimal link distance to the seed node of less than 3, only including nodes that have written more than 100 tweets and have less than 500 in- or out-bound connections. This can be stated more formally:
Let X be a set of users and let $\mathcal{F}(X)$ be the union of all the sets of followers of the users in X and all the sets of users who are followed by a user in X. Furthermore, let $\kappa(X, n)$ be a function that removes users, with less than n published tweets, from X. An likewise, let $\zeta(X, c)$ be a function that removes users with more than c in- or out-bound connections, from the set X. Then the sub network $N(S)$ grown from "seed set" S, only containing a single element (the seed), can be defined as

$$N(S) = \kappa\left(\zeta\left(\mathcal{F}\left(\zeta(\mathcal{F}(S), c)\right), c\right), n\right) \tag{2.1}$$

The cap on the number of connections $c = 500$ is set as an attempt to extract networks with more personal relationships amongst the nodes. The idea of setting a limit comes from Robin Dunbar's famous theory, that humans can only maintain a certain number of stable social relations. [GPV11] have validated the limiting phenomenon of Dunbar's number in the context of the social network of twitter.
The minimum number of tweets is set to $n = 100$ to ensure that all nodes have some minimal amount of data associated with it. The lenghts of the tweets of the remaining authors are not checked, which means that there is a potential risk that only n words are available for a specific author, and this will probably produce a poor estimate of the particular author's distribution over topics, and hence have a negative influence on the prediction of links to/from that particular node. Furthermore, to be able to run the Gibbs sampling algorithm within an acceptable time slot, only 10 networks consisting of less than 4000 nodes are picked at random from the networks grown following the described criteria.
Table 2.1 shows information on the extracted sub-networks.

2.2.2 Text Pre-processing

As all other natural language, to the computer tweets are just lists of characters, and thus have to be preprocessed to become available for models relying on a representation of texts as word tokens. This process is called tokenisation. The tweets used in this theses are passed through a tokeniser written by Christopher Potts [Pot11]. The tokeniser recognises a number of entity types including "emoticons" (multiple kinds of smileys), URLs, phone numbers, and dates.
To take up the least amount of characters in a tweet, URLs are often available

Name	Seed id	N_A	N_{con}	$\frac{N_{con}}{N_{con}+N_{open}}$	N_{tweets}	N_{tokens}
N1	46582416	869	18845	0.049967	526948	9077788
N2	32345729	874	10668	0.027963	363282	6434665
N3	46159288	822	12412	0.036784	475541	8411009
N4	16178300	1370	24689	0.026327	870147	16156280
N5	25770884	654	4952	0.023191	365288	6396972
N6	48242051	1522	30965	0.026752	848638	14454282
N7	56095948	604	1986	0.010906	611028	10775981
N8	34655473	1152	22866	0.034490	544111	9843801
N9	17915633	3193	89485	0.017560	1695477	29563299
N10	24557123	1179	24344	0.035056	673233	11913903

Table 2.1: Specific Sub-data-sets. “Seed id” is the official Twitter user id of the seed node from which the network is grown. All connections are followed up to a distance of two levels of separation from the seed node, excluding nodes with less than 100 tweets or more than 500 in- or out-bound connections. N_{con} is the number of (undirected) edges in the graph, and N_{open} is the number of non-existing edges. The number of possible connections in the graph is $N_{con} + N_{open} = \frac{N_a*(N_a-1)}{2}$. N_{tweets} is the total number of tweets in the dataset, and N_{tokens} is the total number of tokens in the tweets.

through shorter link-aliases, ending some kind of hash-code e.g. http://t.co/OKXHq3IH. Everything after the top level domain-name of URLs is removed to avoid having a lot of URLs appearing only once in the corpus. The top level domain-name is kept e.g. http://t.co, as it might contain some information on the usage of different link-shortening-services.

All tokens appearing only once in the dataset are removed, and tweets that have become empty in this process are removed from the corpus. Thus there is no guarantee that all authors have at least 100 tweets in the corpus when the pre-processing step is finished.

2.2.3 Dataset Peculiarities

Most user names are shorter than 16 characters, but some user names are up to 20 characters in lenght even though the current limit for username length is 15 characters.

At least two user names contain a blank space character (“adam cary” and “marie äilyñ”) although the current rules for username creation does not allow this [Twi12].

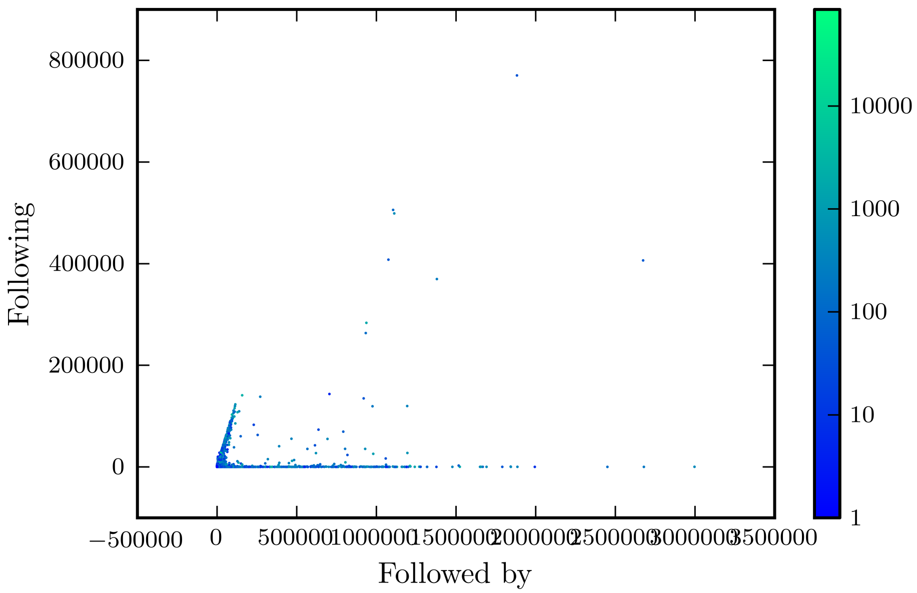


Figure 2.3: Each point in the plane corresponds to a specific user's number of followers (horizontal axis) and the number of people the user is following (the user's followees)(vertical axis). The colour denotes the number of tweets posted by the user, thus it can be seen as a measure of activity. Note that the colour scale is logarithmic. It seems there are two characteristic modes in the data; people who are extremely popular and are followed by thousands of people, but who are only themselves following relatively few others. One theory explaining this could be that these users are celebrities and news media profiles. The other obvious group of users have a very well balanced follower/followee ratio, and a lot of the users in this group have far more connections than could possibly be maintained in a personal manner. Thus the user profiles in the upper part of this group are probably managed automatically in some way, and programmed to follow back everybody who follows them.

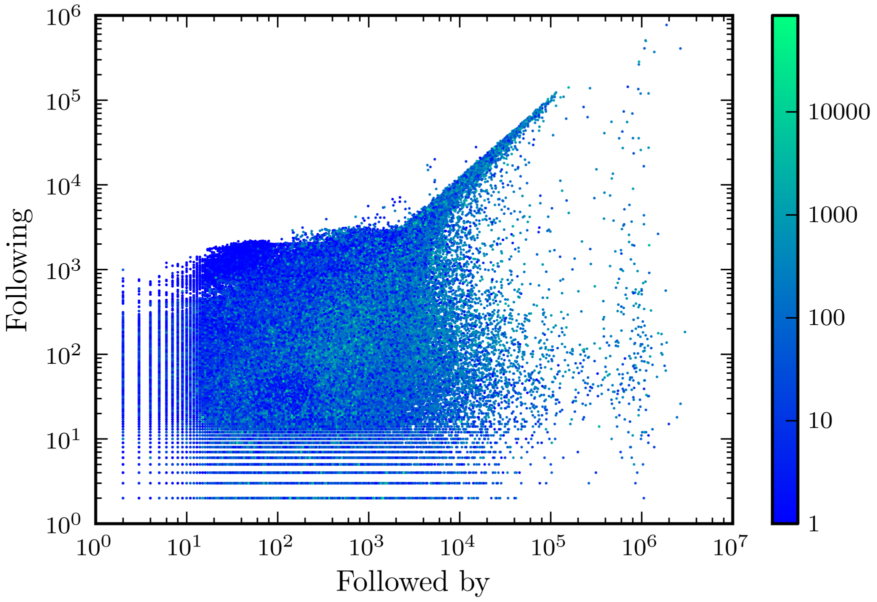


Figure 2.4: Each point in the plane corresponds to a specific user's number of followers (horizontal axis) and followees (vertical axis). The colour denotes the number of tweets posted by the user, thus it can be seen as measure of activity. Note that all scales are logarithmic. Comparing to figure 2.3, this figure indicates that the density of user profiles with a balanced follower/followee ratio is higher than in the “celebrity”-cluster along the horizontal axis. This effect is seen even clearer in figure 2.5.

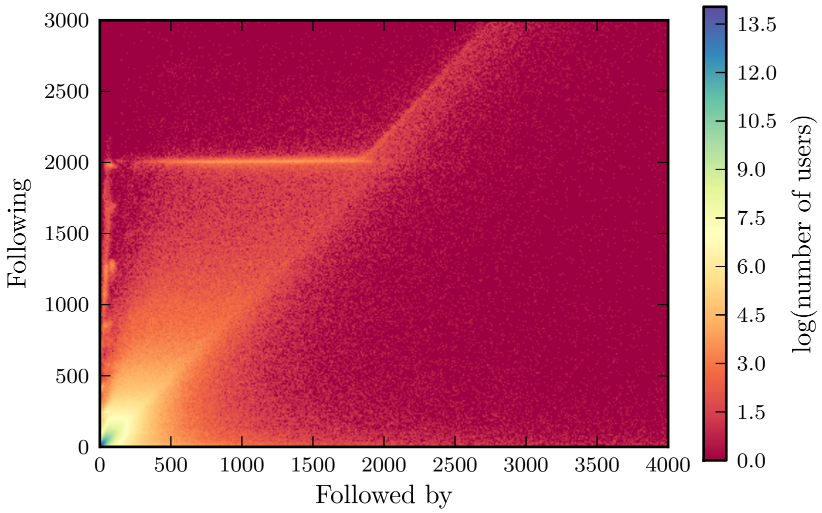


Figure 2.5: Small segment of a fine grained 2D histogram of the number of followers and number of followees. The bin size is 10. The colour denotes the density of users in each bin (log scale). In this figure, one can still make out the diagonal cluster of user profiles, but only vaguely the horizontal. Furthermore, also a nearly vertical cluster and a horizontal one, corresponding to users following 2000 others, catch the eye. The horizontal cluster is supposedly people/robots who have reached Twitter's follow limit, while the vertical is harder to account for. One guess is that these users follow more people than just their friends (for example news media and politicians) but are themselves, to a large extent, only followed by their friends.

Chapter 3

Topic Model Theory

3.1 Latent Dirichlet Allocation

As mentioned in the introduction, topic models for text have been under continuous development for the past 20 years. And numerous different types and variations of models have been proposed. This section will present one very popular method, namely the Latent Dirichlet Allocation (LDA). It was proposed by Blei et al. [BNJ03] as a fully generative alternative to the well known pLSI [Hof99]. The term “fully generative” refers to the fact that in contrast to pLSI, the description of LDA allows for generation of new documents.

Before describing the model itself, it is convenient to define the notion of a corpus. In the present work, a corpus $\mathbf{W}$ is a collection of D documents. The order of the documents in the corpus is assumed to be insignificant. Each document d consists of N_d word tokens, where the i^{th} word is denoted $w_{d,i}$. As the “bag-of-words” assumption is used, also the order of the words in each document is neglected. The vocabulary size of the corpus is denoted J.

The LDA model assumes that each document d can be described as a mixture of T topics represented by multinomial distribution parametrised by $\boldsymbol{\theta}_d$. All these individual document-topic distributions are assumed to be independent samples from a Dirichlet distribution parametrised by $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_T]$. Likewise,

each of the T topics is assumed to be representable by a multinomial distribution over J words parametrised by $\boldsymbol{\phi}_t$. These topic-word distribution parameters are assumed to be independent samples from the a Dirichlet distribution with parameters $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_J]$.
Each document d in a corpus is assumed to be generated in the following way. For each word token $w_{d,i}$, a corresponding latent topic variable $z_{d,i}$ is sampled (independently) from the categorical distribution parametrised by $\boldsymbol{\theta}_d$. The sampled topic decides which topic-word distribution to sample the actual word from: $w_{d,i} \sim Cat(\boldsymbol{\phi}_{z_{d,i}})$.
With the probability distributions for the variables defined as described above, LDA can be represented using a probabilistic graphical model as shown in figure 3.1. This representation conveniently shows the conditional dependence relations in the model in a compact way.
When using LDA one has to decide on a value for the number of topics T. This choice will in most cases depend on the corpus analysed and the intentions of the researcher. Also one has to decide on values for the hyper parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. This choice should reflect the assumptions about the data, as smaller values will tend to express the document-topic and the topic-word distributions less smoothly, thus approaching the maximum likelihood solution. In section 3.2.2, a method for optimisation of the hyper parameters is described. The procedure of performing maximum likelihood estimation of hyper parameters in an otherwise Bayesian framework is commonly known as ML-II.

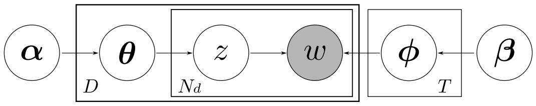


Figure 3.1: Graphical representation of the Latent Dirichlet Allocation model. The model is represented using plates, describing the presence of multiple instances of the variables shown in the plate. The number in the corner of each plate denotes the number of instances of the variables in the plate. The dark nodes represent variables that are observed. $\boldsymbol{\phi}_t \sim Dir(\boldsymbol{\beta})$, $\boldsymbol{\theta}_d \sim Dir(\boldsymbol{\alpha})$, $z_{d,i} \sim Cat(\boldsymbol{\theta}_d)$, and $w_{d,i} \sim Cat(\boldsymbol{\phi}_{z_{d,i}})$

3.1.1 Parameter Inference in LDA

This section will only briefly cover the inference process of LDA, and the reader is referred to section 3.2 where the Author-Topic model is treated in more detail. The process is very similar, therefore only key results will be mentioned here. The goal of applying LDA is often to infer the model parameters $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_t]$ and $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_D]$ that best describe a given corpus. Thus the target for the inference process is the following posterior distribution.

$$p(\boldsymbol{\Phi}, \boldsymbol{\Theta}|\mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{3.1}$$

There is a variety of methods available for estimating (3.1). The original description by Blei et al. [BNJ03] uses Variational Bayes (VB) for making an approximation the desired distribution. Minka and Lafferty [ML02] propose a method based on Expectation Propagation (EP) as a less biased alternative to VB. The experiments in this thesis rely on a third technique called (collapsed) Gibbs sampling. It is widely used in the literature regarding LDA and related models [MWCE07, RZCG+10, GS04]. Gibbs sampling is a Markov chain Monte Carlo algorithm where the chain is designed to converge to a particular joint distribution of interest. It does so by sampling from the conditional distributions of each of the variables in turn, given all the remaining variables. An un-collapsed Gibbs sampler would sample directly from the distribution $p(\boldsymbol{\Phi}, \boldsymbol{\Theta}, \mathbf{z}|\mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, and then sum over the latent variables $\mathbf{z}$. This would be a tedious job because of the amount of variables to sample each iteration of the Gibbs sampler. The trick to reduce the complexity of the sampling process is to integrate out $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ and just sample to approximate

$$\int\int p(\boldsymbol{\Phi}, \boldsymbol{\Theta}, \mathbf{z}|\mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) d\boldsymbol{\Phi} d\boldsymbol{\Theta} = p(\mathbf{z}|\mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{3.2}$$

This is relatively simple because the Dirichlet distribution is conjugate to the categorical/multinomial distribution. Details of the derivation (for AT) are shown in section 3.2.1. The Gibbs sampling algorithm is now used to estimate (3.2) by repeatedly sampling from the conditional

$$p(z_{di} = k|\mathbf{z}_{-d,i}, W_{di} = w, \mathbf{W}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{3.3}$$

This expression can be shown to be proportional to the following very simple fraction

$$\propto \frac{(c_{kw}^{-di} + \beta_w)(v_{kd}^{-di} + \alpha_k)}{(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)} \tag{3.4}$$

where c_{kw}^{-di} is the number of times the word w has been assigned to topic k, excluding the count from the current sample (di). Likewise, v_{kd}^{-di} is the number of word tokens in document d assigned to topic k, again without including the current sample in the count. (3.4) can be normalised to become proper discrete probability distribution by dividing by the sum over k, and then a sample of the topic of the i^{th} word token in the d^{th} document can be drawn. Again the user is referred to section 3.2.1 for details of the derivation, although the presented equations describe Gibbs sampling in the AT model which is very similar to LDA.

After having iterated through the corpus a reasonable number of times (see section 4.1) a sample of the latent topic variables $\mathbf{z}^s$ can be regarded as a sample from the joint distribution of all the latent topic variables $p(\mathbf{z}|\mathbf{W}, \boldsymbol{\alpha}, \boldsymbol{\beta})$. Because the Dirichlet distribution is conjugate to the categorical distribution, and after observing the sample $\mathbf{z}^s$, the posterior distributions of $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_d$ are also Dirichlet distributions with parameter vectors $\mathbf{c}_t + \boldsymbol{\beta}$ and $\mathbf{v}_d + \boldsymbol{\alpha}$ respectively. Thus samples of $\boldsymbol{\phi}_t$ and $\boldsymbol{\theta}_d$ can be obtained using for instance the expected values of the Dirichlet distributions:

$$E(\theta_{td}|\mathbf{z}^s, \mathbf{W}, \boldsymbol{\alpha}) = \frac{v^s_{td} + \alpha_t}{\sum_{k=1}^{T} v^s_{kd} + \alpha_k} \tag{3.5}$$

$$E(\phi_{tw}|\mathbf{z}^s, \mathbf{W}, \boldsymbol{\beta}) = \frac{c^s_{tw} + \beta_w}{\sum_{j=1}^{J} c^s_{tj} + \beta_j} \tag{3.6}$$

where the superscript ^{s} denotes that the quantity is derived from the sample $\mathbf{z}^s$.

3.2 The Author-Topic Model

The Author-Topic model (AT) as described by [RZCG+10] is a modification to LDA, thus all the principles are the same, but are combined to have different meanings and descriptive capabilities. The topic-word distributions as presented for LDA play the same role in AT. Instead of letting each document have a distribution over topics, the AT model describes each author a as a categorical distribution over topics parametrised by $\boldsymbol{\theta}_a$. Thus in LDA and the AT model, documents and authors play similar roles.

The AT model assumes the following document generation process:

Each document has one or more observed (co-)authors, and each word in the document is generated by picking a random author a, uniformly from the set of coauthors, and picking a random topic t from $Cat(\boldsymbol{\theta}_a)$, and a random word w from $Cat(\boldsymbol{\phi}_t)$. Figure 3.2 shows the probabilistic graphical model describing the dependencies in the AT model. Looking at the structure of the model, we see that a corpus where some authors have written multiple documents is equal to a corpus where all documents with identical coauthor sets are concatenated, as the words in these documents will all be generated from the same distributions. This also means that LDA can be regarded as a special case of the AT model, where every document has a single unique author, i.e the author is equal to the document.

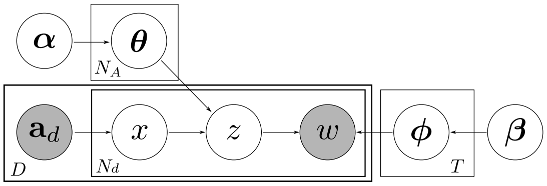


Figure 3.2: Graphical representation of the Author-Topic model. The model is represented using plates, describing the presence of multiple instances of the variables shown in the plate. The number in the corner of each plate denotes the number of instances of the variables in the plate. The dark nodes represent variables that are observed.

3.2.1 Parameter Inference in AT

Just as for LDA, it is possible to use choose between several different inference techniques with the AT model. Just as for LDA, Gibbs sampling has been chosen, and the sampling equations turn out to be very alike.
The method can be characterised as a collapsed, block-Gibbs sampling scheme, as we integrate out $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ and sample both topic z_{di} and author x_{di} at once. The goal is to sample from the conditional distribution of the author- and topic-assignment of the i^{th} word token in the d^{th} document, given all the other tokens and their assignments.

$$p(z_{di}, x_{di}|\mathbf{w}, \mathbf{z}_{-di}, \mathbf{x}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \tag{3.7}$$

Iterating through all the word tokens in a corpus, the gibbs sampling chain is lead towards its stationary distribution: the joint distribution of the author and topic assignments for all words.
We begin with the joint distribution of the random variables in the AT model for a corpus of D documents, which can in accordance to the structure in the model (also see. figure 3.2) be written as

$$\begin{aligned}
&p(\mathbf{w}, \mathbf{x}, \mathbf{z}, \boldsymbol{\Phi}, \boldsymbol{\Theta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \\
&= p(\mathbf{w}|\mathbf{z}, \boldsymbol{\Phi}) p(\mathbf{z}|\mathbf{x}, \boldsymbol{\Theta}) p(\mathbf{x}|A) p(\boldsymbol{\Phi}|\boldsymbol{\beta}) p(\boldsymbol{\Theta}|\boldsymbol{\alpha}) \\
&= \prod_{i=1}^{N} \phi_{z_i w_i} \prod_{i=1}^{N} \theta_{z_i x_i} \prod_{d=1}^{D} \left(\frac{1}{N_{A_d}}\right)^{N_d} \prod_{t=1}^{K} Dir(\boldsymbol{\phi}_t|\boldsymbol{\beta}) \prod_{a=1}^{N_A} Dir(\boldsymbol{\theta}_a|\boldsymbol{\alpha})
\end{aligned}$$

where $\mathbf{A}$ represents the author assignments of the documents (N_A is the total number of authors and N_{A_d} is the number of coauthors of document d), N is the number of word tokens in the corpus, and N_d is the number of words in document d.

Using the definition of the probability density function of the Dirichlet distribution

$$= \left[\prod_{t=1}^{T}\prod_{j=1}^{J}\phi_{tj}^{c_{tj}}\right]\left[\prod_{t=1}^{T}\prod_{a=1}^{N_A}\theta_{ta}^{v_{ta}}\right]\left[\prod_{d=1}^{D}\frac{1}{(N_{A_d})^{N_d}}\right]\left[\prod_{t=1}^{T}C(\boldsymbol{\beta})\prod_{j=1}^{J}\phi_{tj}^{\beta_j-1}\right]\left[\prod_{a=1}^{N_A}C(\boldsymbol{\alpha})\prod_{t=1}^{T}\phi_{ta}^{\alpha_t-1}\right]$$

where c_{tj} denotes the number of times the j^{th} word in the vocabulary is assigned to topic t. Likewise, v_{ta} denotes the number of word tokens written by author a, assigned to topic t.

$$C(\mathbf{q}) = \frac{\Gamma(\sum_{r=1}^{R} q_r)}{\prod_{r=1}^{R}\Gamma(q_r)} \qquad \text{where } \mathbf{q} = [q_1, q_2, \cdots, q_R] \tag{3.8}$$

$$= C(\boldsymbol{\beta})^T C(\boldsymbol{\alpha})^{N_A}\left[\prod_{d=1}^{D}\frac{1}{(N_{A_d})^{N_d}}\right]\left[\prod_{t=1}^{T}\prod_{j=1}^{J}\phi_{tj}^{c_{tj}}\phi_{tj}^{\beta_j-1}\right]\left[\prod_{t=1}^{T}\prod_{a=1}^{N_A}\theta_{ta}^{v_{ta}}\theta_{ta}^{\alpha_t-1}\right]$$

For convenience, a constant is defined as

$$G = C(\boldsymbol{\beta})^T C(\boldsymbol{\alpha})^{N_A}\prod_{d=1}^{D}\frac{1}{(N_{A_d})^{N_d}}$$

$$p(\mathbf{w}, \mathbf{x}, \mathbf{z}, \boldsymbol{\Phi}, \boldsymbol{\Theta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) = G\left[\prod_{t=1}^{T}\prod_{j=1}^{J}\phi_{tj}^{c_{tj}+\beta_j-1}\right]\left[\prod_{t=1}^{T}\prod_{a=1}^{N_A}\theta_{ta}^{v_{ta}+\alpha_t-1}\right]$$

Integrating out $\boldsymbol{\Phi}$ and $\boldsymbol{\Theta}$ we obtain the marginal conditional probability of the words, the author- and the topic-assignments, given the hyper parameters and the authors.

$$p(\mathbf{w}, \mathbf{x}, \mathbf{z}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \tag{3.9}$$

$$= G\iint\left[\prod_{t=1}^{T}\prod_{j=1}^{J}\phi_{tj}^{c_{tj}+\beta_j-1}\right]\left[\prod_{t=1}^{T}\prod_{a=1}^{N_A}\theta_{ta}^{v_{ta}+\alpha_t-1}\right]d\boldsymbol{\Theta}d\boldsymbol{\Phi} \tag{3.10}$$

$$= G\int\prod_{t=1}^{T}\prod_{j=1}^{J}\phi_{tj}^{c_{tj}+\beta_j-1}d\boldsymbol{\Phi}\int\prod_{t=1}^{T}\prod_{a=1}^{N_A}\theta_{ta}^{v_{ta}+\alpha_t-1}d\boldsymbol{\Theta} \tag{3.11}$$

$$= G\prod_{t=1}^{T}\int\prod_{j=1}^{J}\phi_{tj}^{c_{tj}+\beta_j-1}d\boldsymbol{\phi}\prod_{a=1}^{N_A}\int\prod_{t=1}^{T}\theta_{ta}^{v_{ta}+\alpha_t-1}d\boldsymbol{\theta} \tag{3.12}$$

Now the integrals are proportional to integrals over the Dirichlet pdf, and by multiplying by $\left(\frac{C(\mathbf{c}_t+\boldsymbol{\beta})}{C(\mathbf{c}_t+\boldsymbol{\beta})}\right)^T \left(\frac{C(\mathbf{v}_a+\boldsymbol{\alpha})}{C(\mathbf{v}_a+\boldsymbol{\alpha})}\right)^{N_A} = 1$, (3.12) simplifies to

$$= G \prod_{t=1}^{T} \frac{1}{C(\mathbf{c}_t+\boldsymbol{\beta})} \prod_{a=1}^{N_A} \frac{1}{C(\mathbf{v}_a+\boldsymbol{\alpha})} \tag{3.13}$$

$$= \prod_{d=1}^{D} \frac{1}{(N_{A_d})^{N_d}} \prod_{t=1}^{T} \frac{C(\boldsymbol{\beta})}{C(\mathbf{c}_t+\boldsymbol{\beta})} \prod_{a=1}^{N_A} \frac{C(\boldsymbol{\alpha})}{C(\mathbf{v}_a+\boldsymbol{\alpha})} \tag{3.14}$$

Observe that via Bayes' rule we can reformulate the conditional probability of a single token:

$$p(z_{di}=k, x_{di}=u|\mathbf{w}, \mathbf{z}_{-di}, \mathbf{x}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \tag{3.15}$$

$$= \frac{p(z_{di}=k, x_{di}=u, w_{di}=h|\mathbf{z}_{-di}, \mathbf{x}_{-di}, \mathbf{w}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A})}{p(w_{di}=h|\mathbf{z}_{-di}, \mathbf{x}_{-di}, \mathbf{w}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A})} \tag{3.16}$$

where the denominator obviously is a constant with respect to (3.15), thus it can be removed and the equality is changed to a proportionality:

$$\propto p(z_{di}=k, x_{di}=u, w_{di}=h|\mathbf{z}_{-di}, \mathbf{x}_{-di}, \mathbf{w}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \tag{3.17}$$

using Bayes' rule once again, (3.17) can be reformulated to

$$= \frac{p(\mathbf{z}, \mathbf{x}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A})}{p(\mathbf{z}_{-di}, \mathbf{x}_{-di}, \mathbf{w}_{-di}|\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A})} \tag{3.18}$$

which in turn means that the conditional distribution that we seek, (3.15), is proportional to (3.18). The numerator and the denominator are both of the form (3.14) and the denominator only differs from the numerator by excluding the current sample (z_{di}, x_{di}).
We denote the topic-word counts and the author-topic counts without the current sample $\mathbf{c}_t^{-di}$ and $\mathbf{v}_a^{-di}$ respectively. See (3.19) for details.

$$\boxed{\begin{aligned} c_{tj}^{-di} &= \begin{cases} c_{tj}-1 & \text{if } (t=k \wedge j=h) \\ c_{tj} & \text{otherwise} \end{cases} \\ v_{ta}^{-di} &= \begin{cases} v_{ta}-1 & \text{if } (t=k \wedge a=u) \\ v_{ta} & \text{otherwise} \end{cases} \end{aligned}} \tag{3.19}$$

The terms $C(\boldsymbol{\beta})^T$ and $C(\boldsymbol{\alpha})^{N_A}$ conveniently cancel out. Note that also the number of tokens in the current document is decreased by one in the denominator.

$$\frac{p(\mathbf{w},\mathbf{z},\mathbf{x}|\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{A})}{p(\mathbf{z}_{-di},\mathbf{x}_{-di},\mathbf{w}_{-di}|\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{A})} \tag{3.20}$$

$$= \frac{\prod_{d=1}^{D} \frac{1}{(N_{A_d})^{N_d}} \prod_{t=1}^{T} \frac{1}{C(\mathbf{c}_t+\boldsymbol{\beta})} \prod_{a=1}^{N_A} \frac{1}{C(\mathbf{v}_a+\boldsymbol{\alpha})}}{\prod_{d=1}^{D} \frac{1}{(N_{A_d})^{N_d-1}} \prod_{t=1}^{T} \frac{1}{C(\mathbf{c}_t^{-di}+\boldsymbol{\beta})} \prod_{a=1}^{N_A} \frac{1}{C(\mathbf{v}_a^{-di}+\boldsymbol{\alpha})}} \tag{3.21}$$

$$= \frac{1}{N_{A_d}} \prod_{t=1}^{T} \frac{C(\mathbf{c}_t^{-di}+\boldsymbol{\beta})}{C(\mathbf{c}_t+\boldsymbol{\beta})} \prod_{a=1}^{N_A} \frac{C(\mathbf{v}_a^{-di}+\boldsymbol{\alpha})}{C(\mathbf{v}_a+\boldsymbol{\alpha})} \tag{3.22}$$

Using (3.8) we obtain

$$\begin{aligned} &= \frac{1}{N_{A_d}} \prod_{t=1}^{T} \frac{\Gamma(\sum_{j=1}^{J} c_{tj}^{-di}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{tj}+\beta_j)}{\Gamma(\sum_{j=1}^{J} c_{tj}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{tj}^{-di}+\beta_j)} \\ &\qquad \times \prod_{a=1}^{N_A} \frac{\Gamma(\sum_{t=1}^{T} v_{ta}^{-di}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{ta}+\alpha_t)}{\Gamma(\sum_{t=1}^{T} v_{ta}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{ta}^{-di}+\alpha_t)} \end{aligned} \tag{3.23}$$

Keeping in mind that we only need to maintain proportionality to (3.15), the fraction $\frac{1}{N_{A_d}}$ in the above equation can be eliminated. Now the products are split up into parts that do not depend on k and u, and the ones that do.

$$\begin{aligned} &\propto \frac{\Gamma(\sum_{j=1}^{J} c_{kj}^{-di}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{kj}+\beta_j)}{\Gamma(\sum_{j=1}^{J} c_{kj}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{kj}^{-di}+\beta_j)} \prod_{t\neq k} \frac{\Gamma(\sum_{j=1}^{J} c_{tj}^{-di}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{tj}+\beta_j)}{\Gamma(\sum_{j=1}^{J} c_{tj}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{tj}^{-di}+\beta_j)} \\ &\times \frac{\Gamma(\sum_{t=1}^{T} v_{tu}^{-di}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{tu}+\alpha_t)}{\Gamma(\sum_{t=1}^{T} v_{tu}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{tu}^{-di}+\alpha_t)} \prod_{a\neq u} \frac{\Gamma(\sum_{t=1}^{T} v_{ta}^{-di}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{ta}+\alpha_t)}{\Gamma(\sum_{t=1}^{T} v_{ta}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{ta}^{-di}+\alpha_t)} \end{aligned} \tag{3.24}$$

Using (3.19) the two products over $t \neq k$ and $a \neq u$ disappear.

$$\begin{aligned} &= \frac{\Gamma(\sum_{j=1}^{J} c_{kj}^{-di}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{kj}+\beta_j)}{\Gamma(\sum_{j=1}^{J} c_{kj}+\beta_j)\prod_{j=1}^{J}\Gamma(c_{kj}^{-di}+\beta_j)} \\ &\qquad \times \frac{\Gamma(\sum_{t=1}^{T} v_{tu}^{-di}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{tu}+\alpha_t)}{\Gamma(\sum_{t=1}^{T} v_{tu}+\alpha_t)\prod_{t=1}^{T}\Gamma(v_{tu}^{-di}+\alpha_t)} \end{aligned} \tag{3.25}$$

We proceed by splitting the remaining products over t and j and using the definition of the counts (3.19):

$$\begin{aligned}
&= \frac{\Gamma(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)}{\Gamma(1 + \sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)} \frac{\Gamma(c_{kh} + \beta_h)}{\Gamma(c_{kh}^{-di} + \beta_h)} \prod_{j \neq h} \frac{\Gamma(c_{kj} + \beta_j)}{\Gamma(c_{kj}^{-di} + \beta_j)} \\
&\qquad \times \frac{\Gamma(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)}{\Gamma(1 + \sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)} \frac{\Gamma(v_{ku} + \alpha_k)}{\Gamma(v_{ku}^{-di} + \alpha_k)} \prod_{t \neq k} \frac{\Gamma(v_{tu} + \alpha_t)}{\Gamma(v_{tu}^{-di} + \alpha_t)} \qquad (3.26)
\end{aligned}$$

$$\begin{aligned}
&= \frac{\Gamma(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)}{\Gamma(1 + \sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)} \frac{\Gamma(c_{kh}^{-di} + \beta_h + 1)}{\Gamma(c_{kh}^{-di} + \beta_h)} \\
&\qquad \times \frac{\Gamma(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)}{\Gamma(1 + \sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)} \frac{\Gamma(v_{ku}^{-di} + \alpha_k + 1)}{\Gamma(v_{ku}^{-di} + \alpha_k)} \qquad (3.27)
\end{aligned}$$

Using the recurrence relation $\Gamma(z+1) = z\Gamma(z)$ [BM22] the expression can be further simplified:

$$\begin{aligned}
&= \frac{\Gamma(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)}{(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)\Gamma(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)} \frac{(c_{kh}^{-di} + \beta_h)\Gamma(c_{kh}^{-di} + \beta_h)}{\Gamma(c_{kh}^{-di} + \beta_h)} \\
&\quad \times \frac{\Gamma(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)}{(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)\Gamma(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)} \frac{(v_{ku}^{-di} + \alpha_k)\Gamma(v_{ku}^{-di} + \alpha_k)}{\Gamma(v_{ku}^{-di} + \alpha_k)} \qquad (3.28) \\
&= \frac{(c_{kh}^{-di} + \beta_h)(v_{ku}^{-di} + \alpha_k)}{(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)} \qquad (3.29)
\end{aligned}$$

To obtain the probability of a single sample rather than the derived proportional expression (3.29), it has to be normalised resulting in the following probability

$$\begin{aligned}
&p(z_{di} = k, x_{di} = u | \mathbf{w}, \mathbf{z}_{-di}, \mathbf{x}_{-di}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{A}) \\
&\qquad = \frac{\frac{(c_{kh}^{-di} + \beta_h)(v_{ku}^{-di} + \alpha_k)}{(\sum_{j=1}^{J} c_{kj}^{-di} + \beta_j)(\sum_{t=1}^{T} v_{tu}^{-di} + \alpha_t)}}{\sum_{a=1}^{N_{A_d}} \sum_{t=1}^{T} \frac{(c_{tj}^{-di} + \beta_j)(v_{ta}^{-di} + \alpha_t)}{(\sum_{j=1}^{J} c_{tj}^{-di} + \beta_j)(\sum_{t=1}^{T} v_{ta}^{-di} + \alpha_t)}} \qquad (3.30)
\end{aligned}$$

The presented derivations are heavily inspired by various sources [Hei04, Wan08, Car10, RZCG+10, MWCE07].

3.2.2 Maximum Likelihood II: Estimating Hyper Parameters

This section shows an example of how to estimate the hyper parameters using maximum likelihood II. Generally speaking, this is a way to let the data control the parameters of the prior distribution in a bayesian setting. It relies on the assumption that the number of estimated hyper parameters is small compared to the amount of data, so that overfitting is avoided as much as possible. For the LDA, Wallach et al.[WMM09] argue that a configuration with a symmetric Dirichlet prior on the topic-word distributions and an asymmetric Dirichlet distribution as prior for the document-topic distributions provides the best results, and adds an appropriate amount of flexibility to the model, compared to using only symmetric priors. This configuration, denoted AS, provides the possibility for some topics to be more likely than others.

In this section update rules for the Dirichlet hyper parameters in the case of the AT model are derived. With D as the number of documents in the corpus, N_d as the number of word tokens in document d, we start by formulating the model evidence, where $\mathbf{z}$ and $\mathbf{x}$ contain the topic and author assignments for all words in the corpus respectively:

$$p(\mathbf{z}, \mathbf{x}|\boldsymbol{\alpha}, N_A) = \int p(\mathbf{z}, \mathbf{x}, \boldsymbol{\Theta}|\boldsymbol{\alpha}, N_A) d\boldsymbol{\Theta} \tag{3.31}$$

$$= \int p(\mathbf{z}|\mathbf{x}, \boldsymbol{\Theta}, \boldsymbol{\alpha}, N_A) p(\mathbf{x}|\boldsymbol{\Theta}, \boldsymbol{\alpha}, N_A) p(\boldsymbol{\Theta}|\boldsymbol{\alpha}, N_A) d\boldsymbol{\Theta} \tag{3.32}$$

$$= \int \left[\prod_{d=1}^{D}\prod_{i=1}^{N_d}\prod_{t=1}^{T}\theta_{t,x_{di}}^{\delta(z_{di}=t)}\right] \left[\prod_{d=1}^{D}\prod_{i=1}^{N_d}\frac{1}{N_{A_d}}\right] \left[\prod_{a=1}^{N_A} C(\boldsymbol{\alpha}) \prod_{t=1}^{T}\theta_{t,a}^{\alpha_t - 1}\right] d\boldsymbol{\Theta} \tag{3.33}$$

using the definition from (3.8).

Move out the constant terms, and reformulate the products to make use of the author-topic counts v_{ta}.

$$= \left[\prod_{d=1}^{D}\frac{1}{N_{A_d}^{N_d}}\right] C(\boldsymbol{\alpha})^{N_A} \int \prod_{a=1}^{N_A}\prod_{t=1}^{T}\theta_{t,a}^{v_{ta}} \prod_{a=1}^{N_A}\prod_{t=1}^{T}\theta_{t,a}^{\alpha_t - 1} d\boldsymbol{\Theta} \tag{3.34}$$

$$= \left[\prod_{d=1}^{D}\frac{1}{N_{A_d}^{N_d}}\right] C(\boldsymbol{\alpha})^{N_A} \int \prod_{a=1}^{N_A}\prod_{t=1}^{T}\theta_{t,a}^{v_{ta}+\alpha_t - 1} d\boldsymbol{\Theta} \tag{3.35}$$

Now exploit that each $\boldsymbol{\theta}_a$ is drawn independently from a Dirichlet distribution with parameter vector $\boldsymbol{\alpha}$

$$= \left[\prod_{d=1}^{D}\frac{1}{N_{A_d}^{N_d}}\right] C(\boldsymbol{\alpha})^{N_A} \prod_{a=1}^{N_A} \int \prod_{t=1}^{T}\theta_{t,a}^{v_{ta}+\alpha_t - 1} d\boldsymbol{\theta} \tag{3.36}$$

$$= \left[\prod_{d=1}^{D}\frac{1}{N_{A_d}^{N_d}}\right] \prod_{a=1}^{N_A} \frac{C(\boldsymbol{\alpha})}{C(\mathbf{v}_a + \boldsymbol{\alpha})} \int C(\mathbf{v}_a + \boldsymbol{\alpha}) \prod_{t=1}^{T}\theta_{t,a}^{v_{ta}+\alpha_t - 1} d\boldsymbol{\theta} \tag{3.37}$$

$$= \left[\prod_{d=1}^{D}\frac{1}{N_{A_d}^{N_d}}\right] \prod_{a=1}^{N_A} \left(\frac{\Gamma\left(\sum_{t=1}^{T}\alpha_t\right)}{\Gamma\left(\sum_{t=1}^{T}(\alpha_t + v_{ta})\right)} \prod_{t=1}^{T}\frac{\Gamma(\alpha_t + v_{ta})}{\Gamma(\alpha_t)} \right) \tag{3.38}$$

Now we take the logarithm of the model evidence:

$$\log p(\mathbf{z},\mathbf{x}|\boldsymbol{\alpha},N_A) = \sum_{d=1}^{D} -N_d \log N_{A_d}$$
$$+\sum_{a=1}^{N_A}\left(\log\Gamma(\alpha_*) - \log\Gamma(v_{*a}+\alpha_*) + \sum_{t=1}^{T}\log\Gamma(v_{ta}+\alpha_t) - \log\Gamma(\alpha_t)\right) \quad (3.39)$$

Where $\alpha_* = \sum_{t=1}^{T}\alpha_t$ and $v_{*a} = \sum_{t=1}^{T} v_{ta}$ to increase readability.
This quantity can be optimised iteratively by maximising a lower bound, following [Wal08] and [Min00]. The result of this method is often referred to as "Minka's fixed point iteration".
With $n \in \mathcal{N}_+$ and $z, \hat{z} \in \mathcal{R}_+$, the following two inequalities hold

$$\log\Gamma(z) - \log\Gamma(z+n)$$
$$\geq \log\Gamma(\hat{z}) - \log\Gamma(\hat{z}+n) - (\Psi(\hat{z}) - \Psi(\hat{z}+n))(\hat{z}-z) \quad (3.40)$$

$$\log\Gamma(z+n) - \log\Gamma(z)$$
$$\geq \log\Gamma(\hat{z}+n) - \log\Gamma(\hat{z}) + \hat{z}(\Psi(\hat{z}+n) - \Psi(\hat{z}))(\log z - \log\hat{z}) \quad (3.41)$$

where $\Psi(a) = \frac{d}{da}\log\Gamma(a)$ is called the digamma function.
Using 3.40 and 3.41, a lower bound on the (3.39) can be constructed as

$$\log p(\mathbf{z},\mathbf{x}|\boldsymbol{\alpha},N_A) \geq B(\boldsymbol{\alpha}^\star,N_A)$$
$$= \sum_{d=1}^{D} -N_d\log N_{A_d}$$
$$+\sum_{a=1}^{N_A}\Bigg(\log\Gamma(\alpha_*) - \log\Gamma(v_{*a}+\alpha_*) - (\Psi(\alpha_*) - \Psi(v_{*a}+\alpha_*))\,(\alpha_* - \alpha_*^\star)$$
$$+\sum_{t=1}^{T}\Big(\log\Gamma(v_{ta}+\alpha_t) - \log\Gamma(\alpha_t) + \alpha_t(\Psi(v_{ta}+\alpha_t) - \Psi(\alpha_t))(\log\alpha_t^\star - \log\alpha_t)\Big)\Bigg) \quad (3.42)$$

To find the values $\alpha_t^\star$ that maximises the lower bound B, we differentiate it with respect to $\alpha_t^\star$ and set it equal to zero. All the terms not depending on $\alpha_t^\star$ disappear and leave us with just

$$\frac{\partial B(\alpha_t^\star,N_A)}{\partial\alpha_t^\star} = \sum_{a=1}^{N_A}\left(\Psi(\alpha_*) - \Psi(v_{*a}+\alpha_*) + \frac{\alpha_t(\Psi(v_{ta}+\alpha_t) - \Psi(\alpha_t))}{\alpha_t^\star}\right) = 0$$
$$\iff \alpha_t^\star = \alpha_t \frac{\sum_{a=1}^{N_A}(\Psi(v_{ta}+\alpha_t) - \Psi(\alpha_t))}{\sum_{a=1}^{N_A}\Psi(v_{*a}+\alpha_*) - \Psi(\alpha_*)}$$

Following [Wal08], the digamma recurrence relation $\Psi(1+z) - \Psi(z) = \frac{1}{z} \Rightarrow \Psi(n+z) - \Psi(z) = \sum_{f=1}^{n} \frac{1}{f+z-1}$ can be used to simplify the calculations

$$\alpha_t^\star = \alpha_t \frac{\sum_{a=1}^{N_A} \sum_{f=1}^{v_{ta}} \frac{1}{f-1+\alpha_t}}{\sum_{a=1}^{N_A} \sum_{f=1}^{v_{*a}} \frac{1}{f-1+\alpha_*}} \tag{3.43}$$

This update rule is then applied a number of times until convergence is (almost) obtained. In practise for the experiments in this thesis, a fixed number of update iterations are performed every time a certain number of Gibbs sampling iterations have finished. This is mainly due to the simplicity of implementation and because it seems that when the Markov chain converges to its stationary distribution, the hyper parameters do too, making the result more accurate each time we perform an update of the hyper parameters.

The derivations for the update rules for $\boldsymbol{\beta}$ are very similar to the ones presented above, and result in an evidence function of the same form. Thus for an asymmetric prior on the topic-word distributions, the following update rule can be used.

$$\beta_j^\star = \beta_j \frac{\sum_{t=1}^{T} \sum_{f=1}^{c_{tj}} \frac{1}{f-1+\beta_j}}{\sum_{t=1}^{T} \sum_{f=1}^{c_{t*}} \frac{1}{f-1+\beta_*}} \tag{3.44}$$

where $c_{t*} = \sum_{j=1}^{J} c_{tj}$, i.e. the total number of word tokens assigned to topic t. For a symmetric prior, it is convenient to decompose the hyper parameter into a base measure $\mathbf{m}$ describing the mixing proportions and a concentration parameter s controlling the peakiness of the distribution. I.e. $\boldsymbol{\beta} = s\mathbf{m}$, where $\sum_{j=1}^{J} m_j = 1$ and $m_j > 0$. For a symmetric prior we just have a uniform vector with elements $m = \frac{1}{J}$. With this distinction we can now differentiate the lower bound on the log evidence only with respect to the optimal concentration parameter $s^\star$, and the update rules become

$$s^\star = sm \frac{\sum_{t=1}^{T} \sum_{j=1}^{J} \sum_{f=1}^{c_{tj}} \frac{1}{f-1+sm}}{\sum_{t=1}^{T} \sum_{f=1}^{c_{t*}} \frac{1}{f-1+s}} \tag{3.45}$$

3.2.3 Hyper Parameter Optimisation: An Example

To illustrate the influence of the hyper parameter optimisation using the AS configuration described in the previous section, a synthetic corpus is generated from known hyper parameters. Then the model parameters are inferred via Gibbs sampling and the optimised hyper parameters are validated against the true ones.

The synthetic dataset is produced with the settings shown in table 3.1. The true base measure for $\boldsymbol{\alpha}$ is shown, elements sorted in decreasing order, in figure 3.3.

Figure 3.4 shows the result of performing Gibbs sampling and hyper parameter optimisation using the correct number of topics ($T = 20$), while figure 3.5 shows the result of using the same data as for figure 3.4, but with too many topics ($T = 25$). In both cases, the hyper parameters are estimated satisfactorily, and in the latter case even the unused topics are identified. This fits with the results regarding topic stability reported by [WMM09].

N_A	K	J	N_d	D	s_β	s_α
50	20	500	200	600	25	10

Table 3.1: Details of the synthetic data set used for illustration of the hyper parameter optimisation. s_β and s_α are the concentration parameters for the two Dirichlet distributions. As the AS configuration is used we have: $\boldsymbol{\alpha} = s_\alpha \mathbf{m}^\alpha$ with $\sum_{t=1}^{T} m_t^\alpha = 1$ and $\beta_j = s_\beta m_j^\beta$ with $m_j^\beta = \frac{1}{J} \forall j$. The $\mathbf{m}^\alpha$ used for generation of the data is illustrated in figure 3.3

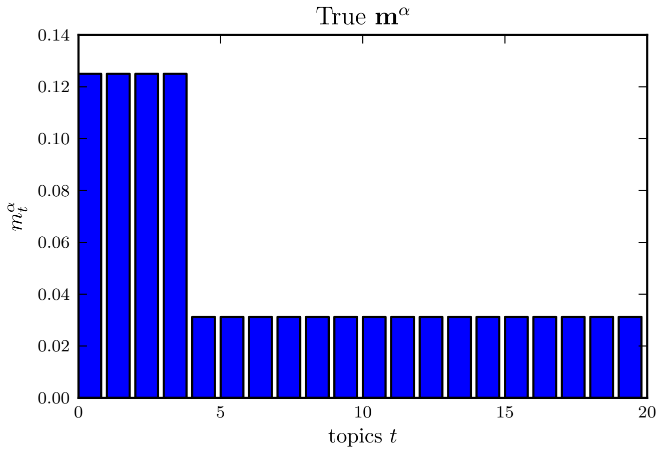
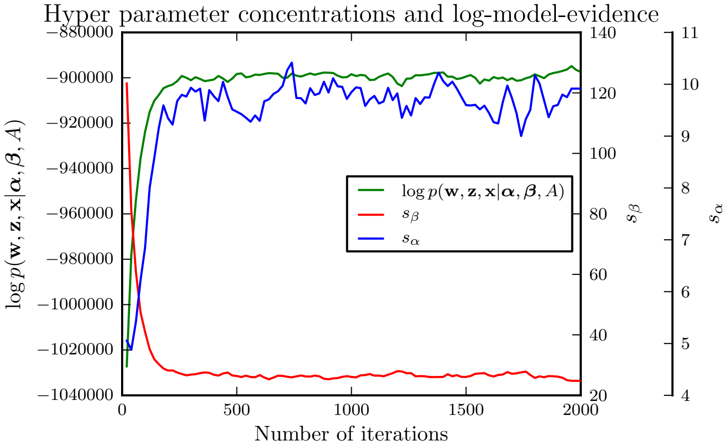
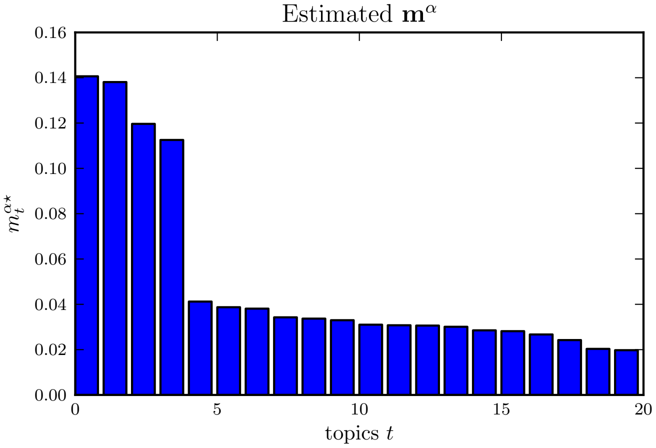


Figure 3.3: True base measure for the author-topic distributions, $\mathbf{m}^\alpha$

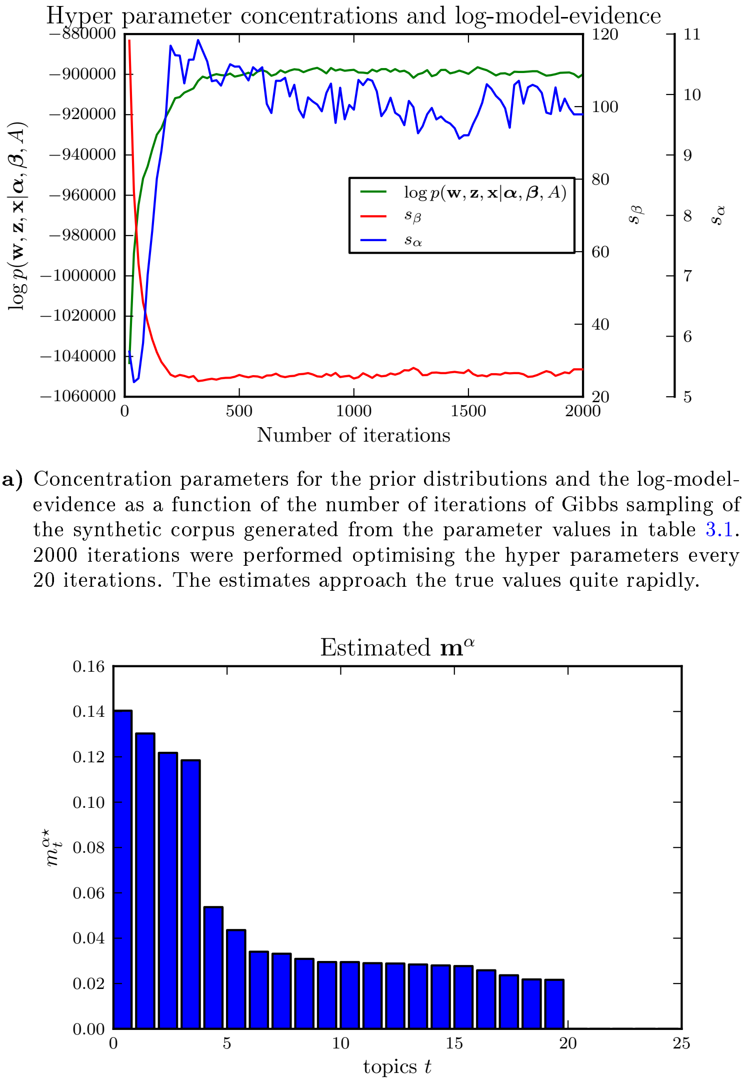


(a) Concentration parameters for the prior distributions and the log-model-evidence as a function of the number of iterations of Gibbs sampling of the synthetic corpus generated from the parameter values in table 3.1. 2000 iterations were performed optimising the hyper parameters every 20 iterations. The estimates approach the true values quite rapidly and only fluctuate slightly.



(b) The estimated optimal base measure $\mathbf{m}^{\alpha\star}$ after 2000 iterations of Gibbs sampling, with the elements sorted in order of decreasing size. We see that the main characteristics are captured satisfactorily, with only small fluctuations.

Figure 3.4: These figures illustrate how the hyper parameter optimisations work when the correct number of topics, $T = 20$, is used for inference.



(a) Concentration parameters for the prior distributions and the log-model-evidence as a function of the number of iterations of Gibbs sampling of the synthetic corpus generated from the parameter values in table 3.1. 2000 iterations were performed optimising the hyper parameters every 20 iterations. The estimates approach the true values quite rapidly.

(b) The estimated optimal base measure $\mathbf{m}^{\alpha\star}$ at the end of the Gibbs sampling chain, with the elements sorted in order of decreasing size. We see that the main characteristics are captured satisfactorily, including the unused topics.

Figure 3.5: These figures illustrate how the hyper parameter optimisations work when the number of topics used for inference is higher ($T = 25$) than the true number of topics that generated the data (Same data as used for figure 3.4).

3.3 Evaluation of Topic Models

To evaluate the topic models, one often look at the likelihood (or perplexity) of a test dataset $\mathbf{W}^{train}$ given the training data $\mathbf{W}^{train}$ and the model parameters. In the AT case, the likelihood for a single test word token $w^{test} = v$ in a document with the coauthors A_d can be formulated as

$$\begin{aligned}&p(w^{test} = v|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train})\\ &= \int_\phi \int_\theta p(w^{test} = v, \mathbf{\Phi}, \mathbf{\Theta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) d\mathbf{\Theta} d\mathbf{\Phi}\\ &= \int_\phi \int_\theta p(w^{test} = v|\mathbf{\Phi}, \mathbf{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) p(\mathbf{\Phi}, \mathbf{\Theta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) d\mathbf{\Theta} d\mathbf{\Phi}\end{aligned}$$

This integral is in most cases intractable to compute exactly, and consequently an approximation is needed. With multiple samples of $\mathbf{\Phi}$ and $\mathbf{\Theta}$, the integral expression can be approximated by the sample mean:

$$p(w^{test} = v|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) \approx \frac{1}{S} \sum_{s=1}^{S} p(w^{test} = v|\mathbf{\Phi}^s, \mathbf{\Theta}^s, \boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) \tag{3.46}$$

This word-likelihood conditioned on a single sample of $\mathbf{\Phi}$ and $\mathbf{\Theta}$ can be expressed as

$$p(w^{test} = v|\mathbf{\Phi}^s, \mathbf{\Theta}^s, \boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) \tag{3.47}$$

$$= \sum_{t=1}^{T} \sum_{a=1}^{N_{A_d}} p(w^{test} = v, z = t, x = a|\mathbf{\Phi}^s, \mathbf{\Theta}^s, \boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) \tag{3.48}$$

$$= \sum_{t=1}^{T} \sum_{a=1}^{N_{A_d}} p(w^{test} = v|z = t, \mathbf{\Phi}^s, \boldsymbol{\beta}, \mathbf{W}^{train}) p(z = t|x = a, \mathbf{\Theta}^s, \boldsymbol{\alpha}, \mathbf{W}^{train}) p(x = a|A_d) \tag{3.49}$$

$$= \sum_{t=1}^{T} \sum_{a=1}^{N_{A_d}} \phi_{tv}^s \theta_{ta}^s \frac{1}{N_{A_d}} \tag{3.50}$$

$$= \frac{1}{N_{A_d}} \sum_{t=1}^{T} \phi_{tv}^s \sum_{a=1}^{N_{A_d}} \theta_{ta}^s \tag{3.51}$$

$$= \frac{1}{N_{A_d}} (\boldsymbol{\phi}_v^s)^\top (\sum_{a=1}^{N_{A_d}} \boldsymbol{\theta}_a^s) \tag{3.52}$$

Thus, combining (3.46) and (3.52) using multiple samples of $\mathbf{\Phi}$ and $\mathbf{\Theta}$ the word likelihood can be approximated by

$$p(w^{test} = v|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) \approx \frac{1}{S}\sum_{s=1}^{S}\frac{1}{N_{A_d}}(\boldsymbol{\phi}_v^s)^\top(\sum_{a=1}^{N_{A_d}}\boldsymbol{\theta}_a^s)$$

Asuncion et al. [AWST09] (although it is in the case of LDA) approximate the likelihood of a full document using multiple samples from the Gibbs sampler by multiplying together the word likelihoods. This approach leads to the following log-likelihood of a document $\mathbf{w}^{test}$

$$\begin{aligned}\ln p(\mathbf{w}^{test}|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train}) &\approx \ln\prod_{i=1}^{N_d}\frac{1}{S}\sum_{s=1}^{S}\frac{1}{N_{A_d}}(\boldsymbol{\phi}_{w_i^{test}}^s)^\top(\sum_{a=1}^{N_{A_d}}\boldsymbol{\theta}_a^s)\\ &= \sum_{i=1}^{N_d}\left(\ln\sum_{s=1}^{S}(\boldsymbol{\phi}_{w_i^{test}}^s)^\top(\sum_{a=1}^{N_{A_d}}\boldsymbol{\theta}_a^s) - \ln(SN_{A_d})\right)\end{aligned} \tag{3.53}$$

This is contrasted by the method used by Rosen-zvi et al. [RZCG$^+$10], where the joined probability of all the words in a document is calculated for each sample from Gibbs sampler:

$$\begin{aligned}&p(\mathbf{w}^{test}|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train})\\ &= \int_\phi\int_\theta\prod_{i=1}^{N_d}\frac{1}{N_{A_d}}\left((\boldsymbol{\phi}_{w_i^{test}})^\top(\sum_{a=1}^{N_{A_d}}\boldsymbol{\theta}_a)\right)p(\mathbf{\Phi}, \mathbf{\Theta}|\boldsymbol{\alpha}, \boldsymbol{\beta}, A_d, \mathbf{W}^{train})d\mathbf{\Theta}d\mathbf{\Phi}\\ &\approx \frac{1}{S}\sum_{s=1}^{S}\left(\prod_{i=1}^{N_d}\frac{1}{N_{A_d}}(\boldsymbol{\phi}_{w_i^{test}}^s)^\top(\sum_{a=1}^{N_{A_d}}\boldsymbol{\theta}_a^s)\right)\end{aligned} \tag{3.54}$$

The question is whether to integrate out $\mathbf{\Phi}$ and $\mathbf{\Theta}$ at word level or at document level. The expression in (3.54) seems to be the most correct when calculating a document likelihood, but one might need to take care to avoid arithmetic underflow. The expression (3.53) is computationally convenient because the logarithm can be used on the terms corresponding to individual words, thus avoiding the risk of arithmetic underflow in the calculations. For the work in this thesis a python library called python library `Decimal` was used for calculation of (3.54). However, it should be noted that the higher precision comes at the cost of increased computational overhead.

Another thing that has to be considered when dealing with the notion of "held-out" or test set likelihood is the fact that the author-topic distributions (document-topic in case of LDA) appear in the expressions. This has consequences for LDA

and the AT model. In the AT case, one needs to have estimates of the topic proportions associated with every single author appearing in the test set. This can be handled by ensuring that all authors in the test set also appear in the training set.
For LDA the problem is a little different, and there seem to be no obvious solution; knowledge of the topic proportions for each test document has to be known in advance, which makes the term "held-out" inadequate. One way to handle the situation is to split all test documents in half, and then infer the topic proportions on one of the halves (keeping the original topic-word distributions) and finally calculate the perplexity on the other half. However, this procedure has the often undesired effect that the borders between training and test set become somewhat muddy.

3.3.1 A Word on Perplexity

Perplexity is a commonly used measure of the quality of language models in general. For topic models, a measure of the predictive performance is often provided as the perplexity of a held-out set of documents (test set) [BNJ03]. The perplexity of a test set $\mathbf{w}^{test}$ consisting of N words is defined as the inverse of the geometric mean value of the likelihoods of the words in the set and is often calculated using the following expression using the logarithm to avoid numerical problems (When using multiple samples of the model parameters, this is only beneficial in the case of (3.53))

$$perp(\mathbf{w}^{test}|\mathcal{M}) = p(\mathbf{w}^{test}|\mathcal{M})^{-\frac{1}{N}} = \exp\left(-\frac{\log p(\mathbf{w}^{test}|\mathcal{M})}{N}\right) \tag{3.55}$$

where $\mathcal{M}$ is shorthand notation for the trained model in question. The perplexity can loosely be interpreted as the mean uncertainty of the model for predicting a word correctly in the test set [Hei04]. This also means that perplexity scores can only be compared within the same corpus because they depend on the size of the vocabulary. Thus when comparing different models using perplexity as a performance measure, the comparison is only valid if exactly the same corpus is used. Furthermore, it is important to remember that perplexity does not say anything about the actual quality and interpretability of the produced document/author and topic distributions, but is merely a convenient statistical measure often used as a guideline in lack of extrinsic evaluation [CBGG⁺09]. By extrinsic evaluation is meant the performance in an actual application in which a topic model is used, such as information retrieval or link prediction.

Another question regarding the measurement of test set perplexity is how to handle "out-of-vocabulary" (OOV) words in the test dataset. If not taken into consideration at inference time, such words will give rise to probabilities of zero resulting in infinite perplexity if included in the perplexity calculation. One way to handle a OOV word is to simply ignore it. This however implicitly means that it is assigned a probability of one, which probably does not reflect the fact very well that the word is so unlikely that it is not even present in the vocabulary. Of course this problem will be reduced by meticulously selecting the analysed data so that it is reasonably representative of the context it is attempted to model. Another way to deal with the problem is to include all words from both the training and the test dataset used into the vocabulary used for training the model. This will cause the probabilities of OOV words to be determined by the hyper parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ because these function as pseudo-counts in the estimates of $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$, see (3.5). If the test data is unknown at the time of the training, another approach could be to add a OOV-substitute word to the vocabulary, and then calculate perplexity of a test set interpreting all OOV words as this artificial word. If one is relying on the hyper parameters to account for the OOV words, it might be beneficial to let their values be guided by an estimate of the amount of OOV words likely to occur in a test set. This has not been treated in the present work, however.

Chapter 4

Experiments and Example Applications of Topic Models

4.1 Gibbs Sampling Analysis

This section describes some simple experiments providing a small scale exploratory analysis of Gibbs sampling applied to Latent Dirichlet Allocation [BNJ03][GS04]. The goal is to investigate the behaviour of the quality of the estimated topic-word and document-topic distribution as well as the speed of convergence while varying the size of the corpus. The experiments are performed using synthetic data generated according to the LDA model as described in section 3.1. All the models in this section are trained on small synthetic data sets and should be regarded as "toy examples". All corpora used for the experiments contain $D = 100$ documents. The number of topics is fixed at $T = 6$ and each of the corresponding multinomial distributions $\boldsymbol{\phi}_t$ over words are fixed as well. The vocabulary size is $J = 100$, and the hyper parameters for the symmetric Dirichlet priors generating the data are set to $\alpha = 0.1$ and $\beta = 0.3$. For training, no hyper parameter optimisation is applied, and the values are set to the generating values. Figure 4.1 illustrates the categorical distributions over the $J = 100$ words for each of the six topics. In the following it will be investigated

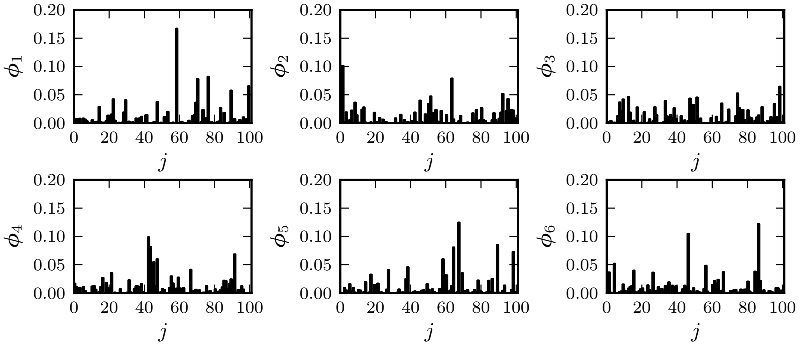


Figure 4.1: Illustration of the parameters for the fixed topic-word probability distributions used for generating all data used in this section.

how the training document lengths (the number of words) affect the quality of the inferred model parameters in terms of both training and test set perplexity. This will be done by generating training corpora all consisting of $D = 100$ documents, but with different document lengths. For convenience, all documents in each corpus has the same number of words N_d.

The perplexity of a set of D documents $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_D\}$ (all of length N_d) can be calculated using only a single sample $\mathbf{\Phi}^s$ and $\mathbf{\Theta}^s$ from a Gibbs sample chain using (4.1) (cf. section 3.3)

$$perp(\mathbf{W}) = \exp\left(-\frac{\sum_{d=1}^{D}\sum_{i=1}^{N_d}\log p(w_{di})}{DN_d}\right) \tag{4.1}$$

To obtain estimates of the perplexity for each value of N_d and the associated uncertainties, multiple pairs of training and test sets are synthesised. For each pair, $D = 100$ document-topic distributions are drawn from $Dir(\theta; \alpha)$ and words for the training and test documents are sampled using these. Note that the same topic proportions are used for both training and test documents. Thus each training document of length N_d has a corresponding test document of length 1000, generated from the exact same distribution over topics as the training document. This is a simple way to ensure that the perplexity of the test documents is well defined without having to estimate $\mathbf{\Theta}$ first. This approach can be criticised for the strong connection between the training and test set which might lead to biased results. However, as also mentioned in section 3.3 this is a general problem for the validity of the evaluation of topic models, and in this particular case the chosen method can be justified because the perplexity values are only used for relative comparisons within the same experiment.

Model parameters are then estimated using each of the training sets and each corresponding test set is used to calculated the perplexity. The results presented later in the section are reported as the mean value and the standard deviation of these perplexities. Note that the only parameters that vary from corpus to corpus are the $D = 100$ document-topic distributions $\boldsymbol{\theta}_d$.

Figure 4.2 shows the perplexity of both the training and the test data as a function of N_d. As mentioned above, each point on the curve represents the mean value of the perplexities calculated using samples from multiple independent Gibbs chains, and the error bars denote one standard deviation σ. For small N_d the training set perplexities are quite low and the corresponding test set perplexities are quite high compared to the values for larger N_d. This shows that the amount of training data is too small to make reasonable estimates of the model parameters; one of the many face of overfitting. As the amount of training data is increased, the perplexities level out. For large N_d the difference is negligible which is to be expected because of the way the training and test documents were generated from the same distributions. The fact that the values of the training set perplexity level out before $N_d = 1000$ suggests that the length of the test documents is more than long enough to provide the amount of data necessary to represent the generating model parameters. One could argue that this is also the reason for the variances of the two perplexities almost being equal. Please note that the perplexities presented in these graphs, are mean values of end-point values of different runs of the Gibbs sampler, therefore, the error bars in the figures are underestimations of the real standard deviation, as they disregard any variance of the obtained samples.

The above deals with the perplexities calculated from samples taken from converged Gibbs samplers. The following will try to illustrate how the perplexities evolve during the process of Gibbs sampling. Figure 4.3 shows the perplexity curves for the testing data for each value of N_d as a function of Gibbs sampling iterations. The perplexities presented in the figures are, like in the above, mean values over a number of repetitions of each configuration. Note that the number of iterations presented in these graphs are iterations through the full training corpus and not individual samples of words. It seems that all it take more or less the same number of iterations to reach the supposedly stationary distribution of the Markov chain. This figure is however a poor indication of the actual computational complexity of the system, as one has to remember that some of the corpora contain far more words to be sampled at each iteration than others. To visualise the differences, figure 4.4 shows the perplexities of the different configurations as a function of individual word samples. The curves show an enormous difference in the efficiency of the different samplers. Looking at the Gibbs sampling algorithm and the equations derived in section 3.2.1, this is not surprising; the influence of each word is inverse proportional to the total number of word tokens in the corpus. Thus the models being trained using large corpora are inherently “heavier”.
Furthermore, many of the “heavy” models reaches approximately the same lower

limit of perplexity (48), but the convergence rates differ significantly. This effect is the same as seen on figure 4.2 where the training and test set perplexities both approximately level out at a common value where $N_d = 500$. An obvious conclusion of the experiment is that one should never include too much redundant data, because it will only result in slower convergence of the Markov chain in the Gibbs sampler, but provide the same level of performance. The usability of this statement is however debatable, as it is very unlikely that one possesses that kind of knowledge before doing the actual model training. One case where it might become useful, is in a situation where time is short. Thus performing more iterations through a subset of the a corpus might result in a better test set perplexity than running only a few iterations with the full corpus. Another possible use of the result is to investigate the performance of a "soft-start" Gibbs sampler, where the a subset of the full corpus is used to make a rough but fast approximation to the stationary distribution of the Markov chain. The distribution could then be refined using more and more of the data. This could potentially speed up Gibbs sampling, but a theoretical validation of such method would be appropriate to ensure the validity of the algorithm.

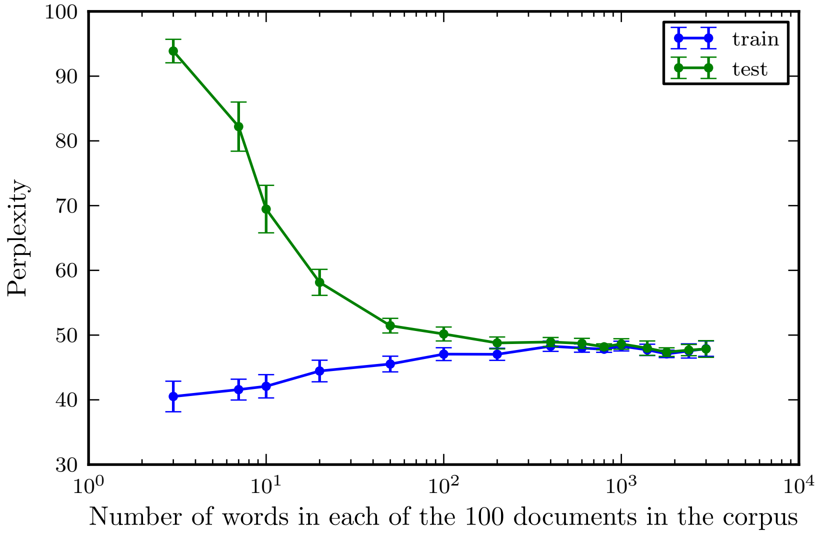


Figure 4.2: Perplexities for the training and test data as functions of N_d. Each of the points on the curves represents the mean value of perplexities calculated from end-samples of multiple Gibbs sampling chains. As expected, the model fits the training data very well for small amounts of training data, but the inferred model parameters are not likely to match the parameters used for generation of the data, hence the high test set perplexity. This is a classical example of overfitting and is caused by the lack of representative data in the training set. As more data is used for parameter estimation both the training and the test set perplexities stabilise at a level of approximately 48. This indicates that the estimated parameters match the model parameters that generated the data quite closely. The error bars on the curves represent $\pm 1\sigma$ of the sample perplexities

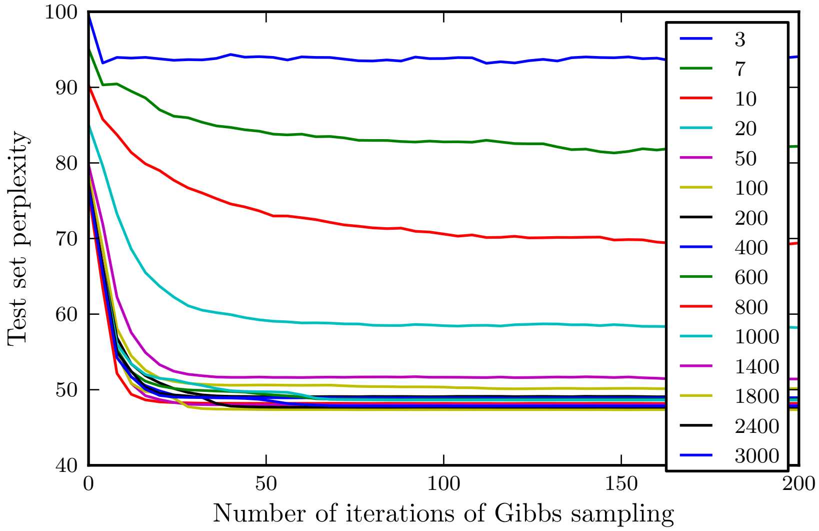


Figure 4.3: Test set perplexities as functions of full iterations through the corpus during Gibbs sampling. The different curves represent different values of N_d. The small corpora are not representative for the distributions that generated then, hence the worse perplexity scores. The curves level out approximately equally fast, but one has to note that the computation time besides the number of iterations also depends on the size of the corpus (See figure 4.4).

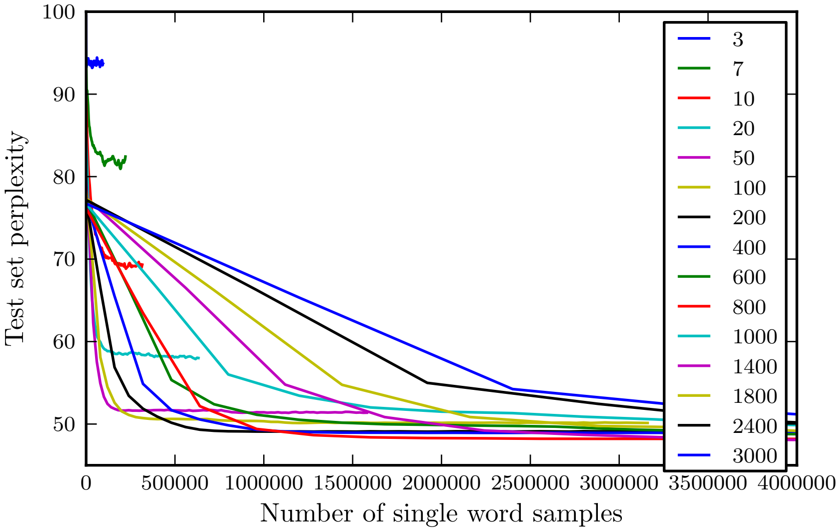


Figure 4.4: Test set perplexities as functions of the number of individual word samples obtained during Gibbs sampling. The different curves represent different values of N_d. Models inferring parameters from smaller corpora tend to have higher test set perplexities. Concurrently they are also much "lighter" which reduces the number of individual word samples before convergence is reached.

4.2 Example Applications of the Author-Topic Model

The author-topic model provides a method to characterise and describe the authors of the documents. In terms of perplexity, AT does not perform as well as LDA [RZCG+10] (at least not on the NIPS dataset). This is probably because the assumption that the authors' distribution over topics is independent of the document (and thus the context in which the document appears), does not hold entirely. Nevertheless, the AT model provides a relatively simple representation of author preferences in terms of subjects/topics, a feature that LDA does not have. Furthermore, as mentioned in section 3.3.1, perplexity is not necessarily the best way to evaluate topic models, and the usefulness of each individual model of course depends on the specific application.

For author-less documents in collections where most of the documents have authors assigned, a possible application of the AT model could be authorship attribution. Similarly, one could imagine teh AT model being used as a tool in investigation of the correctness of a claimed authorship in case of potential forgery. Thus the AT model could for instance be a useful tool, complementing other methods, for analysis of historical documents.
The model provides a simple way to compare authors to each other with regard to similarity in topical interests. In a simple setting, the individual authors' distributions over topics can be directly compared to one another. This might be used as a tool for exploring the use of pseudonyms in a corpus.
The following sections provide two examples of applications of the AT model; The first is outlier detection in document collections. I.e. discovering documents that are unusual for the claimed author. Experiments regarding this use of the AT model are described in depth in section 4.3.
Combined with the knowledge of groups of people frequently coauthoring papers, the information about authors' preferred topics might also be used for discovering sub-groups of authors within a larger group of authors, all acting within the same research field. Such information can for instance help find qualified, unbiased reviewers for scientific papers. Extensive work has been done in this area as the matching task is often a time consuming process, and this thesis will not deal further with the problem. Some examples of approaches to this application of topic models are [ACM07, KZB08, DN92].

The other application of the AT model is treated in section 4.4, and deals with the task of link prediction in the online social network Twitter, by comparing the users' interest via their inferred topic proportions.

4.3 Outlier Detection Using the Author-Topic Model

This section gives illustrative examples of detection of documents that are unusual/unlikely for the stated authors using the Author-Topic model. The applied method is conceptually very simple and is very similar to experiments performed in [RZCG+10].

First, the model is trained using data that is known to be correct, or at least it is known which author attributions are incorrect. This part of the data is referred to as the training set. The training consists of inferring the topic-word and author-topic distributions. After the training stage, it is possible to measure how likely the training documents are in terms of perplexity. Now an upper threshold on the perplexity can be set to split the "good" documents from the "bad". Of course, the optimal value of the threshold depends heavily on both the data and the application.

To measure the performance of the system, a test data set consisting of unseen documents written by authors represented in the training data needs to be defined. Using the threshold from the training data, the test data can then be split and the performance be evaluated. Because the model parameters are inferred using the training data, the measured perplexities of these documents are often lower than what one would expect to get from unseen data. Consequently the threshold will be unrealistically low, producing poor results in the test set. To circumvent this problem, if enough data is available, a third data set is defined. This is in the following called the validation set, and consists of a set documents for which the correctness is known, but has not been used in the training process. This data set is more likely to produce a useful threshold, which can then be used to measure the performance on the test set. This setup is very common within machine learning.

In the following, synthetically generated data is used for illustration of the method, and in section 4.3.2 it is applied to a real world data set consisting of scientific papers from the NIPS conferences.

All experiments with the AT model performed in the following assume fixed symmetric Dirichlet priors on the topic-word and author-topic distributions. Thus the hyper parameters are reduced to scalars $\alpha_t = \alpha \forall t$ and $\beta_j = \beta \forall j$. Perplexities are calculated using multiple independent Gibbs sampling chains with different starting points according to (3.54) [RZCG+10].

4.3.1 Synthetic Outlier Detection Example

This section illustrates the procedure described above using artificial data. The synthetic documents are generated with the model parameters shown in table 4.2, and the sizes of the data sets are summarised in table 4.1. Note that in 10%

of the documents in the test set, the text is generated from another author's distribution over topics.

As proposed in the above, the threshold can conveniently be chosen as some percentile of the perplexity of the validation documents. For simplicity, the most extreme value from the validation set is used for this exemplar analysis (the 100$^{\text{th}}$-percentile). Figure 4.5 shows histograms of the log-likelihoods of the three datasets. The figure shows that some of the test documents are (as expected) indeed very unlikely to have been written by the authors they claim. Examining how the set of unlikely documents matches the set of corrupted documents reveals a recall of 0.850 and a precision of 0.966. This result is an indication that the method works, but it should be noted that this experiment is carried out under artificially advantageous circumstances; the model parameters are inferred using the exact number of topics $T = 5$ and values of hyper parameters $\alpha = 0.1$ and $\beta = 0.01$ with which the data was generated.

The results presented in this section are produced using samples of $\mathbf{\Phi}$ and $\mathbf{\Theta}$ after 2000 iterations, from seven parallel Gibbs sampling chains with different random starting points.

Quantity	training	validation	test
N_A	10		
T	5		
J	500		
D	200	200	1000
N_d	200	200	200
$N_{corrupt}$	0	0	100

Table 4.1: Values of quantities used for generation of the synthetic data in section 4.3.1. Note that 10% of the test documents have been "author-corrupted"

Parameter	Value
α	0.1
β	0.01
T	5

Table 4.2: Model parameters used for generating the synthetic data used in the outlier detection example in section 4.3.1. The same values are used for inference. No hyper parameter optimisation is performed during inference, and both Dirichlet priors are symmetric.

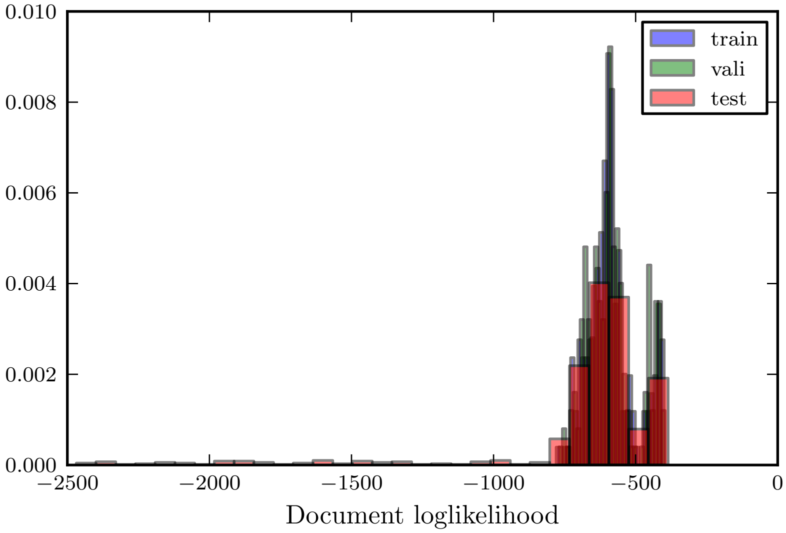


Figure 4.5: Normalized histograms of the log-likelihood of synthetic documents. 10% of the training documents (red) have false authorship information, which is why some of the documents seem very unlikely compared to the training, validation and the remaining majority of the test documents. The log-likelihoods are compared rather than the perplexities for illustrative purposes. As the numbers of tokens in all documents are the same, the presented values can be compared directly.

4.3.2 Identifying Unusual NIPS Papers

This next example makes use of the NIPS data set described in section 2.1, still with the purpose of detecting unlikely documents (outliers). The data was divided into three parts, as described above with the following number of documents in each set: training:1360, validation:190, test:190. The documents were chosen semi-randomly, as all authors represented in the validation or test set also have to appear in the training set, to produce valid results. [RZCG+10] tries to identify unusual papers for a given author, and therefore chooses to measure perplexity for each document as if it were written by only that specific author. The approach taken in this section is a little different in the sense that it uses the full author-list when comparing perplexities amongst documents.
Figure 4.6 shows the distribution of the document perplexities for the three data sets. 95% of the validation documents have a perplexity lower than 5151. This value is used as the threshold for outliers in the test set, and table 4.3 shows the unlikely test documents detected. The two documents in the list are written by David Wolpert. The reason that they are listed as outliers is that another person abbreviated “Wolpert_D”, namely Daniel Wolpert, exists in the data set. David has authored 4 of the 7 papers attributed to Wolpert_D, while Daniel has written the remaining 3. That David ended up on the list is probably due to the particular partitioning of the data set. There seems to be nothing wrong with the entry for “Dietterich_T”, but Thomas Dietterich has coauthored quite different papers, such as “High-performance Job-Shop Scheduling with a Time-delay” and “Locally Adaptive Nearest Neighbor Algorithms” and this might be the reason for his rank in the table. Also, all papers attributed to “Tenorio_M” are written by Manoel Tenorio, so the conclusion of the experiment must be that the method is useful and that irregularities can indeed be discovered. However, it should be noted that in this NIPS data set most authors appear very few times. This sparsity together with the partitioning of the data set into training, validation and test make it hard to infer useful topic proportions for the authors. The lack of more extensive data from the authors could also be the reason for the quite high validation and test perplexities obtained, and experiments in less “author-sparse” data sets would be interesting subject for further analysis in this topic.

Figure 4.7 shows how the perplexity of the training, validation and test set evolves, as the number of iterations of the Gibbs samplers increase. The first data point is recorded at iteration 50, and the validation and test set perplexities do not seem to decrease significantly from this point. Thus the model does not get any better at describing the unseen data. As mentioned already, this might be because the data is not homogeneous enough, i.e. the training set differs too much from the test and validation sets. The author-document assignment matrix is very sparse (see section 2.1), which could give rise to fluctuations

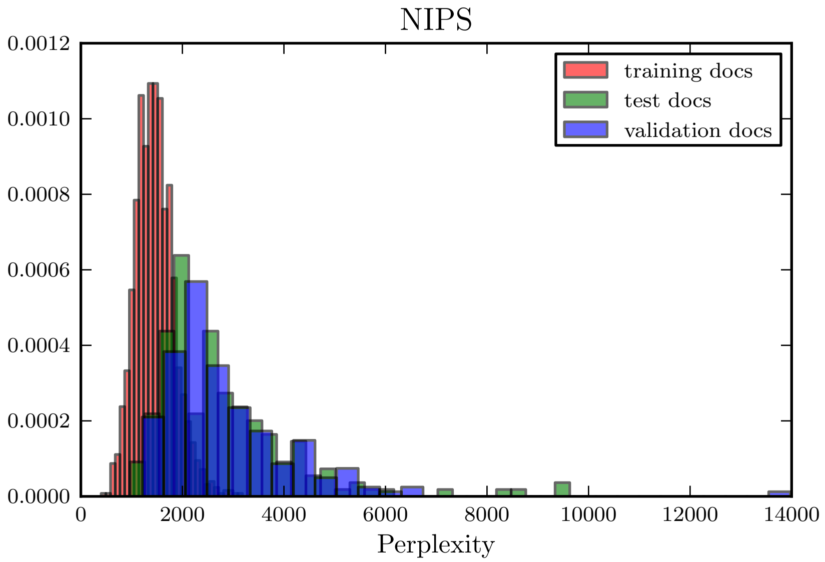


Figure 4.6: Normalized histograms of the perplexities of the training-, validation- and test documents in the NIPS data. Ideally it would be better to have three separate data sets (training validation and test) as descibed in the text.

Perplexity	Title	Postulated authors
9629.48	"Bayesian Backpropagation over I-O Functions Rather Than Weights"	Wolpert_D
9480.17	"On the Use of Evidence in Neural Networks "	Wolpert_D
8736.82	"State Abstraction in MAXQ Hierarchical Reinforcement Learning,"	Dietterich_T
8248.28	"Using Neural Networks to Improve Cochlear Implant Speech Perception"	Tenorio_M
7192.46	"The Computation of Sound Source Elevation in the Barn Owl"	Pearson_J, Spence_C
5511.47	"Learning from Demonstration,"	Schaal_S
5900.76	"Illumination and View Position in 3D Visual Recognition"	Shashua_A
5837.53	"Visual Grammars and their Neural Nets"	Mjolsness_E
5406.32	"A Mathematical Model of Axon Guidance by Diffusible Factors,"	Goodhill_G

Table 4.3: Outliers in the NIPS test set

in the results from different partitionings of the data into training, test and validation sets. One way to deal with the inhomogeneity, to get more level results from run to run, would be to split every document into a number of smaller documents, spreading the information about the authors more equally in the different dataset parts. This approach however, is problematic because is does not reflect reality as well as the full documents, as different parts of the same documents can be found in all three parts of the data set. In some applications this might be just fine, but regarded as invalid in others, like this outlier detection application where it is essential that the documents remain intact.

The results presented in this section were generated using 6 independent Gibbs sampling chains with different random starting points. The perplexities were calculated using samples obtained from the Gibbs samplers after 2000 iterations. The number of topics was set to $T = 100$ and the hyper parameters were fixed at $\alpha = 0.5$ and $\beta = 0.01$.

Ideally, there are no errors in the training data, and the method described above could be applied. Unfortunately, this is not always the case. One way to handle outliers in the training data, is to use a two-step method. First all training data is used for inference in the model. Then the a documents with the highest perplexity scores, where a corresponds to some percentage p of the number of

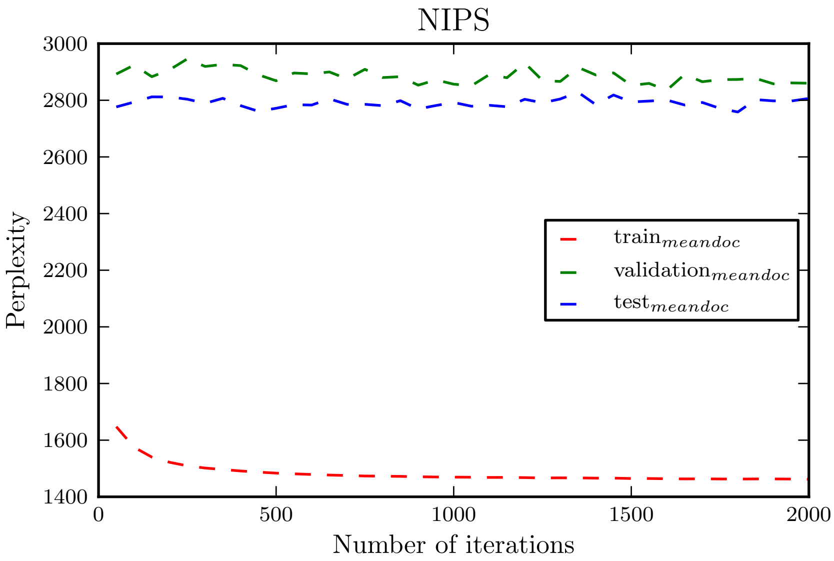


Figure 4.7: Training, validation and test set perplexity as a function of the number of Gibbs sampling iterations. The perplexities presented here are the mean values of the document perplexities each calculated using samples from six independent Markov chains with different random starting points as described by (3.54). That the test and validation set perplexities seem not to decrease at all, stems from the fact that the first recorded point on the curves are recorded after 50 iterations. Thus these perplexities have already settled. The values are however quite high compared to the training set perplexity indicating that all the documents in the combined dataset are very inhomogeneous.

documents, are discarted as outliers. Inference in the model is then performed again using only the accepted part of the data. Using this method, we implicitly assume that we have enough data and that the data is redundant enough to be able to infer the correct distributions after discarding the most unlikely part of the data. If too much is removed, the inferred distributions will probably not model the intended data very well, but if too little is removed, the distributions will be disturbed by noise and hence drop in quality as well. This procedure is heavily inspired by [HSK+00]. As mentioned above, this procedure is applied only as an attempt to minimise the influence of errors in the training set, with regard to author attributions.

To investigate the effect of the described procedure on the NIPS data set, it is split into a training and a test set. The test set consist of 190 documents chosen randomly from the full data set. Note that in the following, the test set is kept untouched for all evaluations. A histogram of the document perplexities of the training set is shown in figure 4.8. From the histogram we observe that there seem to be no obvious outliers in the training set.

When discarding documents from the training set, information about certain authors disappear. It might even happen that authors are eliminated from the training set. This causes potential trouble with the test set, which is kept fixed, if some of the authors featured in the test set are not represented in the training set, because all authors present in the test set must also be represented in the training set to be able to evaluate the inferred model parameters in meaningful way (see section 3.3).

Removing invalidated documents from the test set is not an option, as comparing perplexities across the models trained on the different data is key to the validity of the analysis. Changing the test set, would render the comparison useless. To keep the test set valid, a criterion for a document to be an outlier in the training set is introduced; For a specific value of p, a document is only regarded as an outlier if all of its authors are also represented in the remaining documents. This seems to be the most reasonable approach as we wish to retain the diversity of topics in the data set. This implies that if the only document of an author is very unlikely, it is probably not due to an error in the author attribution, but rather a sign that the inferred word distributions does not describe that single document very well.

Figure 4.9 shows the training and test set perplexities as a function of the amount of data removed from the training set. The training set perplexity decreases a little, as the training set gets smaller. This behaviour is expected because the most unlikely documents are removed from the set. Furthermore, there is a possibility that the vocabulary recognised by the model is reduced when reducing the training set. This leads to incomparable values of perplexity, see section 3.3.1.

If the method works, the ideal shape of the curves would be that the minimum on the test-set curve was located at somewhere above zero, indicating that there could be outliers present in the original training set, and that when these were removed, the inferred model parameters constituted a more accurate description of the test data. The figure does not show this kind of behaviour at all. One of the reasons for this behaviour might be that there are no errors in the original training set. Thus removing documents will only reduce the data basis for the model, probably leading to a less useful model. Another possibility is that there are errors in the test set. As the test documents are chosen randomly from the full data set, there is a possibility that documents with false authorship information is present in the test set, which would only lead to higher test perplexity when excluding other documents with the same “defect” from the training data. These are merely guesses, and further investigations and experiments with other (less sparse) data sets will have to be performed to be able to evaluate the proposed method satisfactorily. Furthermore, a clearer picture of the usability of the method might be given be using an extrinsic performance measure and repeated experiments (possibly with cross validation), rather than a single experiment with usage of perplexity which is merely provides an indication.

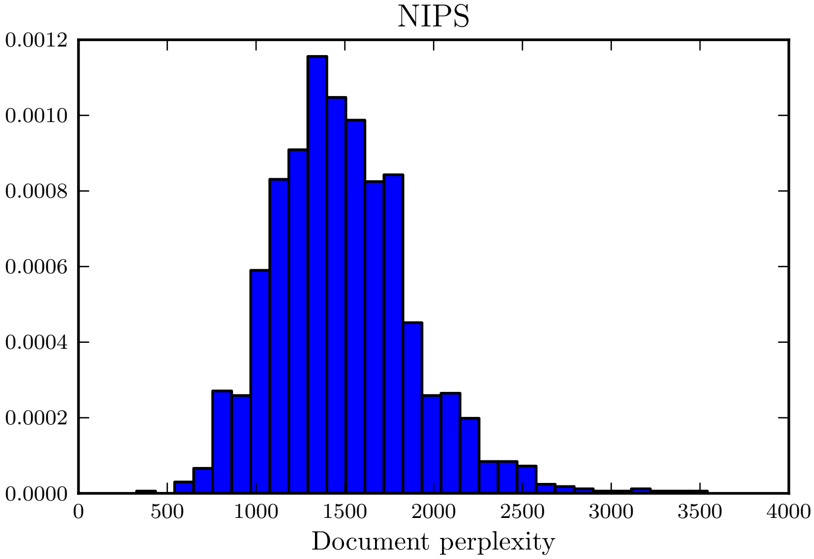


Figure 4.8: Normalised histogram of the perplexities of the documents of the full NIPS training set (used for outlier detection). There is a little probability mass above 3000, but there seem to be no extreme outliers in the training set.

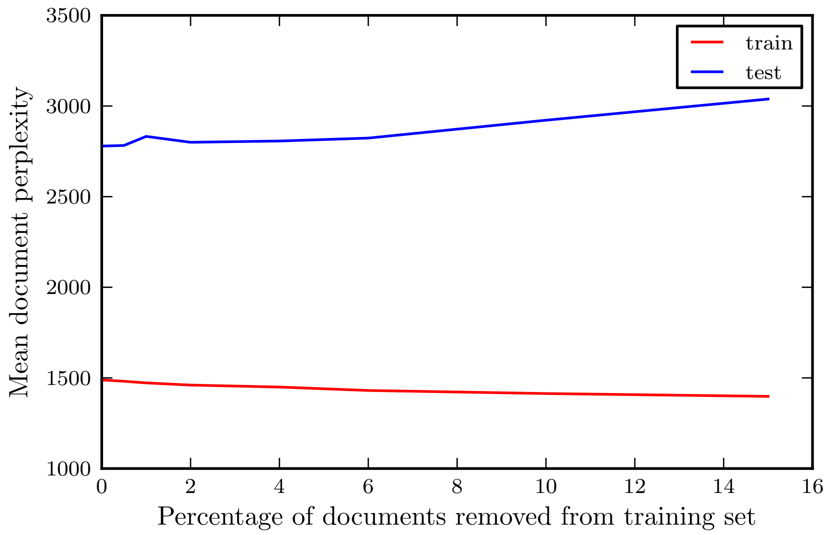


Figure 4.9: This plot shows the mean document perplexities of the training and test NIPS set as a function of amount documents removed from the original training set. There is no sign of improvement in the test set perplexity, and the only noticeable feature of the plot is the classical example of overfitting: increasing test set perplexity as the training set size is reduced.

4.4 Link Prediction in the Twitter Network

Twitter is a large on-line social media platform used by millions of people from all around the world to communicate, share thoughts and have fun.
Some celebreties, companies and organisations use twitter as a communications and marketing tool. They probably have no strong social relation to the majority of the followers. Furthermore, some people might follow a celebrity to go along with the mainstream despite not having a particular interest in the person they are following. These kinds of relationships are likely to be harder to predict from the contents of the posted tweets than relationships of more personal character. On the other hand, if the goal of the link prediction system is to propose new “friends”, the topic model approach might be feasible. To be able to run the inference algorithms within an acceptable time frame only smaller networks with fewer interconnections are analysed. The networks are specifically chosen to fulfil certain criteria discussed in section 2.2. One might imagine that users with relatively few connections are likely to display a more social and personal behaviour, thus having more well defined topical profiles. If this is the case, the

network selection criteria might bias the results in favour of the topic models, but this hypothesis has not been tested in the current work. As a consequence of the few and small networks analysed, the results of this pilot study will not necessarily generalise to larger networks, but will still serve as a useful tool for understanding user interactions. Information about the particular datasets used is summarised in table 2.1 in section 2.2.

Link prediction or graph completion is the task of predicting future or missing links between nodes in a graph. In social media, it is commonly used for recommendation and promotion of new "friends" to the users. A key question to ask is whether the people you are interested in are similar to or very different from you, and in which ways. [WLJH10] contributes to this answer by showing that a topical homophily phenomenon exists in the context of Twitter. This is an important and necessary (but of course not sufficient) condition for being able to predict links successfully, using topic models.

The main idea in this topic model approach to link prediction is to use each user's topic distribution $\boldsymbol{\theta}$ as an indicator of the users taste and interests. The similarity of different users is then assessed by comparison of their respective distributions. See section 4.4.2 for a discussion of similarity measures. As the chosen similarity measures are symmetric, all the inherently directed links in the Twitter graph are interpreted as undirected in this analysis. The estimated similarities are then used as scores for pairs of nodes in the graph, and in sorted order, they represent the ranks of all possible connections in the graph.

[PG11] and [PECX10] uses LDA for link prediction in the same manner as done here. Namely concatenating all tweets written by each user into a "super-tweet" and using it to estimate a document-topic distribution for each user. This work will, however, go further and investigate if presence of user names within the tweets can be used to improve performance of topic models for link prediction. Natively, a twitter message has a single author, but it often contains information related to other people in the network, such as "replys" or "mentions" using the @username notation. This information can be extracted and exploited by expanding the author list of this type of tweets. The author-topic model is ideal for handling this situation, as it is capable of modelling joint authorships. Note that the extracted user names are removed from the tweet text.
This procedure is suggested based on the assumption that if you reply to a message, that message will probably contain some information that is of interest and value to the recipient. For the same reason, original authors of messages that are re-tweeted should also be included in the author-list of the particular message. The question is, however, if this approach will show a significantly better performance than LDA, and an important factor in this question is of course how many @usernames are mentioned in the analysed tweets. See table 4.4.

In the Twitter data, the amount of text written by each user varies immensely.

Name	Proportions of tweets with multiple authors
N1	12.60%
N2	6.96%
N3	11.26%
N4	7.63%
N5	6.10%
N6	10.92%
N7	2.15%
N8	15.87%
N9	5.89%
N10	6.03%

Table 4.4: This table shows the proportions of the tweets in the individual data sets that have more than a single author. This is the result from the extraction of extra authors from the tweets for use with the AT model.

Figure 2.2 in section 2.2 shows the distribution of the number of tweets per user in the full Twitter data set. The first precaution taken, so we do not end up with poor descriptions of some users' topic profiles, is to remove users with less than 100 posted tweets (see section 2.2). At the other end of the scale, a few users have posted several thousands of tweets, leaving them very well-documented. This is another potentially harmful factor for the LDA/AT approach to link prediction, as these few users might impact the inferred word-topic distributions more than users with less data. That being said, if the topics by coincidence fit with the interests of users with few tweets, these might also be described pretty well, but there are no guarantees. This problematic issue is discussed further in section 4.4.1.

To summarise, the main goals for the work presented in this section are to make a small scale comparison of LDA and AT using extra author information extracted from tweets, and to investigate the influence of the amount of available data per user.

4.4.1 Varying the Maximum Number of Tweets per Author

The LDA and AT models model strive to represent the given data (a corpus) in the best possible way, resulting in a low perplexity for the full corpus. Thus all the individual document perplexities are not necessarily low. Very short docu-

ments might have a high perplexity, while long documents have a low one. This behaviour can cause trouble if it is the goal, or at least an important property of a given analysis, that all documents/authors are described equally well.
To explore the effect of this phenomenon, experiments where a maximum number of tweets t_{max} are allowed per user are performed. Users with an abundance of tweets are not excluded, but a subset of tweets of the maximum allowed size is picked from the full set of the respective authors' tweets.
In the case of LDA, the included tweets for each author are selected uniformly at random from the author's set of tweets. In the case of the AT model, only the messages with a single author are considered for removal. This approach only ensures that authors not "collaborating" with others have a hard limit of t_{max} tweets, thus it is possible that some authors exceed the limit if the have "co-authored" more than t_{max} tweets. This procedure is used to emphasise the possible impact of including multi-author information.
In this section, the effect of performing such a thinning of the tweets will be explored and measured using "author perplexity". The "author perplexity" is estimated by collecting all tweets written by each author into a single document and calculating the per-word perplexity of that document.

As an exploratory indicative test of the influence on the inferred model parameters, the "author perplexity" of all the authors in the sub-network N2 (see section 2.2) is monitored as t_{max} is reduced. This test is performed using LDA, i.e. no extra author information has been extracted from the tweets, and thus t_{max} limits the number of tweets for every author. The model parameters have been inferred using four different cutoff values $t_{max} \in \{\infty, 3200, 800, 200\}$. The mean value of the perplexity of all the authors' documents cannot be compared across corpora, as the perplexity measure is dependent on the vocabulary size, which is changed when removing tweets. A corpus with a limited vocabulary will in general show better perplexity than a corpus richer in words. What can be done instead is to compare perplexity for the different authors within a specific corpus. Figures 4.10 and 4.11 show how the author perplexities vary with the number of written tokens (log scale). The lines in the plots are least squares fits of degree one polynomials corresponding to the different values of t_{max} (the different colours). We can observe that the slopes of the lines decrease slightly as the numbers of tokens per author become more equal. This indicates that all authors become more equally described by the model. This experiment is of course very small and non-conclusive, but it contributes to the understanding of the effect of skewness in the amount of available data for the authors/documents. The result does not in itself say anything about the effect on link prediction performance (also see section 3.3.1), but merely suggests that this factor is taken into consideration and studied further.

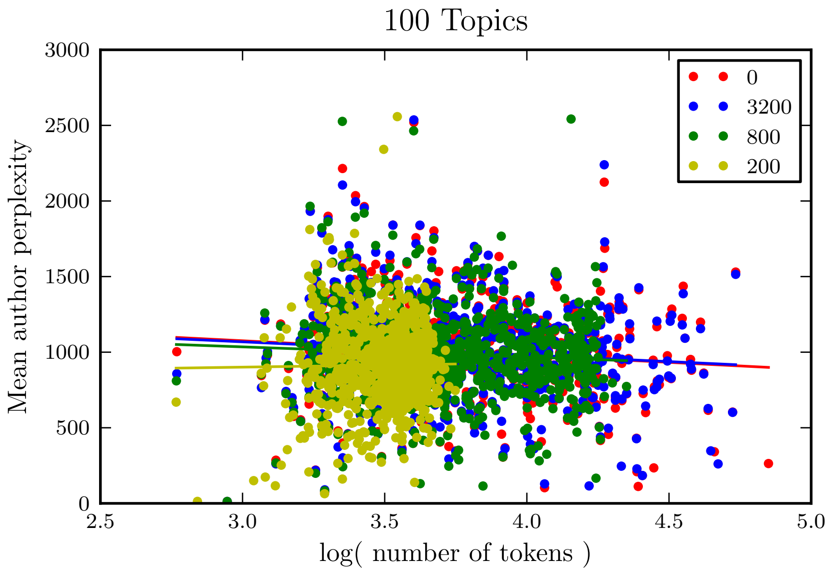


Figure 4.10: Full plot including all authors for all t_{max}. See figure 4.11 for a detailed view and more information.

4.4.2 Similarity Measures

After discovering topics in the data, and estimating the users' topic mixing proportions, the question is how to compare these distributions. Being probability vectors (parametrising multinomial distributions), all elements $e_t \geq 0$ and $\sum_{t=1}^{T} e_t = 1$. All these vectors correspond to points on the $(T-1)$-dimensional simplex defined by the topic-word vectors (parametrising the different topics' distributions over words). There are numerous possible ways to assess the similarity or distance between such two vectors. In this work, only a few are considered. The Euclidean distance is one very easily comprehensible possibility for measuring the distance between two points on the simplex. Also, cosine similarity, effectively measuring the angle between the vectors will be considered. The two last measures, the Manhattan/"Taxi driver" distance (ℓ_1) and the Jensen-Shannon divergence are furthermore investigated. The Jensen-Shannon divergence is also used as a similarity measure in [WLJH10] and [SG05] and is a symmetric measure derived from the KL-divergence; it is the mean of the KL-divergences of two distributions from the mean of the two distributions, and is conveniently bounded to lie in the interval $[0, 1]$. The Jensen-Shannon divergence is not a metric although a real metric can be derived from it if necessary [WLJH10]. This is not a problem in the context of link prediction defined here,

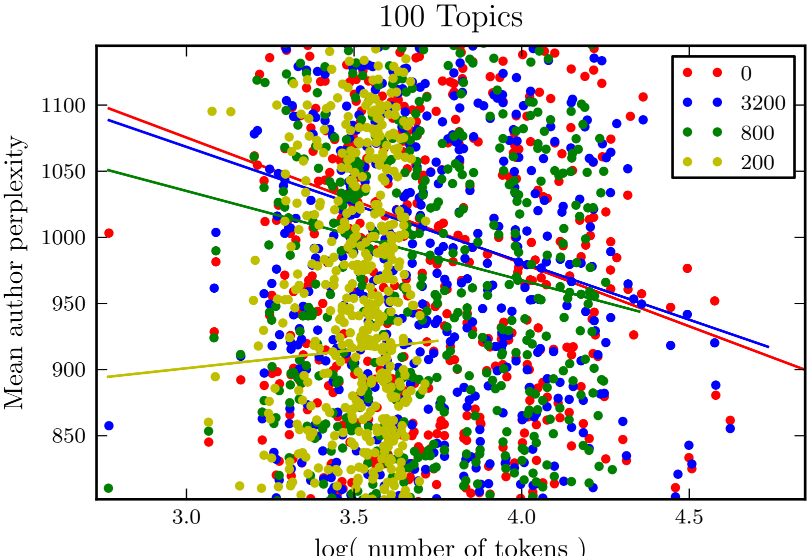


Figure 4.11: Magnified segment from the plot in figure 4.10, emphasising the slope change. Perplexity dependence of the amount of data per author/document. Four subsets, corresponding to the different values of t_{max}, of the data set N2 have been used in this analysis. The numbers in the legend denote t_{max}; the maximum number of tweets allowed per author (0 denotes usage of the unabridged data set). For each colour (data set), each point represents an author, and the corresponding "author perplexity" and logarithm of the number of tokens written by the author. For each data set, a least squares fit of a degree 1 polynomial (note log-transformed number of tokens) is plotted, to indicate the general trend. As suspected, users with a lot of data, are better described by the model than users with less data. This might be because the word-topic distributions have been biased to better fit the productive users. Furthermore, we observe that the mean perplexity across all documents in each data set is reduced when reducing the corpora size. This might be caused by a better description of each author, but, more likely, it is due to the reduced vocabulary resulting from removal of a considerable amount of tweets. The model parameters were inferred using $T = 100$, optimised hyper parameters as described in section 3.2.2 and 5 parallel Gibbs sampling chains were run in parallel (see section 3.3).

because only the ranking of possible connections matter.

4.4.3 Graph Based Methods

As the networks analysed here are quite small compared to the full twitter graph, the results may not be as significant for other sub-networks or the full twitter graph. To see how much information is contained in the graph itself and to be able to relate the results obtained by the topic models to other work on this task, we compare to two commonly used graph-based link prediction methods, the Jaccard coefficient and the Adamic/Adar predictor. Both methods associate a similarity score to each pair of nodes in the graph, based on knowledge of all the other edges, and thus the evaluation can be seen as a leave-one-out framework. Together, the scores form a similarity matrix which can be evaluated just as for the topic models as described in section 4.4.4.
A common setup for the link prediction task is to hide some fraction, f, of the existing connections and score all possible connections in the network using either (4.2) or (4.3), the AUC can be estimated from the resulting ranking of connections. The estimated value of AUC is of course dependent on f, and the choice of this value varies in the literature. In cases where the data contains temporal information and the goal is to predict future links, one can take a more realistic approach by splitting the data into separate time intervals and using the links present in the end of the first, as the "observed" and the new connections formed in the second, as "hidden" [LNK07]. Then the performance becomes dependent on the evolution of the network from one time period to another. Thus f is effectively determined by the chosen time periods.
As no temporal information of the graph structure is available in the examined twitter data set, we settle for $f = 0$; we thus have a fully observed network.
Liben-Nowell and Kleinberg [LNK07] reformulate the measure originally presented by Adamic and Adar [AA03] to fit the problem of link prediction. The score is defined by (4.2) where $\mathit{\Gamma}(x)$ is the set of nodes connected to the node x. As mentioned above, the other graph-based method used for comparison is the widely known Jaccard coefficient given by (4.3).

$$score(x, y) = \sum_{z \in \mathit{\Gamma}(x) \cap \mathit{\Gamma}(x)} \frac{1}{\log |\mathit{\Gamma}(z)|} \tag{4.2}$$

$$score(x, y) = \frac{|\mathit{\Gamma}(x) \cap \mathit{\Gamma}(x)|}{|\mathit{\Gamma}(x) \cup \mathit{\Gamma}(x)|} \tag{4.3}$$

Note that this is not an attempt to compare the performance of the models in order to find and promote the best link prediction method, as the topic models and graph based method are obviously very different and operate on completely different features for the prediction task. Nevertheless, it is interesting how the two different approaches compare on the same data.

4.4.4 Evaluating Link Prediction Performance

Given the ranking of all possible connections in the graph and the knowledge of the true graph i.e. the existing connections, there are several different ways to evaluate the performance. One very simple and commonly used measure is the number of true links amongst the k highest ranking possible connections. This practice fit well with the purpose of link recommendation as it focuses solely on the most probable connections, and incorrectly ranked true connections further down the list are less relevant. This procedure is closely related to the notion of *precision*, often used in IR. The *precision* defined as the proportion of true edges in the set of claimed edges (top-k). The *recall* or *true positive rate* (*TPR*), defined as the proportion of true edges, correctly classified out of the total number of true edges is often combined with the *precision* into a single number; the *F-measure*. These quantities are often used when a classifier has a specific operating point. Using a ranking classifier as in the current case, the operating point can be determined arbitrarily by setting a threshold. To get a more complete picture of the performance, the threshold can be varied and the classifier can be evaluated in the corresponding different operating points. A common way to illustrate the performance is then to plot corresponding values of the *false negative rate* and the *TPR* (stemming from the different thresholds). The resulting curve is called the receiver operating characteristic (ROC) curve and can be used to graphically inspect the performance for different thresholds. It is often desirable to be able to compare performance using a single number, and thus a summary statistic of the performance at all possible operating points can be calculated. For this work, the area under the ROC curve (AUC) has been chosen. A value of 0.5 corresponds to a random ranking of the samples, thus the closer to a value of 1 the better. This measure is used extensively in the literature [KL11, LLC10, CMN08].
The AUC is equivalent to the Wilcoxon-Mann-Whitney statistic, and can be interpreted as the probability of correctly ranking a random pair of samples (possible edges) consisting of a positive and a negative sample [HM82]. The value can be calculated using (4.4) also taking rank ties into consideration [AGH+05].

$$AUC(f, \mathbf{X}, \mathbf{Y}) = \frac{1}{n_n n_p} \sum_{i=1}^{n_p} \sum_{j=1}^{n_n} I(f(x_i) > f(y_j)) + \frac{1}{2} I(f(x_i) = f(y_j)) \tag{4.4}$$

where f is the ranking function, and $\mathbf{X}$ and $\mathbf{Y}$ are the n_p positive and n_n negative examples, respectively. I is an indicator function evaluating 1 if the

argument-condition is true and 0 otherwise.
To be able to compare model performances one also has to assess the uncertainty of estimated AUC value. [HM82] provides the formula (4.5) to calculate the standard error of the estimated AUC.

$$SE(AUC) = \sqrt{\frac{AUC(1-AUC) + (n_p - 1)(Q_1 - AUC^2) + (n_n - 1)(Q_2 - AUC^2)}{n_p n_n}} \tag{4.5}$$

where n_p and n_n are the numbers of existing and non-existing edges, respectively. Q_1 and Q_2 are quantities that depend on the distributions of the positive and negative examples, and [HM82] argues that the approximation (4.6) provides conservative estimates of $SE(AUC)$. These expressions are used when calculating confidence intervals for the estimated AUC values in section 4.4.5.

$$Q_1 = \frac{AUC}{2 - AUC} \qquad Q_2 = \frac{2AUC^2}{1 + AUC} \tag{4.6}$$

4.4.5 Experiment Setup and Results

All results in this section stem from topic model parameters inferred with $T = 100$ topics. The similarity scores are calculated as mean values of the similarities stemming from six parallel Gibbs sampling chains, each with random initialisation.
Instead of removing stop-words which is a popular heuristic method that often works well in practise, the hyper-parameters of the prior distributions are optimised by finding a maximum likelihood estimate. Following the arguments of [WMM09] a symmetric Dirichlet distribution is used as a prior for the word-topic distributions, while the author-topic distributions have an asymmetric Dirichlet as the prior. See section 3.2.2. The value $T = 100$ is in the high end of what is often seen in the literature. This choice was made to avoid having far too few topics to describe the diversity in the corpora. Increasing the number of topics too much would most likely lead to overfitting, and the link-prediction performance would suffer. However, [WMM09] shows that using the asymmetric prior on the author-topic distributions generates much more stable topics, than using an asymmetric prior. Therefore, in the ideal case for a given data set and using a large T, the model is able to adjust and only use the necessary topics. This effect is illustrated in section 3.2.3 using synthetic data. The tables 4.5a, 4.5b and 4.5c are created using LDA with the three different values of $t_{max} \in \{\infty, 1000, 300\}$ (maximum number of tweets per author). The tables show the estimated values of the area under the ROC curve and the corresponding 95% confidence intervals for the four different similarity measures: cosine, euclidean, manhattan/taxi-driver and the Jensen-Shannon divergence. See section 4.4.2. The confidence intervals rely on the assumption that the AUC is

normally distributed, which is reasonable due to the high number of samples (links) in all the cases studied here [HM82].
Likewise, tables 4.6a and 4.6b show the results obtained from using the AT model with $t_{max} \in \{\infty, 1000\}$.
Table 4.7 shows the results obtained using the graph-based link prediction methods discussed in section 4.4.3. For a discussion of the results presented here, see section 4.4.6.

4.4.6 Discussion and Further Work

Looking at just the results from the graph based methods, Adamic/Adar significantly outperforms the Jaccard coefficient in the link prediction problem in all the data sets used here. This is consistent with the results reported from other experiments using Twitter data [LNK07]. These methods operate using features of the local graph structure, and from the results obtained here, it is evident that the graph around a user contains a lot of information about who each user is likely to connect to. Both graph methods significantly outperform the topic models, which is not that surprising since they are fed with information that is more closely related to the prediction task than the topic models are. Therefore, they compete on completely different terms and a direct comparison seems redundant. It is much more interesting to look at the two methods' characteristics, forces and weaknesses and investigate how they might be combined to supplement each other. An analysis of the type of mis-classifications/mis-predictions made by the different methods might reveal a strong independence, and in that case a combination of the topic models could provide an even stronger classifier. This analysis has not been treated here and remains a possible future work.

Both LDA and the AT model perform significantly better than random prediction. This shows that the tweets do indeed contain valuable information about the Twitter users' preferences and interests, and who they are likely to follow. In the tables in section 4.4.5, there seem to be a strong tendency that the Jensen-Shannon divergence produces the most favourable ranking of the possible connections, but studying the uncertainty of the estimated AUCs reveals that the differences are far from always statistically significant at the specified level (95%). This result is a good indication that the similarity measure is not the important factor in the fairly good performance of the topic models. This emphasises the conclusion from [WLJH10] that the notion of topical homophily exists in the Twitter context.

The general trend in the results is that the differences between LDA and the AT model are unremarkable. In most cases the there is no significant difference. Only in a few cases (for example N9-10, $t_{max} = 300$) LDA seems to perform slightly better than the AT model. This indicates that the extra author information does not result in a better prediction of links in the graph. A point of critique of the analysis is that the amount of tweets with multiple authors varies a lot between the data sets and is generally quite low. This might hide a potential difference between the method, i.e. it is possible that the data differences between LDA and AT are simply too small to produce different results. The conclusion from the experiments must be that the AT model and LDA produce very similar results when used with Twitter data. Another possible source of the indifference and even slightly worse performance of AT might in a few cases also stem from the way the model handles the multiple authors. As described in section 3.2, each word token gets assigned to only a single author from the set of coauthors. This means that it is possible that the AT model actually obstructs the whole purpose of the experiment slightly: calculating meaningful similarities between authors, because each tweet is better described by authors with different topical preferences than if they were very similar. This promotes differences rather than similarities in a multi-author situation, which possibly leads to an underestimation of the diversity of the individual authors.

In the current work, the user names extracted from the tweets have only been used for expanding the author-lists of the tweets. This means that the extra information has only an indirect influence on the link prediction, and one might argue that these observed user names could be used much more directly and efficiently because they provide very reliable information about the local graph structure. Work in this direction has not been the primary interest of the approach to link prediction taken in this chapter. The main focus has been on the use of pure topic models to provide an analysis of influential parameters, which will hopefully be useful for further research.

One of the other interesting results obtained here is that the number of tweets written by each individual user does not seem to affect the results remarkably. There is no clear tendency in the results that suggests that performance should depend on t_{max}. A possible explanation could be that Twitter users in general tend to post messages within very few topics and thus are easy to characterise even from quite few observed tweets. This is however an untested hypothesis. Another factor that could influence the results is the number of topics T chosen for the system. Using a different value of T might change the picture, as this will change the premises for the description of the topical user profiles. This is an important question to investigate, but a proper analysis has not been conducted in the current work.

The results are in general very consistent both from method to method and dataset to dataset. This indicates that the analysed networks are neither undersized nor too randomly composed, and this increases the credibility of the analysis. Still we have to remember that the networks were selected using certain criteria, and thus the results only really say something about the particular kind of networks that have been extracted and analysed here; they are quite small and may not be representative for the full Twitter network.

To summarise; topic models can indeed be used for predicting links in the Twitter graph using only the tweets posted by the users, but they are not as precise as methods taking the surrounding graph structure into account. This makes topic models a very interesting subject for further work in network analysis, where no parts of the graph can be explicitly observed, and only the material emitted from the nodes is available. Also the possibility to include sentiment features in the analysis seems to form an interesting research topic. To mention an example use case, such a model might be able to infer the strengths and valences of the interconnections between national politicians to illustrate the variations within the defined political parties. The textual data is often publicly available through sources such as Twitter, published parliament transcripts, websites and newspaper features. This could even be combined with time information to see how similarities change at elections etc.

Another interesting subject for investigation, which is closer to the work performed here, would be to model each Twitter user with two topical profiles, one defined by the tweets posted by the user and another defined by all tweets posted by the people that the user follows. In this case it would be possible to test the hypothesis that people themselves tweet about different (and maybe more narrow) subjects than what they like to read from others. Furthermore, such an approach permits prediction in the directed graph of Twitter as opposed to the simplified undirected graph used in the present work.

Name	cos	euc	taxi	jen-sha
N1	0.6686 ± 0.00432	0.6495 ± 0.00435	0.6788 ± 0.00431	**0.6928** ± 0.00428
N2	0.6763 ± 0.00569	0.6627 ± 0.00572	0.6924 ± 0.00565	**0.7032** ± 0.00561
N3	0.7968 ± 0.00475	0.7812 ± 0.00485	0.7918 ± 0.00478	**0.8016** ± 0.00471
N4	0.6612 ± 0.00376	0.6417 ± 0.00377	0.6662 ± 0.00375	**0.6776** ± 0.00374
N5	0.622 ± 0.00843	0.6188 ± 0.00844	0.6349 ± 0.00842	**0.6485** ± 0.00841
N6	0.7262 ± 0.00324	0.7089 ± 0.00328	0.7369 ± 0.00322	**0.7462** ± 0.00319
N7	0.6701 ± 0.0131	0.6663 ± 0.0132	0.6765 ± 0.0131	**0.6813** ± 0.0131
N8	0.6467 ± 0.00393	0.6239 ± 0.00394	0.6621 ± 0.00391	**0.6715** ± 0.0039
N9	0.6235 ± 0.00198	0.6181 ± 0.00198	0.6333 ± 0.00198	**0.6413** ± 0.00198
N10	0.5931 ± 0.00382	0.5995 ± 0.00382	0.61 ± 0.00382	**0.6137** ± 0.00382

(a) All tweets posted by the users in the network have been used for estimating the users topic proportions.

Name	cos	euc	taxi	jen-sha
N1	0.664 ± 0.00433	0.6469 ± 0.00435	0.677 ± 0.00431	**0.6913** ± 0.00428
N2	0.6731 ± 0.0057	0.6599 ± 0.00572	0.6903 ± 0.00565	**0.7019** ± 0.00562
N3	0.7962 ± 0.00475	0.7795 ± 0.00486	0.7917 ± 0.00478	**0.8012** ± 0.00472
N4	0.6597 ± 0.00376	0.6382 ± 0.00378	0.6684 ± 0.00375	**0.6801** ± 0.00373
N5	0.6199 ± 0.00843	0.6168 ± 0.00844	0.6373 ± 0.00842	**0.6495** ± 0.0084
N6	0.7275 ± 0.00324	0.7109 ± 0.00328	0.7365 ± 0.00322	**0.7458** ± 0.00319
N7	0.6672 ± 0.0132	0.665 ± 0.0132	0.68 ± 0.0131	**0.6856** ± 0.0131
N8	0.6475 ± 0.00393	0.6254 ± 0.00394	0.6613 ± 0.00391	**0.6711** ± 0.0039
N9	0.6376 ± 0.00198	0.6257 ± 0.00198	0.6422 ± 0.00198	**0.6486** ± 0.00197
N10	0.5983 ± 0.00382	0.6039 ± 0.00382	0.6164 ± 0.00382	**0.622** ± 0.00382

(b) At most $t_{max} = 1000$ tweets have been included per user.

Name	cos	euc	taxi	jen-sha
N1	0.6664 ± 0.00433	0.6498 ± 0.00435	0.6781 ± 0.00431	**0.6922** ± 0.00428
N2	0.6712 ± 0.0057	0.6623 ± 0.00572	0.6874 ± 0.00566	**0.7005** ± 0.00562
N3	0.7957 ± 0.00476	0.7821 ± 0.00485	0.7886 ± 0.00481	**0.7998** ± 0.00473
N4	0.6668 ± 0.00375	0.6496 ± 0.00377	0.6744 ± 0.00374	**0.6853** ± 0.00372
N5	0.6151 ± 0.00844	0.615 ± 0.00844	0.6297 ± 0.00843	**0.6438** ± 0.00841
N6	0.723 ± 0.00325	0.7108 ± 0.00328	0.7339 ± 0.00322	**0.7445** ± 0.00319
N7	0.6724 ± 0.0131	0.6575 ± 0.0132	0.6852 ± 0.0131	**0.6912** ± 0.013
N8	0.6402 ± 0.00393	0.6295 ± 0.00394	0.6588 ± 0.00392	**0.6695** ± 0.0039
N9	0.6329 ± 0.00198	0.6241 ± 0.00198	0.6409 ± 0.00198	**0.6488** ± 0.00197
N10	0.6096 ± 0.00382	0.6106 ± 0.00382	0.6243 ± 0.00382	**0.6293** ± 0.00382

(c) At most $t_{max} = 300$ tweets have been included per user.

Table 4.5: AUC of sub-networks using LDA with optimised hyper-parameters. The stated intervals are 95% confidence intervals.

Name	cos	euc	taxi	jen-sha
N1	0.6654 ± 0.00433	0.6419 ± 0.00436	0.6729 ± 0.00432	**0.6862** ± 0.00429
N2	0.6719 ± 0.0057	0.6581 ± 0.00572	0.6886 ± 0.00566	**0.6992** ± 0.00563
N3	0.7816 ± 0.00485	0.7671 ± 0.00494	0.7806 ± 0.00486	**0.7904** ± 0.00479
N4	0.6586 ± 0.00376	0.6366 ± 0.00378	0.6646 ± 0.00375	**0.6762** ± 0.00374
N5	0.6216 ± 0.00843	0.6153 ± 0.00844	0.6342 ± 0.00843	**0.6465** ± 0.00841
N6	0.72 ± 0.00326	0.7053 ± 0.00329	0.7325 ± 0.00323	**0.7427** ± 0.0032
N7	0.6691 ± 0.0131	0.6658 ± 0.0132	0.6751 ± 0.0131	**0.6798** ± 0.0131
N8	0.6409 ± 0.00393	0.6174 ± 0.00394	0.6572 ± 0.00392	**0.6672** ± 0.00391
N9	0.6301 ± 0.00198	0.6247 ± 0.00198	0.6377 ± 0.00198	**0.6438** ± 0.00198
N10	0.5874 ± 0.00382	0.5967 ± 0.00382	0.6071 ± 0.00382	**0.6136** ± 0.00382

(a) All tweets posted by the users in the network have been used for estimating the users topic proportions.

Name	cos	euc	taxi	jen-sha
N1	0.6612 ± 0.00433	0.6413 ± 0.00436	0.6716 ± 0.00432	**0.6854** ± 0.00429
N2	0.6699 ± 0.0057	0.6559 ± 0.00573	0.688 ± 0.00566	**0.6991** ± 0.00563
N3	0.7844 ± 0.00483	0.7643 ± 0.00496	0.7796 ± 0.00486	**0.7895** ± 0.0048
N4	0.66 ± 0.00376	0.6346 ± 0.00378	0.6659 ± 0.00375	**0.6768** ± 0.00374
N5	0.6248 ± 0.00843	0.6156 ± 0.00844	0.6378 ± 0.00842	**0.6517** ± 0.0084
N6	0.7268 ± 0.00324	0.7084 ± 0.00328	0.7345 ± 0.00322	**0.7442** ± 0.0032
N7	0.6693 ± 0.0131	0.6575 ± 0.0132	0.68 ± 0.0131	**0.6858** ± 0.0131
N8	0.6432 ± 0.00393	0.6172 ± 0.00394	0.6587 ± 0.00392	**0.6678** ± 0.00391
N9	0.6297 ± 0.00198	0.6182 ± 0.00198	0.6355 ± 0.00198	**0.6445** ± 0.00198
N10	0.5952 ± 0.00382	0.5984 ± 0.00382	0.6098 ± 0.00382	**0.6165** ± 0.00382

(b) At most $t_{max} = 1000$ tweets have been included per user.

Table 4.6: AUC of sub-networks using the AT model with optimised hyper-parameters. The stated intervals are 95% confidence intervals.

Name	Jaccard	Adamic/Adar
N1	0.8796 ± 0.003198	**0.9156** ± 0.002756
N2	0.8798 ± 0.004235	**0.9168** ± 0.003632
N3	0.9156 ± 0.003392	**0.9357** ± 0.003007
N4	0.8921 ± 0.002665	**0.9263** ± 0.002265
N5	0.862 ± 0.006556	**0.9195** ± 0.005255
N6	0.9008 ± 0.002298	**0.9306** ± 0.001969
N7	0.8793 ± 0.009813	**0.921** ± 0.00822
N8	0.8468 ± 0.003177	**0.8949** ± 0.002742
N9	0.8923 ± 0.001397	**0.9332** ± 0.001138
N10	0.9039 ± 0.002559	**0.931** ± 0.002216

Table 4.7: AUC of sub-networks using the methods of the Jaccard coefficient and Adamic/Adar. The entries in the table are 95% confidence intervals for the estimated AUC.

Chapter 5

Thesis Conclusion

This master thesis has treated a variety of applications of the Latent Dirichlet Allocation model and one of its derivatives, the Author-Topic model. Model parameter inference has been performed by collapsed Gibbs sampling for which the sampling equations have been derived.

Some insight to the functionalism and behaviour of the collapsed Gibbs sampler for LDA has been gained through experiments with synthetic corpora of varying sizes. The main results indicate that it might be computationally beneficial to start the Gibbs sampling on a subset of the data rather than the full dataset to obtain faster initial convergence of the Markov Chain. However, further analysis is needed to make a conclusion.

Furthermore, a method for hyper parameter optimisation using maximum likelihood has been applied to the AT model. The optimisation deals with a configuration of the topic model where a symmetrical Dirichlet distribution is used as prior for each topic's distribution over words, and each author's/document's mixing proportions of topics is provided with an asymmetric Dirichlet prior.

The thesis has presented setups and results of experiments with specific use cases of topic models such as document outlier detection and social network link prediction. The experiments were conducted using both real and synthetic data.

The Author-Topic model proved to be able to detect documents in the NIPS dataset containing incorrect authorship information. The AT model parameters were inferred using a set of training documents, and by means of perplexity each document in a separate test dataset was classified as either normal or abnormal. Also a method for removing influence of possible errors in the training set was investigated but showed no sign of improved performance measured on perplexity. However, a more thorough study using an extrinsic evaluation of performance, and possibly cross validation, would be appropriate.

Last, a pilot study of the use of topic models in the link prediction problem in the Twitter network was carried out, and performances of LDA and the AT model were compared to the two well known graph-based approaches: the Jaccard coefficient and the method of Adamic/Adar. On the Twitter subgraphs used in the study, LDA and AT showed very similar performances but they were not nearly as accurate as the graph based methods. Thus the pure topic model approach to prediction of links seems to have its limits in a context where the notion of a link exists explicitly. Nevertheless, with Areas Under the ROC curves lying in the interval between 0.61 and 0.79 on the different datasets, the topic models perform significantly better than random prediction. This means that it is possible to extract some information about the graph structure using author and topic modelling, which might prove useful for inferring relations in contexts with a latent link structure.

Appendix A

Software

A python implementation of both LDA and AT with constant hyper parameters have been developed and can be obtained upon personal request.
Most of the computations using the author-topic model, was carried out utilising a modified version of source code from the GibbsLDA++ [PN07] project. The source modified source code implements the AT model including means to optimise hyper parameters.
Python tools for extracting and processing tweets from the SNAP twitter data set (no longer available) have been developed as well. The tweets were organised using the Kyoto Cabinet dbm software tools and libraries for python and C++.

Bibliography

[AA03] L. A. Adamic and E. Adar. Friends and neighbors on the Web. *Social Networks*, 25(3):211–230, July 2003.

[ACM07] ACM. *Expertise modeling for matching papers with reviewers*, 2007.

[AGH$^+$05] Shivani Agarwal, Thore Graepel, Ralf Herbrich, Sariel Har-Peled, and Dan Roth. Generalization Bounds for the Area Under the ROC Curve. *Journal of Machine Learning Research*, 6:393–425, April 2005.

[AWST09] Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee-Whye Teh. On smoothing and inference for topic models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 27–34, Arlington, Virginia, United States, 2009. AUAI Press.

[BM22] H.A. Bohr and J. Mollerup. *Lærebog i matematisk analyse af Harald Bohr og Johannes Mollerup*. Number v. 3 in Laerebog i matematisk analyse af Harald Bohr og Johannes Mollerup. J. Gjellerups, 1922.

[BNJ03] David M. Blei, Andrew Ng, and Michael Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[Car10] Bob Carpenter. Integrating Out Multinomial Parameters in Latent Dirichlet Allocation and Naive Bayes for Collapsed Gibbs Sampling. Technical report, LingPipe, 2010.

[CBGG$^+$09] Jonathan Chang, Jordan Boyd-Graber, Sean Gerrish, Chong Wang, and David M. Blei. Reading tea leaves: How humans interpret topic models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Neural Information Processing Systems 22*, pages 288–296, 2009.

[CMN08] A. Clauset, C. Moore, and M.E.J. Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98–101, 2008.

[DFL$^+$88] S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman. Using latent semantic analysis to improve access to textual information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '88, pages 281–285, New York, NY, USA, 1988. ACM.

[DN92] Susan T. Dumais and Jakob Nielsen. Automating the assignment of submitted manuscripts to reviewers. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '92, pages 233–244, New York, NY, USA, 1992. ACM.

[GH99] Daniel Gildea and Thomas Hofmann. Topic-based language models using em. In *Proceedings of the Sixth European Conference on Speech Communication and Technology*, pages 2167–2170. EUROSPEECH, 1999.

[GPV11] Bruno Goncalves, Nicola Perra, and Alessandro Vespignani. Modeling Users' Activity on Twitter Networks: Validation of Dunbar's Number. *PLoS One*, 6(8):e22656+, May 2011.

[GS04] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *PNAS*, 101(suppl. 1):5228–5235, 2004.

[GSBT05] T.L. Griffiths, M. Steyvers, D.M. Blei, and J.B. Tenenbaum. Integrating topics and syntax. *Advances in neural information processing systems*, 17:537–544, 2005.

[HBB10] Matthew Hoffman, David M. Blei, and Francis Bach. Online learning for latent dirichlet allocation. In *NIPS*, 2010.

[HD10] Liangjie Hong and Brian D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 80–88, New York, NY, USA, 2010. ACM.

[Hei04] Gregor Heinrich. Parameter estimation for text analysis. Technical report, University of Leipzig, 2004.

[HG06] Bo-June (Paul) Hsu and James Glass. Style & topic language model adaptation using hmm-lda. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 373–381, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[HGX09] Timothy Hospedales, Shaogang Gong, and Tao Xiang. A markov clustering topic model for mining behaviour in video. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1165 –1172. IEEE, October 2009.

[HM82] J A Hanley and B J McNeil. The meaning and use of the area under a receiver operating (roc) curvel characteristic. *Radiology*, 143(1):29–36, 1982.

[Hof99] Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *In Proc. of Uncertainty in Artificial Intelligence, UAI'99*, pages 289–296, 1999.

[HSK$^+$00] Lars Kai Hansen, Sigurdur Sigurdsson, Thomas Kolenda, Finn Årup Nielsen, Ulrik Kjems, and Jan Larsen. *Modeling text with generalizable Gaussian mixtures*, volume 6, pages 3494–3497. IEEE, 2000.

[KL11] Myunghwan Kim and Jure Leskovec. The Network Completion Problem: Inferring Missing Nodes and Edges in Networks. In *SDM*, pages 47–58. SIAM / Omnipress, 2011.

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 591–600, New York, NY, USA, 2010. ACM.

[KNS09] S. Kim, S. Narayanan, and S. Sundaram. Acoustic topic model for audio information retrieval. In *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09. IEEE Workshop on*, pages 37 –40, oct. 2009.

[KZB08] Maryam Karimzadehgan, ChengXiang Zhai, and Geneva Belford. Multi-aspect expertise matching for review assignment. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 1113–1122, New York, NY, USA, 2008. ACM.

[LLC10] Ryan N. Lichtenwalter, Jake T. Lussier, and Nitesh V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge* *discovery and data mining*, KDD '10, pages 243–252, New York, NY, USA, 2010. ACM.

[LMD10] M. Lienou, H. Maitre, and M. Datcu. Semantic annotation of satellite images using latent dirichlet allocation. *Geoscience and Remote Sensing Letters, IEEE*, 7(1):28 –32, jan. 2010.

[LNK07] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58(7):1019–1031, 2007.

[LNMG09] Yan Liu, Alexandru Niculescu-Mizil, and Wojciech Gryc. Topic-link lda: joint models of topic and author community. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 665–672, New York, NY, USA, 2009. ACM.

[Min00] Thomas P. Minka. Estimating a Dirichlet distribution, 2000. Available at http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/.

[ML02] Thomas Minka and John Lafferty. Expectation-Propagation for the Generative Aspect Model. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, pages 352–359, 2002.

[MWCE07] Andrew McCallum, Xuerui Wang, and Andrés Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *J. Artif. Int. Res.*, 30(1):249–272, October 2007.

[NASW09] David Newman, Arthur Asuncion, Padhraic Smyth, and Max Welling. Distributed algorithms for topic models. *J. Mach. Learn. Res.*, 10:1801–1828, December 2009.

[NC08] R. Nallapati and W. Cohen. Link-pLSA-LDA: A new unsupervised model for topics and influence of blogs. *International Conference for Weblogs and Social Media*, 2008.

[PECX10] Kriti Puniyani, Jacob Eisenstein, Shay Cohen, and Eric P. Xing. Social links from latent topics in microblogs. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pages 19–20, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[PG11] Marco Pennacchiotti and Siva Gurumurthy. Investigating topic models for social media user recommendation. In *Proceedings of* *the 20th international conference companion on World wide web*, WWW '11, pages 101–102, New York, NY, USA, 2011. ACM.

[PN07] Xuan-Hieu Phan and Cam-Tu Nguyen. Gibbslda++, 2007.

[Pot11] Christopher Potts. Twitter-aware tokenizer. "http://sentiment.christopherpotts.net/code-data/happyfuntokenizing.py", 2011. Downloaded 14$^{\text{th}}$ of March 2012.

[RFE$^+$06] B.C. Russell, W.T. Freeman, A.A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1605–1614, 2006.

[RZCG$^+$10] M. Rosen-Zvi, C. Chemudugunta, T. Griffiths, P. Smyth, and M. Steyvers. Learning author-topic models from text corpora. *ACM Transactions on Information Systems (TOIS)*, 28(1):4:1–4:38, 2010.

[SG05] Mark Steyvers and Thomas Griffiths. Probabilistic topic models. book chapter published online, March 2005.

[SN10] Alexander Smola and Shravan Narayanamurthy. An architecture for parallel topic models. *Proc. VLDB Endow.*, 3:703–710, September 2010.

[SSRZG04] Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas Griffiths. Probabilistic author-topic models for information discovery. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 306–315, New York, NY, USA, 2004. ACM.

[TNW07] Y.W. Teh, D. Newman, and M. Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. *Advances in neural information processing systems*, 19:1353–1360, 2007.

[Twi12] Twitter.com. How to change your username. "https://support.twitter.com/groups/31-twitter-basics/topics/107-my-profile-account-settings/articles/14609-how-to-change-your-username", June 2012. On-line at 2$^{\text{nd}}$ of June 2012.

[Wal06] Hanna M. Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 977–984, New York, NY, USA, 2006. ACM.

[Wal08] Hanna M. Wallach. *Structured Topic Models for Language*. PhD thesis, University of Cambridge, 2008.

[Wan08] Yi Wang. Distributed gibbs sampling of latent topic models: The gritty details, 2008.

[WLJH10] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM.

[WM09] Yang Wang and G. Mori. Human action recognition by semilatent topic models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(10):1762 –1774, oct. 2009.

[WMM09] Hanna Wallach, David Mimno, and Andrew McCallum. Rethinking lda: Why priors matter. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1973–1981. Curran Associates, Inc., 2009.

[YHD11] Dawei Yin, Liangjie Hong, and Brian D. Davison. Structural link analysis and prediction in microblogs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 1163–1168, New York, NY, USA, 2011. ACM.

[YL11] Jaewon Yang and Jure Leskovec. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 177–186, New York, NY, USA, 2011. ACM.

[YMM09] Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 937–946, New York, NY, USA, 2009. ACM.

[ZCL11] Jia Zeng, William K. Cheung, and Jiming Liu. Learning topic models by belief propagation. *ArXiv e-prints*, September 2011.