THE UNIVERSITY of EDINBURGH

BAYESIAN NONPARAMETRIC MODELS FOR NAME DISAMBIGUATION AND SUPERVISED LEARNING

ANDREW M. DAI

Doctor of Philosophy

Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh

2013

ABSTRACT

This thesis presents new Bayesian nonparametric models and approaches for their development, for the problems of name disambiguation and supervised learning. Bayesian nonparametric methods form an increasingly popular approach for solving problems that demand a high amount of model flexibility. However, this field is relatively new, and there are many areas that need further investigation. Previous work on Bayesian nonparametrics has neither fully explored the problems of entity disambiguation and supervised learning nor the advantages of nested hierarchical models. Entity disambiguation is a widely encountered problem where different references need to be linked to a real underlying entity. This problem is often unsupervised as there is no previously known information about the entities. Further to this, effective use of Bayesian nonparametrics offer a new approach to tackling supervised problems, which are frequently encountered.

The main original contribution of this thesis is a set of new structured Dirichlet process mixture models for name disambiguation and supervised learning that can also have a wide range of applications. These models use techniques from Bayesian statistics, including hierarchical and nested Dirichlet processes, generalised linear models, Markov chain Monte Carlo methods and optimisation techniques such as BFGS. The new models have tangible advantages over existing methods in the field as shown with experiments on real-world datasets including citation databases and classification and regression datasets.

I develop the unsupervised author-topic space model for author disambiguation that uses free-text to perform disambiguation unlike traditional author disambiguation approaches. The model incorporates a name variant model that is based on a nonparametric Dirichlet language model. The model handles both novel unseen name variants and can model the unknown authors of the text of the documents. Through this, the model can disambiguate authors with no prior knowledge of the number of true authors in the dataset. In addition, it can do this when the authors have identical names.

I use a model for nesting Dirichlet processes named the hybrid NDP-HDP. This model allows Dirichlet processes to be clustered together and adds an additional level of structure to the hierarchical Dirichlet process. I also develop a new hierarchical extension to the hybrid NDP-HDP. I develop this model into the grouped author-topic model for the entity disambiguation task. The grouped author-topic model uses clusters to

model the co-occurrence of entities in documents, which can be interpreted as research groups. Since this model does not require entities to be linked to specific words in a document, it overcomes the problems of some existing author-topic models. The model incorporates a new method for modelling name variants, so that domain-specific name variant models can be used.

Lastly, I develop extensions to supervised latent Dirichlet allocation, a type of supervised topic model. The keyword-supervised LDA model predicts document responses more accurately by modelling the effect of individual words and their contexts directly. The supervised HDP model has more model flexibility by using Bayesian nonparametrics for supervised learning. These models are evaluated on a number of classification and regression problems, and the results show that they outperform existing supervised topic modelling approaches. The models can also be extended to use similar information to the previous models, incorporating additional information such as entities and document titles to improve prediction.

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere thanks to my supervisor, Amos Storkey, for all the encouraging support, valuable advice and patient guidance he has given me during my studies. His knowledge and insight have helped guide me around a number of obstacles and towards more robust methods. I am also very grateful for him for giving me the opportunity to pursue internships. I would like to thank my second supervisor, Chris Williams, for advice and the broad knowledge of existing work that he brought into discussions. I have been very lucky to be a student at the Institute for Adaptive and Neural Computation, which has shown me how people in very different fields can work and collaborate. I would like to thank faculty members, including Iain Murray, Sharon Goldwater and Guido Sanguinetti for their valuable discussions.

I would like to thank the current and past members of PIGlets including Felix Agakov, Edwin Bonilla, Kian Ming Chai, Ali Eslami, Nicholas Heess, Jyri Kivinen, Lawrence Murray, Peter Orchard, David Reichert and Athina Spiliopoulou for their input and discussions during the many journal club and brainstorm meetings.

I am grateful to the Google Adsense and Google Translate teams, who gave me the chance to understand how really large problems are tackled at a huge scale. I would like to thank the people I worked with there including Ashok Popat, Klaus Macherey, David Talbot and Franz Och for their valuable advice and discussions.

I would like to thank people for their help and support at various stages of my Ph.D., including Jonathan Chang, Jun Zhu and Li Xiang.

I am grateful to those who financially supported me, including the EPSRC and the PASCAL network of excellence.

Finally, I would like to thank my parents, Jian Dai and Jian Cong, for their advice, long support and valuable input through these years, and from various parts of the world.

DECLARATION

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Andrew M. Dai

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

This thesis presents new methods for modelling texts, their authors and their other metadata. This motivation originates from the huge difficulties in analysing the increasing number of document collections and databases. These collections are often large, preventing the use of traditional and manual text analysis techniques, and often contain errors and mistakes, requiring error-tolerant methods. Being able to integrate these collections is valuable as it can often reveal additional patterns or information in the data that cannot be obtained from any individual source. However, there are many obstacles to integration. A lack of unique identifiers for each object, such as people or documents, is one obstacle that can result in information being duplicated, and incomplete or missing document metadata can be another obstacle since it can be expensive to manually fill in the metadata fields for new documents. Therefore, the ability to predict document metadata based on document text would be highly useful.

Further, authors of documents are not typically associated with personally identifiable information such as national identity numbers. This can be due to privacy concerns, for individuals may not wish to be linked across different databases. Another reason may be the extra work involved in tracking down personally identifiable information. More generally, the problem of record linkage involves finding duplicate records or record fields which refer to the same entity, or the same author in this case. Names are usually used to match together authors, where in some methods, two identical names are assumed to refer to the same person. However, names are often ambiguous in that multiple people share the same name, and names have variations in that the same person can be referred to by different names. Name ambiguity is an increasingly common problem given the growing volume of document collections. Name variation is also common due to different ways of formatting a name and as a result of name initialling, aliases and typographical errors.

Following these problems, resolving an author name into a person requires analysing any extra data that appears with the name. This could include the email address of the author or the affiliation of the author. But both properties become obsolete when a person changes institution. These pieces of information are also usually missing from author metadata. On the other hand, the title or abstract of a document is commonly available and can be used to help resolve names. For example, two authors working in

unrelated fields of expertise are unlikely to be the same person even if they have similar names. But two authors working on the same field may be the same person even if their names are slightly different. Existing techniques for solving this problem often calculate pairwise distances between names and use these to cluster similar names together, where a cluster represents the references or names for a single person or entity. Even though they perform well, these techniques cannot evaluate a cluster as a whole and are often unable to estimate the probability of a name belonging to a cluster. These existing techniques also fail to make use of coauthorship relationships within a document, where resolving one name helps resolve the others.

Finally, being able to predict document metadata from the text of a document is highly useful. The metadata might include labels such as category, rating, popularity or other information. Given a set of examples of documents and their metadata, it would be useful to have a model that can learn to predict the metadata of an unseen document. Supervised latent Dirichlet allocation, a type of supervised topic model, is an existing model for this problem, but it has shortcomings in that the number of topics must be fixed in advance and the effect of different words on the document labels is not modelled directly.

This thesis addresses the problems of these existing techniques and presents four new approaches for solving these problems by developing new models and by using ideas from Bayesian nonparametrics, topic models and generalised linear models. The thesis is laid out as follows.

- In Chapter 2, I describe the background to the name disambiguation problem and review existing approaches.

- In Chapter 3, I give an overview of Bayesian methods, which are relevant to this thesis. In addition, I review some of the main Bayesian nonparametric techniques, which are a recurring theme in later chapters.

- In Chapter 4, I present the *author-topic space model for disambiguation*. This is a novel unsupervised model that integrates name variation modelling and topic modelling. Two name variation models are proposed, one is based on the bigram topic model and the other is based on a bag-of-words assumption for a set of character-level trigrams. The models are evaluated with two different inference algorithms on real-world datasets.

- In Chapter 5, I extend the work in Chapter 4 and present the *grouped author-topic model*, which is a new unsupervised model that uses coauthorship relationships along with document text to improve disambiguation. The chapter also describes

an extension of the nested Dirichlet process. This work was presented at the NIPS workshop on applications for topic models (Dai and Storkey, 2009) and at ICANN (Dai and Storkey, 2011).

- In Chapter 6, I present two novel models for supervised learning with topic models. The *keyword-supervised LDA* model directly models the effect of different words on document labels, giving more accurate label predictions than existing models. The model also allows words to have different effects on the document labels according to their context. The *supervised HDP* model is an extension of supervised latent Dirichlet allocation to infinite topics. These models are evaluated on classification and regression tasks and compared with existing models.

- In Chapter 7, I give an overview and comparison of the models presented in this thesis.

- Finally, in Chapter 8, I review the contributions of the thesis and discuss future work.

2

BACKGROUND

In this chapter, I describe the background to the problems to be tackled in later chapters and existing approaches to solving the problems.

2.1 NAME DISAMBIGUATION

Name disambiguation or *identity disambiguation* is the task of linking real people identities with their names in databases and records. It is highly similar to the problem of *record linkage* (Fellegi and Sunter, 1969), also known as *de-duplication*, which is an important problem in data integration and data mining. Data is stored using many methods and formats and with many different conventions. As a result, it can be difficult to gain the benefits that result from integrating data from different sources. The most commonly encountered problem is performing name or entity de-duplication. Many data sources about people, for example, neither record a person's national identification number nor any other personally identifiable information nor store information that can be easily cross-referenced with other data sources. The terms *person*, *entity* and *identity* are often interchangeable when referring to these problems.

Fellegi and Sunter (1969) developed a framework by Newcombe et al. (1959) into a mathematical formulation for record linkage, with the emphasis on developing an efficient linkage rule. This fundamental model has gone through much analysis and development. These approaches all went through the same process of defining a similarity measure and then specifying and estimating distributions for matches and non-matches. The methods were motivated by a need for linkage in census data (Hogan, 1992; Winkler and Thibaudeau, 1991), but it has also seen much use in medical data (Torvik et al., 2005), brain imaging (Clayden, Storkey and Bastin, 2007) and citation indexing (Giles, Bollacker and Lawrence, 1998). Coupled clustering (Marx et al., 2002) is also a relevant method that uses linked information from multiple datasets to enable better clustering than can be obtained from a single dataset. An overview of some recent approaches can be found in Winkler (2006).

Most common approaches calculate pairwise scores and match records on the basis of those scores, though alternative approaches can have better performance, such as those that can compute scores for entire clusters of records or can better model prediction

errors. This indicates that more sophisticated models can better solve the problem such as those from machine learning. Machine learning techniques (Mitchell, 1997; Bishop, 2006) are resistant to noisy data and work well with unstructured data. They include models such as the single-layer perceptron and naive Bayes models.

Some approaches also exploit the large quantity of data on the Internet (Tan, Kan and D. Lee, 2006). Other methods for disambiguation use other fields of documents including co-authors, author emails, institution affiliation and venue of publication (Culotta et al., 2007). In general, unified approaches such as these are needed for the best disambiguation performance, though they come with a computational cost.

The core of these methods usually involves string matching. This is the task of matching together strings which likely refer to the same thing, for which nearest neighbour methods are often used. One area of development that can help to reduce the cost associated with finding these nearest neighbours during clustering is the locality sensitive hashing (LSH) method (Indyk and Motwani, 1998). LSH is related to the idea of dimensionality reduction in that similar objects are probabilistically mapped to the same set of buckets. These developments can significantly improve the performance of some of the clustering approaches by allowing the nearest neighbour search to be completed in faster than linear time. Comparisons of string matching algorithms and metrics for names have been made by Cohen, Ravikumar and Fienberg (2003) and Christen (2006). They compared methods including phonetic techniques such as Soundex and pattern matching techniques such as n-gram similarity and edit distances. N-grams are sequences of words or characters of length n commonly used as a similarity measure. They found that small changes in similarity thresholds can significantly affect name matching performance and that thresholds can vary between datasets. They also could not find any single best technique for name matching. Although they found that an n-gram similarity approach was competitive with the Jaro and Winkler algorithms. Further, Gong, L. Wang and Oard (2009) proposed modelling name similarity by finding the minimum number of transformations applied to a name such as abbreviation, omission and rearranging.

Authorship attribution is a related problem to *name disambiguation* where the goal is to find the authorships of anonymous text solely based on the text itself. The problem has a long history as there has long been interest in identifying the true author of texts written under a pseudonym or when the author of old texts may be unknown. Traditional methods have relied on examining writing styles using text features such as word length and sentence length whereas more recently, statistical methods have been used to analyse word frequencies and find common patterns that a particular author uses. Mosteller and Wallace (1964) was one of the most influential works in authorship attribution. They used Bayesian statistical techniques to find the authorship of *The*

Federalist Papers, of which 12 essays have unclear authorship. Stamatatos (2009) and Koppel, Schler and Argamon (2009) give surveys of past and recent methods used for authorship attribution. Modern techniques such as SVMs have also been applied to the attribution problem (Diederich et al., 2003), which have been valuable since the input text has high dimensionality. Recent techniques have used learning ensembles of character-level n-grams (Stamatatos, 2006) to represent the style of a text and identify the author. More recent applications for these methods include plagiarism detection and author verification.

2.1.1 *Citation matching*

Automated bibliographic services such as CiteSeerX (Giles, Bollacker and Lawrence, 1998), Google Scholar (Google, 2004) and Rexa (Wellner et al., 2004) are growing in popularity thanks to the speed at which they index new publications and the breadth of publications that they index. A common task when working with bibliographic databases or proceedings is to cross-reference the references in a paper with papers in the database. This is known as citation matching and can be thought of as entity disambiguation where the entities are papers. Since papers are usually referenced without unique identifiers such as digital object identifiers (DOIs), citations are often matched by titles, authors and venues. Complications can arise, however, when citation formats change, such as in the order of the fields or in the varying punctuation styles that are used to separate citation fields. Abbreviations are also used inconsistently, most often for defining venue names (e.g. Int. for International). Finally, author names can be written in confusing formats or orders, in which it can be hard to distinguish the first, middle and last names. This has led to many wrongly filled out fields in automated citation indices. Citation matching is also a popular application for the record linkage methods as test citation data is often freely available.

Approaches include rules-based methods, which can quickly detect errors such as dates in the wrong formats. However, the number of exceptions to the rules quickly grow to be unwieldy, as is often the case with varying address formats. The problem can be tackled by both relational and machine learning techniques. Relational models have moved beyond pairwise matching, and modern methods learn both pairwise and clusterwise scoring functions to integrate cluster-level information. Relational models are also good at integrating data using rules and multiple data fields, such as for structured data in databases. They include approaches such as relational clustering (Bhattacharya and Getoor, 2007), the relational probability model (Pasula et al., 2003) and the probabilistic relational model (J. Li, G. A. Wang and Chen, 2008) to capture depend-

encies between co-authors and papers. Dredze et al. (2010) also proposed a model that integrates information from other sources. Methods have also been explored for scaling the task to large datasets. One approach is the canopy approach (McCallum, Nigam and Ungar, 2000). This uses a cheap approximate distance to find overlapping subsets containing data elements which are potentially close to each other. A more expensive distance metric is used between elements that appear in the same subset. Elements that do not appear in the same subset are assumed not to match.

The problem with many of these approaches is that they need a threshold or the number of author identities to be fixed in advance. To overcome this problem, probabilistic, nonparametric approaches have been explored. Bhattacharya and Getoor (2006) presented the LDA-ER model, which is based on latent Dirichlet allocation, and which can infer the number of author entities in the data. However, their method requires pre-specifying the number of author groups and their algorithm assumes that author references with identical names refer to the same entity, which is troublesome for ambiguous names. Daumé III and Marcu (2005) showed that supervised training of Dirichlet processes (DPs) provided gains over traditional supervised clustering approaches. They also found that the number of entities in a citation database, along with similar linkage problems, match the expected number of clusters under a DP. This showed that DPs are an appropriate model for these kinds of problems.

Though these techniques have worked well, many of them have difficulties in integrating information from multiple fields. Often, models which use information from other fields perform better than those that solely disambiguate based on names (Hall, Sutton and McCallum, 2008). The author-topic model (Rosen-Zvi et al., 2004) is one example that associates latent topics with authors and identifies the topics that authors frequently write on. In these works, a latent topic is characterised by a distribution over the vocabulary of the corpus. However, instead of entity resolution, their goal was to model the tendencies of authors to write on certain topics or subject areas assuming the authors for each document in the training set are already known. The model then allocates words in the document to one of the known authors and does not model author names so cannot take name variation into account. A similar generative model was presented by Newman, Chemudugunta and Smyth (2006) for the entities in a document, where the task is again to associate entities with topics. However, both these approaches can require a significant amount of data in that, an author must appear many times in a corpus for that author's topic distribution to be sufficiently peaked to allow for disambiguation.

2.2 COREFERENCE RESOLUTION

Coreference resolution is also related to the problem of name disambiguation. Coreference occurs when multiple phrases in a document refer to the same thing, in this case the phrases are said to be coreferent. There are two variants of coreference resolution, within-document and cross-document. These problems extend the classical problem of record linkage by considering situations where additional information about the entity is available, an entity can be referenced in a greater number of ways and entity metadata must be extracted from the text. For example, additional considerations can include the free text of a document, the context of the entity reference, the entity's gender or other biographical information about the entity.

Within-document coreference resolution is the problem where an entity is referenced first as a named entity in a document (e.g. *David Cameron*) and then future references in that document may consist of nominal references (e.g. *the prime minister*), pronominal references (e.g. *he*) or named references using aliases (e.g. *David*). This is a harder problem than named entity recognition as linking together these references is dependent on semantics and syntax. The task involves creating clusters of references where each cluster contains all and only those references that refer to the same real-world identity.

Tackling the problem involves generating a coreference chain that links together pairs of anaphors and antecedents using features extracted from noun phrases in the document. For example, *Ms. Thatcher–her–She–Ms. Thatcher*. Soon, H. T. Ng and Lim (2001) proposed one of the first learning approaches for within-document coreference resolution. This involved learning a decision tree classifier on noun phrases. Later approaches used various linguistic and feature set extensions (V. Ng and Cardie, 2002). Approaches also include those that can generate multiple candidate clusterings (V. Ng, 2005) and then training a support vector machine (SVM) ranker to rank the clusterings. A supervised method was proposed by Wick, Culotta et al. (2009) that in addition to resolving entities, also canonicalises the entities by finding a standardised representation of the entity.

Unsupervised models, which are models that do not require a training dataset of labelled examples, have also been proposed for the problem. Haghighi and Klein (2007) proposed an unsupervised generative model, which is based on the hierarchical Dirichlet process, that explicitly models pronoun heads and salience. Their model yields results only slightly lower than supervised systems. V. Ng (2008) modified their method to use an EM clustering algorithm but showed that performance is still not comparable to a fully supervised coreference system. Poon and Domingos (2008) developed an unsupervised system that is based upon Markov logic networks. Finkel and C. D. Man-

ning (2008) explored the use of transitivity constraints enforced using integer linear programming (ILP). Their approach uses logistic regression to classify entity references and then an ILP solver to find the most probable solution under the constraints. Haghighi and Klein (2009) developed an unsupervised pairwise model focusing on syntactic and semantic information and using it to decide if two references refer to the same entity. An overview of recent developments in supervised coreference resolution was offered by V. Ng (2010).

There have been many competitions that recognised the problem of coreference resolution, though most of these have been for within-document coreference resolution. Recently there have been large competitions focused on the problem of cross-document coreference resolution. This variant is when an entity is referenced across several documents and the references must be resolved to the correct entities. The problem is harder than within-document coreference resolution as there are additional difficulties. For example, different entities can have the same name or the same entity might be referred to by different names. However, this problem is more general, and there are more applications including entity tracking, name disambiguation and alias identification. Cross-document coreference resolution has been the focus of the SEMEVAL 2007 workshop on *web people disambiguation* (WePS) (Artiles, Gonzalo and Sekine, 2007), later WePS workshops (Artiles, Gonzalo and Sekine, 2009; Artiles, Borthwick et al., 2010) and the *Global Entity Detection and Recognition* task of the NIST Automated Content Extraction (ACE) 2008 evaluation (NIST Speech Group, 2008).

The majority of cross-document coreference resolution systems use agglomerative clustering. Bagga and Baldwin (1998b) proposed one of the earliest systems for cross-document coreference resolution based on an existing within-document coreference resolution system. The entity clusters for each document that are found by the within-document coreference system are passed through a sentence extractor that extracts sentences relevant to each of the entities. The vector space model is used by storing the sentences as vectors of terms. This allows the cosine similarity to be calculated between the sentences for each pair of entities. If the similarity is above a preset threshold then the two clusters are considered to be coreferent, that is they refer to the same entity. Other approaches generally involve extending the feature set or improving feature extraction. These include using patterns to extract biographical facts and using these as features in a centroid agglomerative clustering algorithm (Mann and Yarowsky, 2003). Their system is evaluated on artificial test data where pseudo-names are generated and real entities are merged together. Gooi and Allan (2004) extended the use of pseudo-names even further by creating a test set where each name reference is rewritten to *Person X*. There have also been approaches that use classifiers. Fleischman and Hovy (2004)

trained maximum entropy classifiers to model the probability that two references refer to the same entity and then used this classifier in an agglomerative clustering algorithm. They used a wide range of features including the frequency of the name in census data, the number of hits returned from web queries and semantic distance obtained from the WordNet ontology.

Several approaches also use information extraction systems to extract named entities and entity attributes from data surrounding the entity. Baron and Freedman (2008) explicitly handled the problems of name variation and entity disambiguation. They used an existing information extraction system along with web-mined alias lists, character-level edit distance and so on to model name variation. To model name ambiguity, they used features such as name uniqueness in Wikipedia. The names are then clustered using an agglomerative clustering algorithm. There have also been approaches that use fuzzy clustering instead of the previous hard clustering algorithms. Fuzzy clustering allows references to be associated with more than one entity. Huang et al. (2009) used an information extraction system to extract entity attributes and relations and then performed clustering using a kernelised fuzzy clustering algorithm. Finin et al. (2009) used a knowledge base that was constructed from various online databases. They used features extracted from this knowledge base to train a support vector machine (SVM) classifier to determine which pairs of references refer to the same entity. Mayfield et al. (2009) used a within-document coreference resolution system using soundex, character n-gram similarity, hashing and name alias lists to account for name variation. They also used features including document similarity, thesauruses and information from Wikipedia. They then trained SVM and decision tree classifiers to determine if any two clusters refer to the same entity. The classifiers were run on pairs of clusters output by the within-document resolution system with the result given by the transitive closure of the pairs.

Finally, there has been work on tackling coreference resolution in large-scale data. Rao, McNamee and Dredze (2010) described a system that uses a streaming clustering algorithm and name hashing to scale to streaming data. Singh et al. (2011) describe a distributed parallel inference technique to disambiguate authors and Wick, Singh and McCallum (2012) describe a hierarchical latent entity model for disambiguating at large scales. Entity disambiguation can also potentially be used for word sense disambiguation and other problems in machine translation (Macherey et al., 2011).

Methods for phrase modelling are also relevant to the problem of coreference resolution. Though topic models are typically applied on unigrams with the bag of words assumption, developments have been made towards latent Dirichlet allocation (LDA)

n-gram models (Wallach, 2006; X. Wang, McCallum and Wei, 2007) that help prevent the over-conflation of terms and topics as datasets grow larger.

A problem with many of the previous methods is that they are supervised and require similarity thresholds to be trained on a training set or they may need outside sources such as the web or Wikipedia. Many of them are also discriminative rather than generative, which means they often cannot take advantage of unlabelled data and they may have difficulty integrating new information.

3

BAYESIAN METHODS

Bayesian statistics is a system for describing uncertainty, arising from lack of knowledge about the world, using probability. Bayesian statistical methods begin with a set of prior beliefs and repeatedly update these with data to give posterior beliefs. The posterior beliefs can then be used for statistical inference. Bayesian methods are an important part of modern machine learning, since they give a principled way to update a model when new information is seen. Traditionally, due to the significantly increased computational cost needed for Bayesian methods, Bayesian methods could only be used for conjugate analysis, where the prior and likelihood must be jointly chosen so that the posterior can be tractable. Recent improvements in computational performance and advances in Markov chain Monte Carlo (MCMC) have made Bayesian methods computationally tractable and more attractive. The main difference between Bayesian methods and frequentist methods is the idea that uncertainty about model parameters can be expressed in Bayesian methods using probability distributions, commonly known as prior distributions, and the use of probabilistic inference through marginalisation of unknowns.

Statistical inference in general is about discovering information about unobserved quantities. These can usually be divided into two types, the potentially observable quantities such as those from future data and the unobservable quantities such as parameters. Parameters that are not directly observable are denoted θ and the observed data are denoted x. Later in this chapter, Greek letters are used for parameters and Roman letters for observable quantities. $p(\cdot|\cdot)$ denotes a conditional probability density and $p(\cdot)$ denotes a marginal density. When referring to standard probability distributions I will write $\theta \sim \mathrm{N}(\mu, \sigma^2)$ to mean $p(\theta) = \mathrm{N}(\theta|\mu, \sigma^2)$ where θ has a normal distribution with a mean of μ and variance σ^2.

Usually a set of n observations, $x = (x_1, \ldots, x_n)$ are assumed to be exchangeable. This means that $p(x_1, \ldots, x_n)$ is invariant under permutation of the indices. A nonexchangeable model would mean that some information is contained in the unit indices. In this thesis, the words in documents are assumed to be exchangeable. This is called the bag-of-words assumption, and even though the assumption is unrealistic, it is a reasonable assumption to make if the goal is to infer the abstract semantic themes in the corpus. In general, observing a set of key words in a document is enough to work out

the theme of the document regardless of the order of those words. However, this may not hold when the key words can be used in many different contexts, in which case a phrase would better identify the theme. For example, the phrase *investment bank* is more useful at identifying the theme of investment banking than *investment* or *bank*. In addition, some themes may be very popular resulting in many different authors writing on that theme. Thus, more information than the theme of a document is needed to disambiguate authors.

Bayesian inference is usually used to make statements about θ given x. To do this, a model is constructed giving a joint probability distribution for θ and x. This distribution is usually factored into a product of two distributions, the prior distribution $p(\theta)$ and the likelihood $p(x|\theta)$:

$$p(\theta, x) = p(\theta)p(x|\theta). \tag{3.1}$$

Using Bayes' theorem, which states that

$$p(\theta|x) = \frac{p(\theta)p(x|\theta)}{p(x)} \tag{3.2}$$

and conditioning on the known x we can arrive at the posterior density:

$$p(\theta|x) = \frac{p(\theta, x)}{p(x)} = \frac{p(\theta)p(x|\theta)}{\sum_\theta p(\theta)p(x|\theta)} \tag{3.3}$$

where the denominator is replaced with $\int p(\theta)p(x|\theta)\,\mathrm{d}\theta$ when θ is continuous. $p(x)$ is usually considered a normalising constant as x is fixed.

To make inferences about unknown observable quantities before observing any data, the distribution for the unknown quantities can be found by marginalising θ:

$$p(x) = \int p(x, \theta)\,\mathrm{d}\theta = \int p(\theta)p(x|\theta)\,\mathrm{d}\theta. \tag{3.4}$$

This gives the marginal distribution of x, also known as the prior predictive distribution of x since it is not conditioned on previous observations. After conditioning

on previous observations, we get the posterior predictive distribution for unobserved future quantities x^*:

$$p(x^*|x) = \int p(x^*, \theta|x)\,\mathrm{d}\theta \tag{3.5}$$

$$= \int p(x^*|\theta, x)p(\theta|x)\,\mathrm{d}\theta \tag{3.6}$$

$$= \int p(x^*|\theta)p(\theta|x)\,\mathrm{d}\theta. \tag{3.7}$$

which follows since x and x^* are conditionally independent given θ.

The likelihood function $p(x|\theta)$ is treated as a function of θ for fixed x. Since the data affect the inferences solely through the likelihood function, Bayesian inference follows the likelihood principle. This states that all the information in a set of data is contained within the likelihood function.

Bayesian modelling is concerned with finding the posterior distribution over the model parameters. Bayesian and frequentist methods agree on the solutions for most problems as sample size increases, however, for small sample sizes the Bayesian prior can significantly change the result.

3.1 GRAPHICAL MODELS

A graphical model is a family of probability distributions defined using a directed or undirected graph. Jordan (2004) reviewed graphical models in detail along with inference in them and their applications. The nodes in the graph represent random variables and joint probability distributions are defined by taking products over functions of the nodes. Shaded nodes denote conditioning or observed variables. This method allows for general algorithms to be used to compute marginal and conditional probabilities and additionally provides an intuitive way of describing the model. Directed graphical models (directed acyclic graphs) are commonly used to describe hierarchical Bayesian models. The models described in this thesis are represented in terms of directed acyclic graphs.

Plate notation (Buntine, 1994) is used to express repeated, shared or tied parameters in a graphical model by enclosing the corresponding nodes in plates or rectangles, with the figure in the corner indicating the number of times the nodes and arcs in that plate should be replicated.

3.2 EXPONENTIAL FAMILY MODELS

Distributions are often grouped into families that share the same parametric form but have different parameters. Exponential families are especially important in Bayesian models and have the following form.

$$p(\mathbf{x}|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} A(\mathbf{x}) \exp(\mathbf{t}(\boldsymbol{\theta})^\top \boldsymbol{\tau}(\mathbf{x})) \tag{3.8}$$

where

$$Z(\boldsymbol{\theta}) = \sum_{\mathbf{x}} A(\mathbf{x}) \exp(\mathbf{t}(\boldsymbol{\theta})^\top \boldsymbol{\tau}(\mathbf{x})) \tag{3.9}$$

is the partition function of p, which ensures the distribution is normalised and so must be finite. The likelihood of $\boldsymbol{\theta}$ depends on the data only through the function $\boldsymbol{\tau}$, so $\boldsymbol{\tau}$ is known as the sufficient statistics function, which is a function from values of $\mathbf{x}$ to the real numbers. The value of the statistic contains all the information needed to compute the posterior distribution of the parameters given the data. $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ are the parameters for the distribution $p(\cdot|\boldsymbol{\theta})$ in the family and $\boldsymbol{\Theta}$, the parameter set, is the set of legal parameters for that family. $\mathbf{t}$ is the natural parameter function that maps the parameters to the space of sufficient statistics. When $\mathbf{t}(\boldsymbol{\theta}) = \boldsymbol{\theta}$, the family is said to be in canonical form. Finally, A is a measure over $\mathbf{x}$. A is used for observations that do not depend on the parameters but later on it will be used as a constant.

The K-dimensional multinomial distribution is a multivariate generalisation of the binomial distribution and is used extensively in text modelling. The multinomial distribution models a vector of counts $\mathbf{x}$ of the number of observations for each dimension k. In the case where there is only one observation, with parameters $0 \leq \theta_k \leq 1$ and $\sum_{k=1}^{K} \theta_k = 1$, then

$$p(\mathbf{x}|\boldsymbol{\theta}) = \prod_{k=1}^{K} \theta_k^{x_k} \tag{3.10}$$

which can be written in exponential family form as

$$p(\mathbf{x}|\boldsymbol{\theta}) = \exp\left(\sum_{k=1}^{K} x_k \ln \theta_k\right) \tag{3.11}$$

$$= \exp(\mathbf{t}(\boldsymbol{\theta})^\top \mathbf{x}) \tag{3.12}$$

for

$$t(\theta_k) = \ln \theta_k \tag{3.13}$$

$$\tau(\mathbf{x}) = \mathbf{x} \tag{3.14}$$

$$A(\mathbf{x}) = 1 \tag{3.15}$$

$$Z(\boldsymbol{\theta}) = 1. \tag{3.16}$$

Exponential families have many properties useful for statistical inference. One of these is that only exponential family distributions have a sufficient statistic whose dimension remains bounded as the sample size of the data increases, enabling statistics for large sample sizes to be stored efficiently. Exponential family distributions are also at the core of distributions used in generalised linear models. Finally, with certain choices of priors, known as conjugate priors, inference with exponential families is simpler than with other distributions.

3.2.1 *Conjugate priors*

One of the useful properties of exponential families is that all members of the exponential family have conjugate priors. A conjugate prior distribution $p(\boldsymbol{\theta})$ for a likelihood function $p(\mathbf{x}|\boldsymbol{\theta})$ is a distribution that will give rise to a posterior distribution that has the same functional form as the conjugate prior. This simplifies inference on models that can use them. For a member of the exponential family, its conjugate prior is:

$$p(\boldsymbol{\theta}|\boldsymbol{\chi}, \nu) = f(\boldsymbol{\chi}, \nu)\frac{1}{Z(\boldsymbol{\theta})^{\nu}} \exp(\nu \boldsymbol{\theta}^{\top} \boldsymbol{\chi}) \tag{3.17}$$

where $f(\boldsymbol{\chi}, \nu)$ is a normalisation constant and ν can be interpreted as the number of pseudo-observations in the prior, where each pseudo-observation has the sufficient statistic $\boldsymbol{\chi}$.

For the K-dimensional multinomial distribution, the conjugate prior distribution is the K-dimensional Dirichlet distribution which is a multivariate generalisation of the beta distribution. The Dirichlet distribution is defined over $0 \leq \theta_k \leq 1$ with $\sum_{k=1}^{K} \theta_k = 1$ and parameters $\alpha_k > 0$:

$$p(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \tag{3.18}$$

where

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{K} \alpha_k)} \tag{3.19}$$

This prior distribution is equivalent to the likelihood from $\sum_{k=1}^{K} \alpha_k$ observations where there are α_k observations for dimension k. Typically, a non-informative symmetric prior distribution is used in which case, α_k are all set to the same value such as $\alpha_k = 1$. The posterior distribution for $\boldsymbol{\theta}$ is a Dirichlet distribution with the new parameters $\alpha_k + x_k$. The parameters $\boldsymbol{\alpha}$ of the conjugate prior distribution are called hyperparameters.

3.3 CLUSTERING

Clustering is an unsupervised method that can be useful in recovering structure from data. Clustering methods find groups of similar data points in a dataset. There is no training dataset of labelled examples. The method is useful in identity disambiguation and record linkage problems since usually the true entity or identity of any reference is unknown, due to the name variation and name ambiguity problems. A reasonable approach is to find references or records that are most similar to one another and if they satisfy the right conditions, infer that they are coreferent. Greedy agglomerative clustering is one common method of clustering a dataset of N data points $x_1, \ldots, x_N$ into K clusters given a value for K. The algorithm proceeds iteratively where initially each data point is assigned to its own cluster. The pair of clusters closest to each other (defined by a distance metric and a linkage criterion) are found and merged and this repeats until there are K clusters remaining.

3.4 MCMC INFERENCE

One common approach for posterior inference is Markov chain Monte Carlo (MCMC) (Metropolis et al., 1953). A recent overview of MCMC is given in Brooks et al. (2011). MCMC methods scale well with the dimensionality of the samples and are often easy to implement. The method works by iteratively drawing θ from approximate distributions that depend on the last value drawn. These approximate distributions are improved at each step of the process and converge to the target distribution $p(\theta|x)$. This is necessary when it is too inefficient to sample directly from $p(\theta|x)$.

MCMC began as a way to find out about the thermodynamic equilibrium of a liquid by simulating the dynamics of the system, running it until it reached equilibrium. The major idea was that the exact dynamics did not need to be simulated, only a Markov chain that has the same equilibrium distribution. The Metropolis algorithm (Metropolis et al., 1953) pioneered MCMC, which was later generalised to the Metropolis-Hastings algorithm (Hastings, 1970). Later on, the Gibbs sampler (S. Geman and D. Geman, 1984) was introduced initially to find posterior modes rather than simulating the posterior distribution. This greatly improved the popularity of Bayesian methods as it was found that MCMC enabled many previously infeasible methods.

More formally, a sequence $X_1, X_2, \ldots$ of random variables is a Markov chain if the conditional distribution of X_n given $X_1, \ldots, X_{n-1}$ only depends on X_{n-1}. The set which contains the possible values of X_i is then the state space of the Markov chain. The transition probabilities are usually assumed to be stationary in that the conditional distribution of X_n given X_{n-1} is independent of n. The joint distribution of a Markov chain is then completely determined by the marginal distribution of X_1 (the initial distribution) and the conditional distribution of X_n given X_{n-1} (the transition probability distribution). Markov chains are a special case of a stochastic process, which is essentially a sequence of random variables.

The transition probability distributions must be constructed so that the Markov chain has a unique stationary distribution which is the target distribution. Firstly, the sequence can be shown to have a unique stationary distribution if the Markov chain is irreducible (it is possible to get to any state from any other state), aperiodic (each state may be returned to at irregular times) and not transient (each state will be returned to in finite time with probability 1). Secondly, the transition probabilities can be chosen to satisfy detailed balance with respect to the target distribution, so that the target distribution is the stationary distribution.

While, in Metropolis-Hastings, the full state vector may be updated in a transition, in Gibbs sampling only part of the state is updated. In Gibbs sampling, a special case of Metropolis-Hastings, the proposal for the next state in the Markov chain is from the conditional distribution, which is proportional to the target distribution. The conditional distribution used is of one component of the state given the rest of the components. A variant of this known as block Gibbs is to use the conditional distribution with several of the components omitted so that several variables are updated in one transition. Finally, collapsed Gibbs is when some of the components can be integrated out.

3.4.1 *Convergence checking and posterior summaries*

It can be unclear whether the MCMC chain has converged after a number of iterations. Informally, a common method is to look at traces or autocorrelation plots of summaries of important variables in the state of the chain, such as the size of the largest cluster or values of parameters. The number of iterations before the chain has appeared to visually converge is then often used as the burn-in, which is the number of iterations at the start of MCMC that are ignored since they do not come from the stationary distribution and are overly influenced by the initial distribution.

More formally, there have been many proposed convergence diagnostic techniques, with the two extremes being running a single chain for a long time (Geyer, 1992) and running several chains from different starting points (Gelman and Rubin, 1992). There is no consensus about which method is the best diagnostic, however, due to the growing ease of running multiple chains with modern computers, the multiple chain diagnostic is often used. The idea behind the single chain method is that by looking at autocorrelation within the chain, it can be estimated how long the chain takes to converge. The multiple chain method, on the other hand, compares the within-chain variance of a variable with the between-chain variance of the variable. If the between-chain variance is significantly greater than the within-chain variance then it is likely the case that the chains have not yet fully mixed and so have not yet converged.

The Gelman and Rubin diagnostics describe the calculation of potential scale reduction factors which collate the data from multiple Markov chains to estimate the between-chain and within-chain variance for various variables and use those to monitor convergence. Assuming there are m Markov chains each of length n after discarding the burn-in samples and the scalar estimand, such as the size of the largest cluster, for each

sample is denoted as $\psi_{ij}, i = 1, \ldots, m, j = 1 \ldots, n$ then the potential scale reduction factor can be calculated as

$$B = \frac{n}{m-1} \sum_{i=1}^{m} (\overline{\psi}_{i\cdot} - \overline{\psi}_{\cdot\cdot})^2 \tag{3.20}$$

$$\overline{\psi}_{i\cdot} = \frac{1}{n} \sum_{j=1}^{n} \psi_{ij} \tag{3.21}$$

$$\overline{\psi}_{\cdot\cdot} = \frac{1}{m} \sum_{i=1}^{m} \overline{\psi}_{i\cdot} \tag{3.22}$$

$$W = \frac{1}{m} \sum_{i=1}^{m} s_i^2 \tag{3.23}$$

$$s_i^2 = \frac{1}{n-1} \sum_{j=1}^{n} (\psi_{ij} - \psi_{i\cdot})^2 \tag{3.24}$$

where B and W are the between and within sequence variances respectively. The marginal posterior variance of the estimand, var(ψ), can be estimated by

$$\widehat{\text{var}}(\psi) = \frac{n-1}{n} W + \frac{1}{n} B. \tag{3.25}$$

The potential scale reduction is then an estimate of the factor which the scale of the distribution for ψ would be reduced in the limit of infinite simulations:

$$\hat{R} = \sqrt{\frac{\widehat{\text{var}}(\psi)}{W}}. \tag{3.26}$$

When $\hat{R}$ is near 1, this indicates that the chain has appeared to converge and no further simulations are needed. However, even if a simulation passes all convergence tests, it can still be far from convergence if important areas of the target distribution were not in the starting distribution and if it is hard to reach those areas with the transition probabilities.

Once the Markov chain has converged to the posterior distribution, the question of how to summarise the probability distribution arises. Commonly, summary statistics are taken across a final number of samples in the Markov chain. The more samples that are used, the more confidence can be given to the posterior distribution that has been estimated. Some algorithms advocate subsampling the chain to reduce autocorrelation, in which every kth state of the chain is used and all other states are removed. However, it has since been proved that throwing away this data does not improve the answer or accuracy of the approximation (MacEachern and Berliner, 1994). Thus, the only reason

for subsampling is to reduce the quantity of data that needs to be analysed for long running simulations. No subsampling is used for the results in this thesis.

Finally, identifiability is a common problem in mixture models involving MCMC due to the use of discrete hidden variables. This happens when the posterior distribution is invariant when the values (labels) of the discrete hidden variable are permuted. This results in non-identifiable labels. For example, in a Gaussian mixture model with two clearly defined mixture components A and B, label-switching can happen. This is because permuting the component assignments to B and A results in the same posterior probability. The mean location of the data points from a certain mixture component can then be equal to the mean of the whole data set. However, label-switching is not a problem if the quantities of interest are also invariant under label permutation. This is the case for the posterior summaries used later in the thesis. However, if they are not invariant, such as when examining a topic distribution across many samples, then comparing values across samples becomes a problem. There are several ways to deal with the problem including imposing constraints to break symmetry, building a sampler for which switching labels is unlikely and relabelling mixture components. To avoid this, I will examine topic distributions from one sample of the posterior.

3.5 TOPIC MODELS

The algorithms presented in Chapter 2 focus on matching entities, records or citations by scanning for similar strings. However, often two records that have several matching fields do not refer to the same entity. This is especially the case when integrating records from multiple domains, such as in combining citation databases from neuroscience and machine learning. In this case, different people may have identical names, concepts may be described using different terminology within the different domains and the same term may have different meanings in the different domains. When this kind of ambiguity exists, topic models can help by inferring the abstract themes that are described within the free text in the record. When two records have similar themes, then it can be inferred that some of their fields may refer to the same entity. An approach for inferring the themes among a collection of documents is topic modelling, where a set of latent topics can be thought to represent the abstract themes of the corpus.

Probabilistic topic models are based on the idea that documents are mixtures of latent topics, where a topic is a probability distribution over the vocabulary. Steyvers and T. Griffiths (2007) gave an overview of modern approaches. Topics have an advantage over traditional spatial representations, such as the vector space model (Salton, Wong and Yang, 1975), by being a generative model for documents. Since they represent the

data in a lower-dimensional form, they are related to traditional dimensionality reduction techniques such as PCA (Pearson, 1901). Topic models can also be viewed in a relational context as modelling differing classes of objects, which after extension to hierarchical topic models can include class inheritance.

Topic models make an exchangeability assumption: that the words in the documents and the documents themselves are unordered. This is weaker than the commonly used independent and identically distributed (i.i.d.) assumption.

David M. Blei, Jordan and A. Y. Ng (2003) introduced the most common topic model, the latent Dirichlet allocation (LDA) model, as a Bayesian extension to an existing probabilistic topic model, the probabilistic latent semantic indexing model (pLSI) (Hofmann, 2001), which was itself an extension of LSA (Deerwester et al., 1990). LDA uses a finite mixture generative model in which multinomial document-specific mixing proportions are drawn from a Dirichlet distribution. Each word in the document is an independent draw from this mixture model (proportions of topics). LDA takes the essential idea that a document is generated from a mixture of topics; which is modelled as the following process: a distribution over topics is chosen, for each word, a topic is chosen from that distribution and then a word is drawn from that topic. Inference techniques invert these steps to find the set of topics that generate the corpus. The LDA model has proved extremely versatile, and a variety of extensions have been proposed that extend it to modelling topic dynamics (David M Blei and Lafferty, 2006), relations (Chang and D. Blei, 2009) and n-grams (Wallach, 2006). Real-world applications where topic models have proved useful are for modelling and learning correlations of topics across multiple documents or corpora and for modelling the genetic haplotypes among human subpopulations (Xing et al., 2006).

LDA is a hidden variable model in that the observed data are the words of each document and the hidden variables indicate to which topic each of the words are allocated. The posterior distributions of the hidden variables are useful for tasks from information retrieval to document browsing. Since multiple topics are allocated to each document, this can be thought of as a mixed-membership model or admixture as opposed to traditional topic models where each document is limited to one topic. The generative process, a random process that can be thought of having produced the observed data, is as follows. Let K be the fixed number of topics, V the size of the vocabulary of the corpus, $\boldsymbol{\alpha}$ a positive K-dimensional vector and $\boldsymbol{\eta}$ a positive V-dimensional vector.

1. For each topic k,
    a) Draw a distribution over the vocabulary $\boldsymbol{\beta}_k \sim \text{Dirichlet}(\boldsymbol{\eta})$.
2. For each document i,

a) Draw a vector of topic proportions for the document $\boldsymbol{\theta}_i \sim \text{Dirichlet}(\boldsymbol{\alpha})$.

b) For each word j,

   i. Draw a topic for the word $z_{ij} \sim \text{Multinomial}(\boldsymbol{\theta}_i)$.

   ii. Draw the word $w_{ij} \sim \text{Multinomial}(\boldsymbol{\beta}_{z_{ij}})$.

The components of $\boldsymbol{\eta}$ are usually set to the same value, leading to a symmetric Dirichlet distribution. Wallach, Mimno and McCallum (2009) examined the effect of a fixed symmetric $\boldsymbol{\eta}$ compared with optimising $\boldsymbol{\eta}$ during the inference process and find that a symmetric $\boldsymbol{\eta}$ yields the best performance. $\boldsymbol{\alpha}$ is also usually set to a symmetric Dirichlet distribution as there is no prior reason to prefer any topics over any other. Optimisation strategies for $\boldsymbol{\alpha}$ are also explored in their paper. The posterior of a LDA model can be explored by analysing the distribution of each topic. $\boldsymbol{\beta}$ gives the probability of a word appearing in a topic and this allows a topic to be summarised by the highest probability words in a topic.

3.5.1 *Inference*

The hidden structure of the corpus is described by the hidden variables, the topics $\boldsymbol{\beta}$, the topic proportions for each document $\boldsymbol{\theta}$ and the topic allocations for each word $\mathbf{z}$. The inferred topics in the corpus and topic mixture for each document can be analysed in the posterior distribution.

The posterior distribution is as follows:

$$p(\boldsymbol{\theta}, \mathbf{z}, \boldsymbol{\beta}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\eta}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}, \boldsymbol{\beta}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\eta})}{\iiint p(\boldsymbol{\theta}, \mathbf{z}, \boldsymbol{\beta}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\eta}) \, \mathrm{d}\boldsymbol{\theta} \, \mathrm{d}\mathbf{z} \, \mathrm{d}\boldsymbol{\beta}} \tag{3.27}$$

The exact posterior is intractable due to the need to sum over all possible allocations of words to topics in the denominator. Gibbs sampling is commonly used for LDA inference as the sampling can easily be adapted to extensions of LDA and Gibbs sampling is easy to implement.

In LDA, since the prior distributions for $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ are Dirichlet distributions, they are conjugate to the multinomial distributions and so $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ can be integrated out allowing the joint distribution $p(\mathbf{w}, \mathbf{z}|\boldsymbol{\alpha}, \boldsymbol{\eta})$ to be calculated. These can be calculated by just keeping counts of which words are allocated to which topics. Assuming a symmetric $\boldsymbol{\eta}$, let η be the value of each component of $\boldsymbol{\eta}$. Assuming a symmetric $\boldsymbol{\alpha}$, let α be the value of each component of $\boldsymbol{\alpha}$.

$$p(\mathbf{w}|\mathbf{z}) = \left(\frac{\Gamma(V\eta)}{\Gamma(\eta)^V}\right)^K \prod_{k=1}^{K} \frac{\prod_w \Gamma(n_k^w + \eta)}{\Gamma(n_k^{\cdot} + V\eta)} \tag{3.28}$$

where w in n_k^w indexes into the vocabulary and n_k^w is the number of times word w has been assigned to topic k and $\cdot$ means summing over that index so that $n_k^{\cdot} = \sum_{v=1}^{V} n_k^v$.

$$p(\mathbf{z}) = \left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^D \prod_{i=1}^{D} \frac{\prod_k \Gamma(n_{ik} + \alpha)}{\Gamma(n_{i\cdot} + K\alpha)} \tag{3.29}$$

where n_{ik} is the number of words in document i that have been assigned to topic k and $n_{i\cdot} = \sum_{k=1}^{K} n_{ik}$. The posterior is then

$$p(\mathbf{z}|\mathbf{w}) = \frac{p(\mathbf{w}, \mathbf{z})}{\sum_z p(\mathbf{w}, \mathbf{z})}. \tag{3.30}$$

After cancelling some of the terms in the above equations, the full conditional distribution for each topic allocation variable needed for Gibbs sampling, $p(z_{ij}|\mathbf{z}^{-ij}, \mathbf{w})$, can be obtained:

$$p(z_{ij} = k|\mathbf{z}^{-ij}, \mathbf{w}) \propto \frac{n_k^{-ij,w_i} + \eta}{n_k^{-ij,\cdot} + V\eta} \frac{n_{ik}^{-ij} + \alpha}{n_{i\cdot}^{-ij} + K\alpha} \tag{3.31}$$

where the counts do not include the current allocation, z_{ij} as indicated by the $-ij$ superscripts. For sampling, the topic allocation variables are initialised randomly and the values of $\mathbf{z}$ sampled using the above equation for a number of iterations until the chain appears to have converged to the posterior. A number of samples from the posterior are then taken and $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ can be estimated from the samples.

$$\hat{\beta}_k^w = \frac{n_k^w + \eta}{n_k^{\cdot} + V\eta} \tag{3.32}$$

$$\hat{\theta}_{ik} = \frac{n_{ik} + \alpha}{n_{i\cdot} + K\alpha} \tag{3.33}$$

Once convergence has been reached, the model can be evaluated based on the perplexity of a held-out test set. Perplexity is equivalent to the inverse of the geometric

mean per-word likelihood so that a lower perplexity score indicates a higher likelihood of the words and better performance. This score is defined as

$$\exp\left(-\frac{\sum_{i=1}^{D}\log p(\mathbf{w}_i)}{\sum_{i=1}^{D} n_i}\right) \tag{3.34}$$

where n_i is the number of words in document i and $p(\mathbf{w}_i)$ can be approximated using $p(\mathbf{w}_i|\mathbf{w})$ with $\mathbf{z}$ being sampled from the posterior.

3.5.2 *Model selection*

LDA requires model selection to determine the number of topics, K, in the model. When K is too large then incoherent topics that consist of rare word co-occurrences can be learnt and when K is too small the topics may be too broad to be useful. One of the most popular ways of choosing K is to use cross validation to find the K that gives the lowest perplexity. This requires many runs of the model and so can be costly to perform. The hyperparameters α and η can be optimised during inference or set to commonly used hand-tuned values such as $\eta = 0.1$ and $\alpha = 50/K$ (Thomas L. Griffiths and Steyvers, 2004).

Bayesian nonparametric models present one solution to the problem of choosing K as discussed in Section 3.8.7.

3.6 AUTHOR-TOPIC MODELS

Author-topic models are a way to combine authorship information with topic models. The model of Rosen-Zvi et al. (2004) extends LDA by modelling authors. In their model, each author is associated with a multinomial distribution over topics. This allows authors to be compared using their distributions over topics. Each document is modelled as a mixture of the topic distributions that are associated with the authors of the document. In the generative process for the document: for each word, an author is drawn from the authors of the document, then a topic is chosen from the distribution over topics for that author, and finally a word is drawn from that topic. Compared to LDA, there is an additional latent variable for every word that indicates which author that word is associated with.

In the author-topic model, the true authors for each document are known and so the model cannot handle author ambiguity. There is also no model for author names so name variation cannot be handled. Their model instead predicts authors for unseen documents

where only the full text of the document is available. Their model also assumes the number of authors is fixed so that the model cannot learn to predict unseen authors. Since in their model, every word must be associated with an author, the predictive distribution for authors overweights authors who have authored many long documents compared with authors who have authored few or many short documents. Authors who are assigned to few words or who have authored few documents will also have a broad distribution over topics making them hard to distinguish from other authors.

3.7 NONPARAMETRIC MODELS

Nonparametric statistics is an approach that makes as few assumptions as possible partly by using infinite-dimensional models. One of the main techniques in frequentist nonparametrics is the use of statistical tests that do not assume a specific distribution. A simple example of a nonparametric density estimation approach is a histogram with fixed-size buckets. Since there are fewer assumptions, nonparametric methods can be used on a wider range of problems than traditional parametric methods and allow for robust models. On the other hand, using more robust techniques sacrifices statistical power, and a larger dataset can be needed to achieve the same confidence levels that a parametric approach gives. Bayesian nonparametrics are commonly used for problems where no fixed model is assumed for the data. One way of thinking about this is the need for a model where the structure can change and adapt to the data and can work in large parameter spaces. As a result, Bayesian nonparametric models can be more difficult to work with than the models just described. Computation is also more difficult as probability distributions on potentially infinite-dimensional spaces need to be manipulated. These models can also be more difficult to set up, however, a major benefit is that parts of the model structure no longer need to be fixed a priori. Bayesian nonparametric models are rapidly attracting interest, due to the increasing quantity of data in problems and advances in computation.

Recently, the Dirichlet process (DP) and Pitman-Yor process are growing in popularity among Bayesian nonparametrics. The DP mixture model (Antoniak, 1974) is one of the most popular models, which has partly been due to the ease of posterior computation. These models are attractive since they are not restricted to a finite number of latent clusters or features and so offer significantly extra modelling flexibility with infinite mixture models. Posterior computation remains tractable since only a finite number of clusters is needed to represent a finite quantity of data. These types of models have been successfully used in density estimation, computational biology, computer vision and natural language processing (Teh and Jordan, 2009).

3.8 DIRICHLET PROCESSES

A Dirichlet process (DP) (Antoniak, 1974; Ferguson, 1973) is a stochastic process that can be thought of as a probability distribution on the space of probability measures. The name of the process accurately describes that the DP results in finite-dimensional Dirichlet marginal distributions, similar to the Gaussian process that has Gaussian distributed finite-dimensional marginal distributions. DPs are commonly used as a prior on the space of probability measures, which give wider support and so improved flexibility over using traditional parametric families as priors. In addition, DPs have tractable posteriors so making them important in Bayesian nonparametric problems. The term nonparametric can be interpreted as meaning that the number of model parameters may grow indefinitely with the sample size. This contrasts with parametric models where the number of parameters are independent of sample size.

3.8.1 *Formal definition*

Ferguson (1973) initially introduced the idea of a Dirichlet process to exploit the conjugacy of the process for grouped data. A Dirichlet process (DP) is a distribution over measures. With a DP, there is a positive probability of drawing a previously drawn value, and thus the draws are discrete with probability 1. If the draws were continuous, then there would be zero probability of exactly drawing a previously drawn value. The property that these draws are discrete is very useful for clustering in DP mixtures. The DP is a natural generalisation of Dirichlet distributions to infinite dimensions.

Let Q be a finite measure on a measurable space $\mathcal{Y}$. A random measure, P, on $\mathcal{Y}$ follows a Dirichlet process if for every finite measurable partition $(B_1, \ldots, B_k)$ of $\mathcal{Y}$, the distribution of $(P(B_1), \ldots, P(B_k))$ is a Dirichlet distribution with parameters $Q(B_1), \ldots, Q(B_k)$. Q can be decomposed into $G_0(A) = Q(A)/\alpha$ and $\alpha = Q(\mathbb{R})$ so that $Q = \alpha G_0$. G_0 is a probability function and is known as the base measure of the Dirichlet process as it gives the expectation of P. α is a positive scalar known as the concentration or precision parameter. We then write that $P \sim \text{DP}(\alpha, G_0)$. If $X_1, \ldots, X_n$ is a sample from P then the posterior distribution of P given $X_1, \ldots, X_n$ is also a Dirichlet process but with parameter $Q + \sum_{i=1}^{n} \delta_{X_i}$, where δ_x denotes a measure with the point x having mass one. This property allows the posterior to be relatively easy to calculate, but as a result, there is no smoothing in the posterior.

Ferguson also offered an alternative definition of the DP in terms of the gamma process and used it to show that the Dirichlet process is a discrete probability measure.

3.8.2 *Stick-breaking construction*

The stick-breaking representation described by Sethuraman (1994) is another representation of the Dirichlet process. This representation gives a method to directly sample from the DP and shows that draws from the DP are composed of a weighted sum of point masses. The point masses can be referred to as atom locations and the weights as stick lengths.

$$\beta'_k \sim \text{Beta}(1, \alpha) \tag{3.35}$$

$$\beta_k = \beta'_k \prod_{q=1}^{k-1}(1 - \beta'_q), \qquad \text{for } k = 1, \dots, \infty \tag{3.36}$$

$$\theta_k \sim G_0 \tag{3.37}$$

$$\boldsymbol{\pi}|G_0 \sim \text{DP}(\alpha, G_0) = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k} \tag{3.38}$$

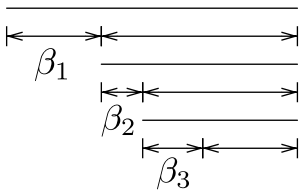


Figure 3.1: The stick-breaking weights where $\beta_1, \beta_2, \dots$ are the non-overlapping lengths of pieces gradually broken from a stick of unit length.

The distribution on the sequence $\{\beta_1, \dots\}$ is sometimes referred to as the GEM(α) distribution where GEM stands for Griffiths, Engen and McCloskey.

The stick-breaking construction is based on independent sequences of random variables $(\beta_k)_{k=1}^{\infty}$ where $\sum_{k=1}^{\infty} \beta_k = 1$ almost surely. These weights exponentially tend towards zero so only a small number of clusters are needed to model a finite dataset. An example of the distribution of the stick weights is shown in Figure 3.1. This construction also shows that draws from the DP are discrete as it puts weight β_k on the atoms δ_{θ_k} where θ_k is distributed according to the base distribution G_0.

3.8.3 *DP mixture models*

The Dirichlet process mixture (DPM) (Radford M. Neal, 2000) is very useful in Bayesian density estimation and comes about when parametric families are mixed nonparametrically. It is often used in clustering to model a countably infinite number of mix-

ture components. Finite mixture models usually require finding the number of clusters through model averaging or model selection whereas infinite mixture models makes this work unnecessary. DPMs were originally motivated as a method of smoothing a DP by convolving with parametric models. In a DPM model, the n observations x_i are modelled by a parametric family with latent parameters $\chi(x_i|\theta_i), i = 1, \ldots, n$. These observations are required to be exchangeable so that de Finetti's theorem applies. The theorem states that for any infinitely exchangeable sequence, there exists a probability distribution F where there is an underlying parameter and the observations are conditionally independent given that parameter. The marginal density for a x_i is then $f(x_i) = \int \chi(x_i|\theta_i)\,\mathrm{d}G(\theta_i)$. Since G is unknown, a Dirichlet process prior can be used as a prior for G especially since it has wider support than parametric family priors. The prior for the density f is then known as the Dirichlet process mixture. Assuming the DP has a concentration parameter α and a base distribution G_0 then:

$$G|G_0 \sim \mathrm{DP}(\alpha, G_0) \tag{3.39}$$

$$\theta_i|G \sim G \qquad i = 1, 2, \ldots, n \tag{3.40}$$

$$x_i|\theta_i \sim F(\theta_i) \tag{3.41}$$

Since G is discrete, there is a non-zero chance that more than one θ_i shares the same value. This can be viewed as a mixture model where the x_i that share the same θ_i are in the same cluster.

From a mixture modelling perspective and the stick-breaking construction, (3.39) can be written in terms of a cluster assignment or indicator variable z_i. Taking $\boldsymbol{\pi}$ to be a probability distribution over the positive integers, then z_i takes the value k with a probability π_k. Since the θ_i are exchangeable, the cluster indicators z_i can be reordered.

$$\pi'_k \sim \mathrm{Beta}(1, \alpha) \tag{3.42}$$

$$\pi_k = \pi'_k \prod_{q=1}^{k-1}(1 - \pi'_q) \qquad \text{for } k = 1, \ldots, \infty \tag{3.43}$$

$$z_i|\boldsymbol{\pi} \sim \boldsymbol{\pi} \qquad i = 1, 2, \ldots, n \tag{3.44}$$

$$\boldsymbol{\phi}_k|G_0 \sim G_0 \tag{3.45}$$

$$x_i|z_i, \boldsymbol{\phi} \sim F(\boldsymbol{\phi}_{z_i}) \tag{3.46}$$

where $G = \sum_{k=1}^{\infty} \pi_k \delta_{\boldsymbol{\phi}_k}$ and $\theta_i = \boldsymbol{\phi}_{z_i}$. $\boldsymbol{\phi}_k$ are the parameters that determine the distribution of observations in cluster k, the x_i are drawn from a mixture of distributions $F(\cdot)$ and G_0 is the base distribution for the cluster parameters. Since the mixing pro-

portions drop exponentially quickly, a finite number of clusters is used to model the observations.

The DP has been extended to new distributions as well as new classes of nonparametric objects including the Pitman-Yor process (Pitman and Yor, 1997) and the hierarchical Dirichlet process (HDP). The DP has proved very useful in modelling observations with an unknown number of cluster components. The dependent Dirichlet process (DDP) (MacEachern, 1999) extends the Dirichlet process to cover multiple dependent distributions, either in the cluster locations or the cluster weights. This can be used for spatial and dynamic processes.

3.8.4 *Choosing concentration parameters*

Setting the value of the concentration parameter α for a Dirichlet process $\mathrm{DP}(\alpha, G)$ can be a difficult problem. A priori, the number of clusters represented in a dataset of a given size is dependent on α. In general for a sequence of Dirichlet processes, as $\alpha \to \infty$, the sequence converges to the base distribution G. On the other hand, as $\alpha \to 0$, the limit is a process that places all the mass on one single point sampled from the prior G. In the mixture model, there will be many clusters as $\alpha \to \infty$ to cover G and few clusters as $\alpha \to 0$ since the mass will be concentrated on a few points sampled from G. The prior distribution of the number of clusters K given α and a sample size n is defined (Escobar and West, 1995):

$$p(K|\alpha, n) = c_n(K) n!\, \alpha^K \frac{\Gamma(\alpha)}{\Gamma(\alpha + n)} \tag{3.47}$$

where $c_n(K)$ are the absolute values of Stirling numbers of the first kind and satisfy: $c_{n+1}(K) = n c_n(K) + c_n(K-1)$ where $c_n(n) = 1$. For large n, $p(K|\alpha, n)$ reduces to:

$$\mathrm{E}(K|\alpha, n) \approx \alpha \ln(1 + \frac{n}{\alpha}) \tag{3.48}$$

The formula (3.47) allows the conditional posterior distribution of α given the sampled number of clusters K to be calculated, so α can be updated by sampling from this posterior. When the prior for α is a mixture of gamma distributions, it becomes possible to sample from the exact posterior of α. When a conjugate prior is not used for α, slice sampling (Radford M Neal, 2000) can be used to sample from its posterior.

Assuming that $p(\alpha) = \mathrm{Gamma}(\alpha|a, b)$ with parameters shape $a > 0$ and rate $b > 0$ then:

$$p(K|a, b, n) = \int_0^\infty p(K|\alpha, n) p(\alpha|a, b)\, \mathrm{d}\alpha \tag{3.49}$$

and from (Escobar and West, 1995), the posterior for α is a mixture of two gamma densities:

$$\alpha|\eta, k \sim \pi_\eta \,\text{Gamma}(a+k, b-\log(\eta)) + (1-\pi_\eta)\,\text{Gamma}(a+k-1, b-\log(\eta)) \tag{3.50}$$

where π_η is defined by the ratio

$$\frac{\pi_\eta}{1-\pi_\eta} = \frac{a+k-1}{n(b-\log(\eta))} \tag{3.51}$$

and η is sampled from

$$\eta|\alpha, K \sim \text{Beta}(\alpha+1, n) \tag{3.52}$$

A non-informative prior for α is obtained when $a = 0$ and $b = 0$. This prior puts an almost uniform probability density over a wide range of values for α. However, with this prior $p(K = 1|a, b, n) \to 1$, so that a priori, there are very few clusters. Other typical priors for α such as Gamma(3.5, 0.5) also result in much of the probability mass being placed on a low number of clusters. Dorazio (2009) suggested an alternative approach where the prior for α is induced by a uniform prior on K. This is found by minimising the Kullback-Leibler (KL) divergence between a uniform distribution for K and $p(K|a, b, n)$.

$$-\log n - \frac{1}{n}\sum_{k=1}^{n} \log(p(K|a, b, n)) \tag{3.53}$$

Since $p(K|a, b, n)$ cannot be expressed in closed form, Dorazio (2009) uses numerical quadrature to calculate (3.49), though this is very slow for large n. Instead, I use Monte Carlo integration to calculate (3.49) and use (3.53) to find a and b. This approach is used to choose the prior for α in the remainder of the thesis.

3.8.5 *Chinese restaurant process*

The Chinese restaurant process (CRP) is the process that results from integrating out the random measure in the Dirichlet process. This process is often referred to when using MCMC sampling for DPs as it makes posterior inference tractable. The process can be thought of as follows. A restaurant has an infinite number of round tables in it, each of which can hold one dish as well as an infinite number of customers. There is an infinite queue of customers from which each customer comes in one-by-one into the

restaurant and chooses to sit at a table. The customer can choose to sit at an already unoccupied table or the next unoccupied table. The probability of a customer sitting at an occupied table k is $n_k/(n+\alpha)$ where n_k is the number of customers already sitting at that table, n is the number of customers already sitting in the restaurant and α is the concentration parameter of the CRP. The probability of the customer sitting at an empty table is $\alpha/(n+\alpha)$. This results in an exchangeable distribution over the tables that each customer chooses to sit at. In addition, the probability of a specific customer sitting at a particular table is the same no matter what position it was in the queue. During posterior inference when inferring which table a particular customer is sitting at, the customer can be treated as if he was the last customer to come into the restaurant so significantly simplifying the problem. Since customers prefer to sit at tables with larger numbers of existing customers, this behaviour is also known as the rich gets richer.

3.8.6 *Inference for Dirichlet processes*

Gibbs sampling via Markov chain Monte Carlo (MCMC) is the most common method for posterior inference for Dirichlet processes. Recently additional methods have also been proposed such as variational mean field inference, slice sampling and retrospective sampling. However, MCMC remains the simplest to implement and is relatively easy to adapt to extensions to the DP model. Gibbs sampling involves sampling from the conditional distributions of variables, which is often easier to sample from than the joint distribution. Since samples from a Dirichlet process are exchangeable, the conditional distributions of each variable are relatively easy to sample from.

Conjugate models

Inference is significantly simpler in the DP when G_0 is the conjugate prior for the likelihood F. In this case, the parameters for each cluster can be integrated out, and only the allocations of observations to clusters need to be recorded. Assuming the state of the Markov chain solely consists of the cluster indicator variables, $z_1, \ldots, z_n$ for n observations, $x_1, \ldots, x_n$, then Gibbs sampling occurs by the following.

For $i = 1, \ldots, n$, draw a new value for $z_i | z^{-i}, x_i$ from

$$P(z_i = k | z^{-i}, x_i) = \begin{cases} b\dfrac{n_k^{-i}}{n-1+\alpha} \displaystyle\int F(x_i, \phi)\, \mathrm{d}H_k^{-i}(\boldsymbol{\phi}) & \text{if } k = z_j \text{ for some } j \neq i, \\ b\dfrac{\alpha}{n-1+\alpha} \displaystyle\int F(x_i, \phi)\, \mathrm{d}G_0(\boldsymbol{\phi}) & \text{otherwise.} \end{cases} \tag{3.54}$$

where the superscript $-i$ indicates the variable with the contribution from z_i removed, n_k^{-i} is the number of $z_j = k$ where $j \neq i$, b is a normalising constant and H_k^{-i} is the posterior distribution of $\boldsymbol{\phi}$ based on the prior G_0 and all observations x_j where $j \neq i$ and $z_j = k$.

When F is a M-dimensional multinomial distribution and G_0 is a symmetric Dirichlet(β), $\boldsymbol{\phi}$ can be integrated out as follows:

$$\int F(\mathbf{x}|\boldsymbol{\phi})\,\mathrm{d}G_0(\boldsymbol{\phi}|\boldsymbol{\beta}) = \frac{1}{\mathrm{B}(\boldsymbol{\beta})}\int \prod_{i=1}^{M} \phi_i^{x_i}\phi_i^{\beta_i-1}\,\mathrm{d}\boldsymbol{\phi} \tag{3.55}$$

$$= \frac{1}{\mathrm{B}(\boldsymbol{\beta})}\int \prod_{i=1}^{M} \phi_i^{x_i+\beta_i-1}\,\mathrm{d}\boldsymbol{\phi} \tag{3.56}$$

Since the integrand is a Dirichlet density with parameters $\mathbf{x} + \boldsymbol{\beta}$ then taking into consideration the normalising factor for a Dirichlet density:

$$\int \prod_{i=1}^{M} \phi_i^{x_i+\beta_i-1}\,\mathrm{d}\boldsymbol{\phi} = \mathrm{B}(\mathbf{x}+\boldsymbol{\beta}) \tag{3.57}$$

$$\text{so} \int F(\mathbf{x}|\boldsymbol{\phi})\,\mathrm{d}G_0(\boldsymbol{\phi}|\boldsymbol{\beta}) = \frac{\mathrm{B}(\mathbf{x}+\boldsymbol{\beta})}{\mathrm{B}(\boldsymbol{\beta})} \tag{3.58}$$

$$= \frac{\prod_{i=1}^{M}\Gamma(x_i+\beta_i)}{\Gamma(\sum_{i=1}^{M} x_i+\beta_i)}\frac{\Gamma(\sum_{i=1}^{M}\beta_i)}{\prod_{i=1}^{M}\Gamma(\beta_i)} \tag{3.59}$$

where B is the multinomial beta function defined below. The above density is also known as the Dirichlet-multinomial compound distribution.

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{M}\Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{M}\alpha_i)} \tag{3.60}$$

where Γ is the gamma function, which is an extension of the factorial function and defined as $\Gamma(z) = \int_0^\infty e^{-t}t^{z-1}\,\mathrm{d}t$ for positive z.

Non-conjugate models

Sometimes a conjugate prior might not exist for a certain likelihood function and so the parameters $\boldsymbol{\phi}$ cannot be integrated out. There are several methods for sampling from a non-conjugate DP model, but the one that is used most frequently is algorithm 8 described by Radford M. Neal (2000). This is an auxiliary-variable sampling method that samples $\boldsymbol{\phi}$ as required and proposes new clusters based on samples of $\boldsymbol{\phi}_k$ from the

prior G_0. Singleton clusters, which are clusters that only have one observation allocated to them, are treated specially in this algorithm. The state of the Markov chain now consists of both the cluster indicators $z_1, \dots, z_n$ and the cluster parameters $\phi_1, \dots, \phi_K$ where $K = \max(z_1, \dots, z_n)$. Let m be the number of auxiliary variables to use.

1. For $i = 1, \dots, n$, let k^- be the number of distinct z_j for $j \neq i$, and let $h = k^- + m$. These z_j are labelled between 1 and k^-. If $z_i = z_j$ for some $j \neq i$, i.e. it is not a singleton cluster, then draw values independently from G_0 for the ϕ_k for which $k^- < k \leq h$. If $z_i \neq z_j$ for all $j \neq i$, i.e. it is a singleton cluster, then set z_i to $k^- + 1$ and draw values independently from G_0 for the ϕ_k for which $k^- + 1 < c \leq h$. Then draw a new value for z_i from $1, \dots, h$ according to

$$P(z_i = k | z^{-i}, x_i, \phi_1, \dots, \phi_h) = \begin{cases} b \dfrac{n_k^{-i}}{n - 1 + \alpha} F(x_i, \phi_k) & \text{for } 1 \leq k \leq k^- \\ b \dfrac{\alpha/m}{n - 1 + \alpha} F(x_i, \phi_k) & \text{for } k^- < k \leq h \end{cases} \tag{3.61}$$

   where n_k^{-i} is the number of $z_j = k$ where $j \neq i$ and b is a normalising constant. The ϕ_k that are not associated with any observations can be removed from the state.

2. For $k = 1, \dots, K$ where $K = \max(z_1, \dots, z_n)$, draw a new value from $\phi_k | x_i$ such that $z_i = k$ or perform some other update to ϕ_k that leaves this distribution invariant.

For ease of implementation, my experiments use $m = 1$.

3.8.7 *Hierarchical DPs*

The hierarchical Dirichlet process (HDP) (Teh, Jordan et al., 2006) is a hierarchical extension to DPs. The hierarchical structure provides an elegant way of sharing parameters and atoms. This process defines a set of probability measures G_i for D pre-specified groups of data and a global probability measure G_0. The global measure is distributed as

$$G_0 | \gamma, H \sim \text{DP}(\gamma, H) \tag{3.62}$$

where H is the base probability measure and γ is the concentration parameter.

The random measures for each group are conditionally independent given the global measure

$$G_i | \alpha_0, G_0 \sim \text{DP}(\alpha_0, G_0) \tag{3.63}$$

where α_0 is a concentration parameter. The distribution G_0 varies around H by an amount controlled by γ and the distribution G_i in the group i varies around G_0 by an amount controlled by α_0. This can also be seen as adding another level of smoothing on top of DPM models and allowing atoms to be shared across groups. Let $\theta_{i1}, \theta_{i2}, \ldots$ be i.i.d. variables distributed to G_i and each of these variables is a parameter that corresponds to an observation x_{ij}, the likelihood of these observations being

$$\theta_{ij}|G_i \sim G_i \tag{3.64}$$

$$x_{ij}|\theta_{ij} \sim F(\theta_{ij}) \tag{3.65}$$

where $F(\theta_{ij})$ is the distribution of x_{ij} given θ_{ij}. This prior results in a Dirichlet process (DP) being associated with each group in the model where the DPs are conditionally independent given their parent and the atoms drawn in the parent node are shared among the descendant groups. This structure can also be extended to multiple levels. A diagram of atoms being shared among groups of data is given in Figure 3.2.

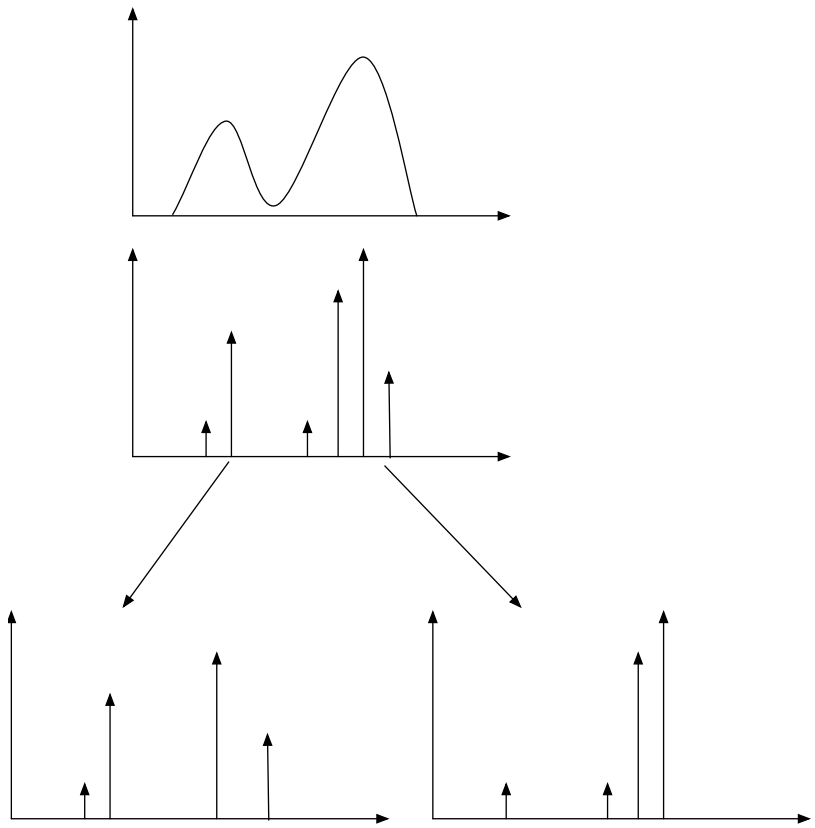


Figure 3.2: The random measures from a HDP. The base measure is continuous at the top level, the middle level is the global random measure that has all the atoms and the atoms are shared between the two resulting group-level random measures at the bottom, though each of them puts different weights on the atoms. The x-axis represents atom locations and the y-axis represents atom weights.

The HDP requires the data to be in a pre-defined nested structure and is unable to discover this structure automatically in unstructured data. The model has been used in information retrieval tasks and used in relation with traditional TF-IDF measures (Cowans, 2004) for measuring the score of documents in relation to a query. A variation of HDP uses pachinko allocation to model topics for documents where there is no predefined hierarchical structure (W. Li, D. Blei and McCallum, 2007).

Similarity to LDA

With the appropriate base measure, the HDP can be thought of as the infinite analogue of LDA. In the HDP, the base probability measure allows for a countably infinite number of multinomial draws and so an infinite number of topics. This allows the number of topics to grow or shrink according to the data. This solves the problem of using cross-validation to find the best number of topics in LDA and reduces the problems of overfitting or underfitting due to a fixed number of topics. One of the main differences between HDP and LDA is the posterior probability of a topic in any document. HDP is similar to a version of LDA where the probability of a topic in a document $\boldsymbol{\theta}$ is optimised during sampling. As a result, the probability of a topic appearing in a document is related to the number of times that topic has appeared in other documents. However, typically in LDA, θ is not optimised and instead drawn from a symmetric Dirichlet distribution so that all topics have equal probability of appearing in a document. Wallach, Mimno and McCallum (2009) recommended that LDA is used with an optimised $\boldsymbol{\theta}$ as it encourages function words (words which are not helpful in describing the topics) to be grouped together in one topic rather than spread across multiple topics.

Sampling concentration parameters

Assuming that $p(\alpha_0) = \text{Gamma}(\alpha_0|a, b)$ with parameters shape $a > 0$ and rate $b > 0$, α_0 can be sampled by the following method as described in (Teh, Jordan et al., 2006). Let n_{it} be the number of observations allocated to table t in document i. For D documents and each document i, we define two auxiliary variables w_i and s_i where w_i takes values in $[0, 1]$ and s_i is a binary variable:

$$\frac{\alpha_0}{\alpha_0 + n_{i\cdot}} = \frac{1}{\Gamma(n_{i\cdot})} \int_0^1 w_i^{\alpha_0} (1 - w_i)^{n_{i\cdot}-1} \left(1 + \frac{n_{i\cdot}}{\alpha_0}\right) \mathrm{d}w_i \tag{3.66}$$

where $n_{i\cdot}$ is the number of observations in document i.

The following distribution is then defined:

$$q(\alpha_0, \mathbf{w}, \mathbf{s}) \propto \alpha_0^{a-1+m_{\cdot\cdot}} \mathrm{e}^{-\alpha_0 b} \prod_{i=1}^{D} w_i^{\alpha_0} (1 - w_i)^{n_{i\cdot}-1} \left(\frac{n_{i\cdot}}{\alpha_0} \right)^{s_i} \tag{3.67}$$

Marginalising out α_0 and then dividing gives the conditional distribution for α_0 as:

$$q(\alpha_0 | \mathbf{w}, \mathbf{s}) \propto \alpha_0^{a-1+m_{\cdot\cdot}-\sum_{i=1}^{D} s_i} \mathrm{e}^{-\alpha_0 (b - \sum_{i=1}^{D} \log w_i)} \tag{3.68}$$

Given α_0, the auxiliary variables are conditionally independent, so they are sampled from the distributions:

$$q(w_i | \alpha_0) \propto w_i^{\alpha_0} (1 - w_i)^{n_{i\cdot}-1} \tag{3.69}$$

$$q(s_i | \alpha_0) \propto \left(\frac{n_{i\cdot}}{\alpha_0} \right)^{s_i}. \tag{3.70}$$

Sampling each of these variables in turn typically mixes within 20 iterations.

4

THE AUTHOR-TOPIC SPACE MODEL FOR DISAMBIGUATION

In Chapter 2, I described the entity resolution problem and its related sub-problems. In Chapter 3, I introduced a number of Bayesian nonparametric and topic models. The framework laid out in this chapter utilises the previous models to build a general approach for tackling identity resolution and author disambiguation. In addition, the approach incorporates information available from the text in documents. The framework uses a generative model of both the text of a document and its list of authors to resolve identities in a corpus of documents. This approach is an unsupervised method, making no assumption about the true number of identities in the corpus. Thus, a separate training set with information about the true authors is not needed.

Author names in papers can often contain transcription or OCR errors, especially when scanned from physical documents. This means that the documents written by an author cannot typically be found by an exact string search on the author name. Another problem is that common names may be ambiguous, and further analysis of the document text is needed to investigate if the documents are by the same author. My joint author-topic space model for disambiguation takes this into account. By using a nonparametric model, no assumption regarding the total number of *author entities* needs to be made. I use each document's abstract to infer the set of topics that compose the paper. The association of these topics with the authors from a name variant model improves the inference of the author entities. Each latent topic is a distribution over the vocabulary as described in Section 3.5 and can be thought of an abstract theme. Compared with parametric topic models, my model makes no assumption regarding the total number of *topics*. My method simultaneously infers the disambiguated author entities and the topics upon which the authors write. The performance of two Gibbs sampling algorithms for inference is tested, and the model is evaluated on real world datasets.

The remainder of this chapter is set out as follows. In Section 4.1, I present a brief background to the problem and how the model developed in this chapter will tackle it. In Section 4.2, I describe a high-level overview of the model, which is then developed fully in Section 4.3. Section 4.5 describes two sampling algorithms that can be used to infer the posterior distributions and quantities in Section 4.4. I then compare my model with related models in Section 4.6 and describe experiments on real world datasets in

Section 4.8. Finally, I conclude in Section 4.9 with a discussion of the benefits and disadvantages of the author-topic space model for disambiguation.

4.1 INTRODUCTION

The task of coreference resolution, also known as record linkage, de-duplication or entity resolution is a difficult and important step that is necessary for pre-processing data before any large-scale data mining. This is important in the area of data integration where records from disparate sources, such as a health care system and a social security number system, need to be merged together. The problem in regard to academic papers is particularly relevant for publication libraries such as Google Scholar, Citeseer and DBLP. Digital Object Identifiers (DOIs), ISBNs and other global IDs all attempt to tackle this problem, but none of them are in wide use. According to D. Lee et al. (2007), the popular book 'Artificial Intelligence: A modern approach' has 23 unique records in CiteSeer, with no direct links between them.

Transcription errors are very common in author disambiguation especially as names cannot normally be corrected by dictionaries or spell checkers. In this chapter, an *author identity* refers to a real-world individual and an *author entity* is a latent representation of the author identity in the model. An *author name* refers to one of the multiple references or names by which the individual is known. One of the most frequent and difficult errors in foreign names originates from their transliteration. There are multiple methods for transliterating names in a foreign script into the Latin alphabet. For example, the most common Chinese last name, *Li*, can be transliterated into more than seven other forms including *Lee*, *Lý*, *Ri* and *Lei*. An author's first name, though informative, is also often abbreviated in citations, and the middle name may not appear or may even be a substitute for the first name. An author's name may also be an uncommon spelling of a common name or different authors may share the same name. For documents where no electronic copy exists, OCR errors typically occur frequently. In contrast to typographic errors, OCR errors have different characteristics since words may be broken, e.g. error → err r, single characters may be recognised as multiple characters, e.g. m → iii, multiple characters may be recognised as single characters, single glyphs from ligatures such as fi may be unrecognised and so on. These errors are propagated to other documents by authors unwittingly citing names that contain errors.

Author disambiguation is often tackled by generating pairwise distance scores, often between author names, and clustering on those scores, with a wide variety of techniques including latent Dirichlet allocation (LDA)(Daumé III and Marcu, 2005; Torvik et al., 2005; Bhattacharya and Getoor, 2006; Culotta et al., 2007; Kanani, McCallum and Pal,

2007) and others that are described in Chapter 2. These models are usually not generative models and so cannot make use of unlabelled data, such as a set of documents from one of the authors. These approaches are also limited by not modelling dependencies between names and other information in the paper such as its title and abstract, which may be strong indicators of whether two author names refer to the same person by writing on the same topic. Hall, Sutton and McCallum (2008) tackles this kind of cross-field dependency for the problem of venue disambiguation. But each document usually only has one venue of publication whereas there are often multiple authors for a document. Models which take account of multiple fields and side information are usually more successful than those that disambiguate solely on name.

In the development of the model, I take a Bayesian nonparametric modelling approach, based on the hierarchical Dirichlet process (HDP) (Teh, Jordan et al., 2006). The HDP mixture model allows data from different groups to be clustered together, as described in Section 3.8.7. This model shares clusters among multiple Dirichlet processes (DPs) and provides advantages over parametric methods such as LDA. In the scenarios considered here, the number of entities (distinct authors) is unknown, and a parametric approach based on a topic model would require this and the number of topics to both be estimated, which can be a lengthy process. The number of entities is unknown due to the errors described earlier, and the number of topics is unknown since it is unknown how many different fields of expertise a corpus might cover. The number of author entities in the corpus is expected to grow slowly since an author is likely to write multiple documents in a corpus. Thus, DP mixtures as described in Section 3.8 and related mixture models are an appropriate model, as they have the two attractive properties that the number of clusters in the model does not need to be known in advance and the number of clusters grows logarithmically fast with the data so inducing sparsity.

The topics and authors are modelled collectively with a single set of latent variables using the pre-defined documents as groups. The posterior cluster assignments of names to entities are used to identify author entities together with the topics about which they write. This means that each entity has its own topic or distribution over words. The author-topic model (Steyvers, Smyth et al., 2004) also models both topics and sets of authors who write about those topics given that the authors are known. However, their author-topic model is unable to model authors that have the same name or authors that have different names or aliases. I present a novel model for the problem of author disambiguation that takes account of these problems in a nonparametric generative approach.

4.2 OUTLINE OF THE MODEL

Though approaches based on disambiguation with the aid of the author's institutional affiliation or email addresses have been explored, it is rare that these pieces of information are available for all the authors in a document. On the other hand, the title and abstract of a document are the basic pieces of a document that are almost always available. Thus, my approach only requires this basic set of available information.

The model in Section 4.3 is based on the idea of identifying author entities from a combination of their name and the topics about which they write. I assume that the dataset is a corpus of documents, where for each document i, the words in the text or abstract w_{ij} are available where j indexes into the words. I also assume a list of author names a_{ij} is available, where j indexes into the author fields. Each word can be represented as a multinomial using one-of-V encoding where V is the vocabulary size of the corpus, so a word v would be represented as a vector with $w_{ijk} = 1$ for $k = v$ and $w_{ijk} = 0$ for $k \neq v$. Each author name can be a set of character n-grams with a bag of words assumption represented by a multinomial vector where a_{ijk} is the number of times the n-gram k appears in the author name. The name can also be a set of character n-grams that assumes a character level language model so that $a_{ijm|n}$ is the number of times the character m appeared after n in the author name. The author names are aliases of the underlying author entity or identity. The text for the document may be composed of a number of different themes or topics.

I assume a form of generative process where the distribution of topics and authors for the corpus of documents is first jointly drawn from a DP prior. For each document, a mixture of topics and authors is drawn. Each word in the abstract is generated by drawing a topic and then a word given the topic. Each author in the author list is generated by drawing an author and then a sequence of character n-grams for the name for that author. The base distribution for the topics is over the vocabulary of words used in the corpus and for the authors it is over the n-grams used in author names. Prior assumptions on the variance of the author names and topics can be made through the DP hyperparameters. By using infinite mixtures of topics and authors, my model makes no prior assumptions regarding how many entities or different topics exist, which is useful as most of the time the true number of authors in a corpus is unknown. I evaluate the model by inferring the author identities for the real-world citation database CiteSeer and on a conflated version of that dataset. I compare against different name variation models, inference algorithms and a basic baseline.

A n-gram is a sub-sequence of strings of length n from a sequence. The set of bigrams at the character level for a string consists of all the 2 character sequences in the string

as well as bigrams to cover the beginning and end of string markers. Similarly, trigrams are the 3 character sequences. I explore a model that uses the bag of words assumption (BOW), a simplifying assumption that the n-grams are independent. I also extend the generative bigram model (Wallach, 2006) to a nonparametric setting. These n-gram based approaches can model author entities that have name variants or last names which have been corrupted or split into multiple words. The model of author names with BOW trigrams assumes a bag-of-trigrams model where the ordering of the trigrams is ignored and the parameters are assumed to be exchangeable. For example, the probability of the name *John* would be calculated from marginal trigram frequencies as

$$p(\text{John}) \propto p(\text{<s><s>J})p(\text{<s> Jo})p(\text{Joh})p(\text{ohn})p(\text{hn</s>}) \tag{4.1}$$

where $p(abc)$ is the probability of the sequence of characters 'abc' appearing in an author name, <s> is the start of name marker, </s> is the end of name marker and the trigrams are assumed to be independent. The independence assumption is clearly false since the string has been divided into overlapping substrings. However, some existing string similarity techniques (Christen, 2006) make this assumption when using n-gram similarity as a metric.

On the other hand, the generative bigram model for author names does not assume exchangeability at the character level and is a model that can generate full author names. The probability of the same name under this model using marginal and conditional character frequencies is

$$p(\text{John}) = p(\text{J}|\text{<s>})p(\text{o}|\text{J})p(\text{h}|\text{o})p(\text{n}|\text{h})p(\text{</s>}|\text{n}) \tag{4.2}$$

$$p(\text{John}) = \frac{p(\text{<s>J})p(\text{Jo})p(\text{oh})p(\text{hn})p(\text{n</s>})}{p(\text{<s>})p(\text{J})p(\text{o})p(\text{h})p(\text{n})} \tag{4.3}$$

where $p(a|b)$ is the probability of the character 'a' appearing after the character 'b' in an author name. Thus in contrast to a BOW bigram model, the generative bigram model has additional terms for the marginal frequencies of each character.

The marginal character probabilities in the model are typically estimated from the number of times a character has occurred in the corpus, and the conditional probabilities are estimated by the count of the number of times a character has followed another character. Typically this model needs to be smoothed since many bigram orderings will not be observed in the training set. The hierarchical Dirichlet language model (MacKay and Peto, 1995) offers one way of smoothing counts using Dirichlet priors. The resulting model then allows predicting a character c_i given the previous character c_{i-1}. Overall,

the BOW trigram model will have more parameters to be estimated than the generative bigram model.

I use the bag-of-words assumption to model the words in the document. The model can represent function words, which are words which only serve a grammatical purpose, by allocating them to a cluster that is not associated to any author of the document. Common words which are unlikely to be used in a specialised topic will also be more likely to be allocated to this cluster.

My model in Section 4.3 assumes the same author entity can appear multiple times in the author list for a document. This is a result of authors being sampled from the base distribution with replacement whereas sampling without replacement would be a more accurate model. In reality, this has little practical effect as the data rarely makes such a repetition likely in the posterior. Even so it is worth commenting on why the model takes this form: my use of the multinomial distribution has a simple Dirichlet conjugate prior allowing the parameters to be integrated out and making inference easier.

I assume exchangeability for the parameters of observations, which is reasonable for short abstracts. In longer documents, certain words that represent a more general topic are more likely to appear in the background section of the document and more specialised words are likely to appear later when the methods are described. This could be modelled by using separate models for the different parts of the document or a dynamic topic model to describe each document's progression of topics.

I make no assumptions that author names that reference the same author identity must have the same last name or the same first initial. This allows the model to disambiguate a wider range of ambiguous names than if that assumption is made, as on some existing disambiguation test datasets. I also make no assumption that two authors with identical names are the same person.

4.3 THE AUTHOR × TOPIC SPACE MODEL

In my model, I utilise concepts of topics and author-entities. Each topic is defined by a multinomial distribution over the possible word occurrences, and each author entity defines an author name model, representing the fact that a particular author entity may be referred to in various ways. The basic idea is that if an author publishes on a particular topic on one occasion, they are more likely to publish on that same topic on another occasion, or use the same distribution over the vocabulary. I will describe the approach in terms of the generative model illustrated in Figure 4.1.

At the top level, the model describes the corpus using a Dirichlet process over the joint space of topics (word distributions) and author entities (author name models).

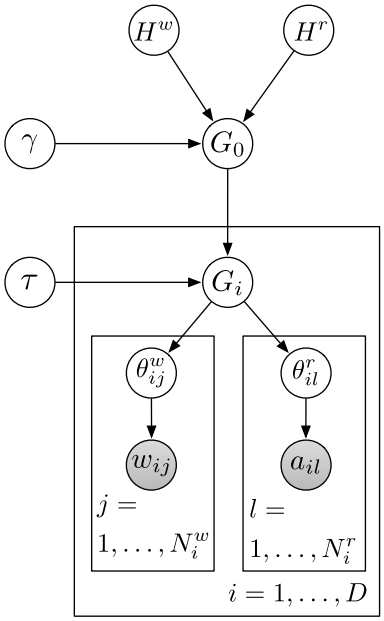


Figure 4.1: The author-topic space model for disambiguation in plate notation. w denotes words and otherwise a denotes authors. G denotes random measures and H denotes base measures. D is the number of documents, N^w_i is the number of words in document i and N^r_i is the number of authors in document i.

Because this DP is defined on this joint space, it incorporates the fact that a particular author is likely to publish repeatedly on the same topic or use the same distribution over the vocabulary. Generatively speaking, a random measure, denoted G_0, is drawn from this DP to represent the specific corpus. The base distribution of this DP is the product measure, $H^w \times H^r$, where $H^w = \text{Dirichlet}(\boldsymbol{\alpha}^w)$ is the base distribution over the space of topics (i.e. a distribution over word distributions). $H^r = \text{Dirichlet}(\boldsymbol{\alpha}^r)$ is the base distribution over the space of author-entities (i.e. a distribution over author name models). $\times$ is the Cartesian product so that $(H^w \times H^r)(t, a) = H^w(t)H^r(a)$ where t is a topic and a is an author name model. Superscript w in my notation represents distributions over words whereas r represents distributions over author name models. The base distribution $H^w \times H^r$ provides the prior representation of how likely particular author-name distributions (which are parameterised with θ^r_{ij}) and word distributions (parameterised with θ^w_{ij}) are, across many different corpora.

At the document level, for each document i, G_0 acts as a base measure for a draw G_i from another DP. This encapsulates the fact that each document will focus on only a small subset of all the possible topics, and will only be written by a small number of possible authors. These authors and topics will potentially be different from those of other documents. This DP is still defined in the joint topic-author space. The DP concentration parameter for each document can also be modified independently if the variability in the author word pairs is expected to be different. However, for the rest of

this chapter I assume that the concentration parameter is the same value for all documents.

For each document, the final stage is a generative model for the words and author names in document i. Each observation x_{ij} from document i denotes either the jth author field (a_{ij}) or jth word (w_{ij}) in the document. For each author, I sample from G_i and ignore the topic component, giving parameters θ^r_{ij}. These are used to generate an author name a_{ij} depending on the name variant model. For n-gram based name variant models, names that appears with different transcription errors are likely to share many n-grams.

Then for each word j, I also sample from G_i and ignore the author entity component of G_i giving a multinomial parameter θ^w_{ij}. I draw words $w_{ij} \sim \text{Multinomial}(\theta^w_{ij})$ from this multinomial distribution over the corpus vocabulary.

Ignoring one or other component of the joint distribution does not break the exchangeability assumption since every word can be assumed to have hidden data (the author who writes on the topic for that word). Each author also has hidden data (the topic they write upon). As a result, exchanging author and word parameters in the document results in the same joint distribution, and exchangeability is preserved.

In summary, I draw

$$G_0 \sim \text{DP}(\gamma, H^w \times H^r) \tag{4.4}$$

$$G_i \sim \text{DP}(\tau, G_0) \tag{4.5}$$

$$(\theta^w_{ij}, \theta^r_{ij}) \sim G_i \tag{4.6}$$

$$w_{ij} \sim \text{Multinomial}(\theta^w_{ij}) \tag{4.7}$$

$$a_{ij} \sim F^r(\theta^r_{ij}) \tag{4.8}$$

to obtain the multinomial parameters associated with each word and author name, and the words and author names generated from them. The distribution $F^r(\theta^r_{ij})$ is the model for the author name, which models variants of the author name as well as the transcription, OCR or other errors in the author name. Since these errors normally occur at the character level, it would be likely for names with long shared substrings to be variants of the original name. As inference is significantly simpler and faster when using conjugate priors, I considered two substring models that have conjugate priors. The BOW n-gram model is a bag-of-words model for strings that share substrings of length n, where in this chapter I use $n = 3$ for performance. Wallach (2006) proposed a bigram topic model that does not need to make a bag of words assumption. I extend this model nonparametrically in Section 4.4.1 since the number of topics is not fixed. This enables

a generative model of author names based on substrings of length n, with $n = 2$ for performance. I refer to this model as the generative bigram model.

4.4 AUTHOR AND TOPIC CLUSTERS

In this section, I describe the posterior distributions for the probabilistic models, including both the generative n-gram model and the bag-of-words n-gram model for modelling name variants. Observations are modelled as being drawn from a mixture of distributions of the form $f^w(w|\boldsymbol{\theta}) = \prod_{t=1}^{V} \theta_t^{w_t}$ for each word where $f^r(a|\boldsymbol{\theta})$ for each author where f^r depends on the author variant model and $\boldsymbol{\theta}$ denotes the distribution's parameters. The observations are either authors or words. For each cluster k, the parameters $\boldsymbol{\phi}_k = (\boldsymbol{\phi}_k^w, \boldsymbol{\phi}_k^r)$ determine the distribution of observations from cluster k.

For ease of computation, additional latent indicator variables $\mathbf{z}$ are used for each author and each word to identify which cluster they are allocated to in the infinite mixture. $f_{\text{new}}^w(w_{ij})$ is the posterior distribution for a word with the $(H^w \times H^r)$ prior and the single observation w_{ij} and similarly for the author $f_{\text{new}}^r(a_{ij})$.

$$\begin{aligned} f_{\text{new}}^w(w_{ij}) &= \int f^w(w_{ij}|\boldsymbol{\phi}^w)\,\mathrm{d}H^w(\boldsymbol{\phi}) \\ &= b\frac{\Gamma\left(\sum_t \alpha_t^w\right)}{\Gamma\left(\sum_t \alpha_t^w + w_{ijt}\right)} \prod_t \frac{\Gamma(\alpha_t^w + w_{ijt})}{\Gamma(\alpha_t^w)} \end{aligned} \tag{4.9}$$

where b is a normalisation constant. Since the Dirichlet distribution is the conjugate prior, this integral is analytically tractable and samples can be drawn from this posterior. The conditional distributions for the words given cluster k and all other data points except for w_{ij} is

$$f_k^{w,-w_{ij}}(w_{ij}) = \int f^w(w_{ij}|\boldsymbol{\phi}^w)\,\mathrm{d}H_{-ij,k}(\boldsymbol{\phi}) \tag{4.10}$$

where $H_{-ij,k}$ is the posterior distribution of $\boldsymbol{\phi}$ with the prior and all data points allocated to cluster k for which $(i', j') \neq (i, j)$:

$$H_{-ij,k}(\boldsymbol{\phi}) \propto \prod_{\substack{(i',j')\neq(i,j) \\ z_{i'j'}=k}} f(x_{i'j'}|\boldsymbol{\phi})(H^w \times H^r)(\boldsymbol{\phi}) \tag{4.11}$$

Since clusters are shared, the parameters $\boldsymbol{\phi}$ are shared between documents. $\boldsymbol{\phi}_k$ factorises into $\boldsymbol{\phi}_k^w$ for the parameters for the distribution that is over the words and $\boldsymbol{\phi}_k^r$ for the remaining distribution over bigrams. Since the Dirichlet distribution is the conjugate prior, the parameters can be integrated out so that, in terms of bookkeeping, only the

allocations of data points to clusters need to be kept rather than the cluster parameters. We get the following conditional densities,

$$f_k^{w,-w_{ij}}(w_{ij}) = b\frac{\Gamma\left(\sum_t s_{kt}\right)}{\Gamma\left(\sum_t s_{kt} + w_{ijt}\right)} \prod_t \frac{\Gamma(s_{kt} + w_{ijt})}{\Gamma(s_{kt})} \tag{4.12}$$

where b is a normalisation constant and $\mathbf{s}_k$ is defined as

$$\mathbf{s}_k = \alpha^w + \sum_{\substack{(i',j')\neq(i,j) \\ z_{i'j'}=k}} w^w_{i'j'} \tag{4.13}$$

which are the posterior parameters derived from (4.9) for the words in cluster k with the prior $(H^w \times H^r)$ and all data points w_{ij} and a_{ij} that are assigned to cluster k for which $(i', j') \neq (i, j)$. When cluster k does not contain any words when sampling for a word or any authors when sampling for an author, then I draw a sample from the posterior distribution in (4.9) and similarly for the authors.

4.4.1 *Nonparametric generative n-gram model*

To model the corruption and variants of author names, a flexible model is needed that can tolerate small changes in the name. A generative approach to this can be taken following the bigram topic model (Wallach, 2006). The bigram topic model is restrictive since like in the LDA model, it still requires the specification of the number of topics to use for the model whereas the nonparametric model I describe in this section does not require this to be prespecified. This allows the model to adapt to the complexity of the data and use more or less topics as needed. In this chapter, I use $n = 2$ for the generative n-gram model. The likelihood model for the authors with this model is $f^r(a|\boldsymbol{\theta}) = \prod_n \prod_m \theta_{m|n}^{a_{m|n}}$ where m and n are character indices, $a_{m|n}$ is the number of times character m has appeared after character n in the author name a and $\boldsymbol{\theta}$ denotes the distribution's parameters.

The conditional distributions for the authors given cluster k and all other data points except for a_{ij} is

$$f_k^{r,-a_{ij}}(a_{ij}) = \int f^r(a_{ij}|\boldsymbol{\phi}^r)\,\mathrm{d}H_{-ij,k}(\boldsymbol{\phi}) \tag{4.14}$$

Since the Dirichlet distribution is the conjugate prior, the parameters can be integrated out and now we only need to Gibbs sample the cluster indicators rather than the cluster parameters. We get the following conditional densities,

$$f_k^{r,-a_{ij}}(a_{ij}) = b \prod_n \frac{\Gamma(u_{kn})}{\Gamma(u_{kn} + a_{ijn})} \prod_m \frac{\Gamma(u_{km|n} + a_{ijm|n})}{\Gamma(u_{km|n})} \tag{4.15}$$

where b is a normalisation constant, a_{ijn} is the number of occurrences of the character n in the author name a_{ij}, $u_{kn} = \sum_m u_{km|n}$ and

$$u_{km|n} = \alpha_n^r + \sum_{\substack{(i',j') \neq (i,j) \\ z_{i'j'} = k}} a_{i'j'm|n} \tag{4.16}$$

which are the posterior parameters for the authors in cluster k. Thus, in terms of bookkeeping for this author name model, a matrix of how many times one character followed another character in author names needs to be stored for each cluster.

These posterior parameters for the Dirichlet compound multinomial distribution in (4.15) have the effect of 'damping' the word and bigram counts for cluster k by the addition of the concentration parameter α^r and reducing the effect of large counts on the likelihood of the author entity.

4.4.2 *BOW n-gram model*

An alternate model for the corruption or variation of author names is to use a bag-of-words assumption with a set of n-grams. This model is degenerate in that it models non-names where the trigrams don't tile as well as full names but it is simpler to implement. However, as described in Section 4.2, this model makes a false independence assumption and so is probabilistically deficient. Each author name is represented as a multinomial for a set of n-grams, where in the experiments, n is set to 3. The likelihood model for the authors with this model is $f^r(a|\theta) = \prod_n \theta_n^{a_n}$ where n indexes into the n-grams used for author names in the corpus and a_n is the number of occurrences of the n-gram n in the author name a and θ denote the distribution's parameters.

The conditional distributions for the authors given cluster k and all other data points except for a_{ij} is given by (4.14) again.

Since the Dirichlet distribution is the conjugate prior, the parameters can be integrated out and now we only need to Gibbs sample the cluster indicators rather than the cluster parameters. We get the following conditional densities,

$$f_k^{r,-a_{ij}}(a_{ij}) = b\frac{\Gamma(u_k)}{\Gamma(u_k + a_{ij})}\prod_n \frac{\Gamma(u_{kn} + a_{ij,n})}{\Gamma(u_{kn})} \tag{4.17}$$

where b is a normalisation constant and u_k is defined as

$$u_{kn} = \alpha_n^r + \sum_{\substack{(i',j')\neq(i,j)\\ z_{i'j'}=k}} a_{i'j'n} \tag{4.18}$$

which are the posterior parameters for the authors in cluster k. The bookkeeping for this author name model is simpler than the generative model as only a vector of counts of trigram occurrences in author names needs to be kept for every cluster.

The posterior parameters in the Dirichlet compound multinomial distribution in (4.17) have the same effect as in the generative n-gram model.

4.5 INFERENCE

Since calculating the exact posterior under DP models is intractable as described in Section 3.8.6, I use approximate algorithms. Due to the popularity and ease of implementing and verifying a MCMC approach for Dirichlet processes, I use collapsed Gibbs sampling based on the Pólya urn scheme for inference in this model. In this section, I describe two Gibbs sampling algorithms which arise from either integrating out the global random measures or sampling them.

Inference in my model follows from the auxiliary variable Gibbs sampler in the Chinese restaurant process (CRP) description of the marginal probabilities of the DP (Teh, Jordan et al., 2006) as outlined in Section 3.8.6. For the HDP mixture, the CRP is extended to the Chinese restaurant franchise (CRF) in which there is a set of Chinese restaurants, one for each group of observations and in this case, each document. Cluster assignments are done through two layers of the model, the table level and the data point level. First, customers or data points, which may be authors or topics, are assigned to tables. Each table is then allocated a dish, which corresponds to the parameters for that table. Tables that have the same dish are essentially assigned to the same mixture component or cluster. The number of unique dishes is then the number of latent clusters. These clusters are shared between all the restaurants. The result is that data points at the same table share the same cluster, which in my model is the author entity and topic

that generates the data points. In relation to the CRF, I refer to a table when referring to an inferred grouping of data points in a single document and a group when referring to a document.

The final inferred authors for a document are the clusters that have been allocated to the author field data points. Each cluster, which is in the author × topic space, contains the *n*-gram parameters and topic for an author entity. Those clusters that do not have an assigned author can be thought of as function word clusters that are not specific to any author.

The likelihood of the data given the other clusters comes from the CRP, which induces a clustering of the data due to the DP prior. This can be seen from the connection to the Pólya urn model (Blackwell and Macqueen, 1973). Since the parameters are exchangeable, I can reorder the current parameter to be the one that was most recently sampled and draw from a distribution conditional on the previous data points. After integrating out G_i, the likelihood of the data point conditional on it being assigned to a new table, t^{new}, is then

$$p(x_{ij}|\mathbf{t}^{-ij}, t_{ij} = t^{\text{new}}, \mathbf{k}) = \sum_{k=1}^{K} \beta_k f_k^{-x_{ij}}(x_{ij}) + \beta_{\text{new}} f_{\text{new}}(x_{ij}) \tag{4.19}$$

$$G_0' \sim \text{DP}(\gamma, H) \tag{4.20}$$

$$G_0 = \sum_{k=1}^{K} \beta_k \delta_{\boldsymbol{\phi}_k} + \beta_{\text{new}} G_0' \tag{4.21}$$

$$(\beta_1, \ldots, \beta_K, \beta_{\text{new}}) \sim \text{Dirichlet}(m_{\cdot 1}, \ldots, m_{\cdot K}, \gamma) \tag{4.22}$$

where G_0' is the global measure, G_0 is the posterior for G_0', m_{ik} is the number of tables in document i allocated to cluster k and $\mathbf{k}$ are the allocations of tables to clusters. Eq. (4.22) represents samples from the posterior of the base distribution, G_0, where $\boldsymbol{\beta}$ are the stick-breaking weights for the sampled G_0. The parameters $\boldsymbol{\phi}_k$ of each cluster are also integrated out.

Each data point in document i, position j is first allocated to a table t_{ij} proportional to how many other data points have already been allocated to that table. Table t in document i is then allocated to a cluster k_{it} in the topic and author product space. I will refer to this method of indirect cluster assignments as the CRF sampler. In this sampler, both G_0 and G_i are integrated out so allocations of tables to clusters need to be tracked.

The cluster and table allocations are sampled without any distinction between authors and words. Allocating words to a table is only affected by the posterior parameters with respect to the other words on the table and not to any of the authors. Similarly, allocating

authors to a table is only affected by the posterior parameters with respect to the other authors. This is a result of the factorisation of the base distribution.

4.5.1 *CRF sampler*

The CRF sampler involves alternately sampling the table allocation for each observation and the cluster allocation for each table. In this sampler, both G_0 and G_i are integrated out.

1. For each document i,
   a) For the jth word (w_{ij}) and author (a_{ij}) in document i, which is denoted as x_{ij}, sample t_{ij} for the table allocation, where n_{it}^{-ij} is the number of observations allocated to table t in document i excluding x_{ij},

$$p(t_{ij} = t|\mathbf{t}^{-ij}, k_{it} = k, \mathbf{k}^{-ij}) \propto \begin{cases} n_{it}^{-ij} f_k^{-x_{ij}}(x_{ij}), & \text{if } t = t_{ij'} \text{ for some } j' \neq j, \\ \tau \frac{m_{\cdot k}^{-ij}}{m_{\cdot\cdot}^{-ij} + \gamma} f_k^{-x_{ij}}(x_{ij}), & \text{if } t = t^{\text{new}}, k_{it^{\text{new}}} = k_{i't'}, \text{for some } (i', t') \neq (i, t), \\ \tau \frac{\gamma}{m_{\cdot\cdot}^{-ij} + \gamma} f_{\text{new}}(x_{ij}), & \text{if } t = t^{\text{new}}, k_{it^{\text{new}}} = k^{\text{new}} \end{cases} \tag{4.23}$$

   where f is the corresponding likelihood function for the word and author name variant model and when a new table is chosen, a cluster ($k_{it^{\text{new}}}$) is sampled for it according to the probabilities above. If a table becomes empty, the table and its cluster allocation is removed since $n_{it}^{-ij} = 0$ so the table will never be allocated to any observations in the future.

   b) Sample k_{it} for the table to cluster allocation, where t ranges over the tables in document i,

$$p(k_{it} = k|\mathbf{t}, \mathbf{k}^{-it}) \propto \begin{cases} m_{\cdot k}^{-it} f_k^{-\mathbf{x}_{it}}(\mathbf{x}_{it}), & \text{if } k = k_{i't'} \text{ for some } (i', t') \neq (i, t), \\ \gamma f_{\text{new}}(\mathbf{x}_{it}), & \text{if } k = k^{\text{new}}. \end{cases} \tag{4.24}$$

In the CRF sampler, many data points can change cluster at the same time since changing the cluster of one table changes the cluster of all the data points that are allocated to that table. This can potentially help clusters to merge. However, since the allocation of data points to tables are a result of their prior clustering due to the DP, the probability that a table will change its cluster will often be small.

4.5.2 *Direct sampler*

In the CRF sampler, data points are indirectly allocated to clusters through first being allocated to tables and this requires significant bookkeeping. As described in Teh, Jordan et al. (2006), in the direct sampler G_0 is sampled from instead of integrated out to reduce the bookkeeping that is required. Thus to represent the tables for each document, only counts for the number of tables, m, allocated to each cluster are needed. This can be used to sample the stick-breaking weights β for each cluster. Data points can then be directly allocated to clusters instead of needing to be allocated to tables first. The direct allocation is done via a cluster indicator z variable.

1. For each document i,

   a) For each observation j in document i, sample z_{ij} for the cluster allocation, where n_{ik}^{-ij} is the number of data points allocated to cluster k in document i excluding x_{ij},

$$p(z_{ij} = k|\mathbf{z}^{-ij}) \propto \begin{cases} (n_{ik}^{-ij} + \tau\beta_k) f_k^{-x_{ij}}(x_{ij}), & \text{if } k = z_{i'j'} \text{ for some } (i', j') \neq (i, j), \\ \tau\beta_{\text{new}} f_{\text{new}}(x_{ij}), & \text{if } k = k^{\text{new}}. \end{cases} \tag{4.25}$$

   This samples the cluster allocation proportional to the number of existing data points allocated to the cluster plus a pseudo-count representing the prior probability of selecting a new table that is allocated to that cluster ($\tau\beta_k$). If a new cluster k^{new} is sampled during one of the steps above, then draw $b \sim \text{Beta}(1, \gamma)$, set the new weight $\beta_{k^{\text{new}}} = b\beta_{\text{new}}$ and set the new β_{new} to $(1 - b)\beta_{\text{new}}$. b corresponds to the weight of the new atom that is instantiated from the DP.

   b) Sample m_{ik}, where k ranges over the clusters, by generating n_{ik} uniformly distributed random variables $u_1, \ldots, u_{n_{ik}}$ in $[0, 1]$, and setting:

$$m_{ik} = \sum_{m=1}^{n_{ik}} \mathbf{1}\left[u_m \geq \frac{\tau\beta_k}{\tau\beta_k + m}\right] \tag{4.26}$$

   where $\mathbf{1}$ is the indicator function. Intuitively this is sampling the number of document-level clusters by counting how many new document-level clusters would be needed using the Chinese restaurant process. This is calculated for all the data points for each document that are assigned to each corpus-level cluster.

This gives the following distribution (Antoniak, 1974), where $s(m, n)$ are unsigned Stirling numbers of the first kind,

$$p(m_{ik} = m|\mathbf{z}, \mathbf{m}^{-ik}, \boldsymbol{\beta}) = \frac{\Gamma(\tau\beta_k)}{\Gamma(\tau\beta_k + n_{ik})} s(n_{ik}, m)(\tau\beta_k)^m. \tag{4.27}$$

2. Sample $\boldsymbol{\beta}$ as in (4.22).

4.5.3 *Parameter optimisation*

I use Gamma priors for the concentration parameters τ and γ and sample from their posteriors every iteration as part of Gibbs sampling by the auxiliary variable method (Escobar and West, 1995) as described in Section 3.8.4 and Section 3.8.7. I estimatedthe parameters of the base distributions over topics via a fixed-point method (Minka, 2000), which results in Dirichlet parameters that correspond to the frequently used TF-IDF measure weighing rare words more heavily. However, that did not perform as well as using a symmetric base distribution. I use an uninformative symmetric prior for the n-grams.

4.6 EXISTING WORK

In comparison with the author-topic model described in Section 3.6, the author-topic space model for disambiguation has a number of advantages. Chief among these is that a training set is not needed and that the real authors of the documents do not need to be known. As a result, a nonparametric model is needed to model the unknown number of authors that exist in an unseen dataset. In addition, the author-topic space model has no explicit model of author names so authors with slight differences in names cannot be handled without preprocessing. My model can take into account changes at the character level to author names so is more tolerant with regard to errors in author fields. Since the n-gram based author name likelihood has a conjugate prior, inference in the model is simpler than more complex name models.

The association of authors with topics are also handled differently in the two models. The author-topic model associates authors with a distribution over topics whereas the model described in this chapter associates each author with a unique topic. Modelling an author as being associated with a range of topics results in a broader distribution over the vocabulary, in contrast to modelling an author with a single topic. This broader distribution would likely make it harder for authors to be disambiguated on the basis of

any distinctive vocabulary they use in documents. Modelling an author with a unique topic also prevents an author from being allocated to function word topics. These topics are ones that mostly contain words that are shared amongst almost all documents in the corpus and no are not useful for discriminating between them. The author-topic model does not explicitly model these words and so it is possible for function word topics to be assigned to authors and so overweight authors which are prolific in the corpus.

Compared to the LDA-ER and similar models for citation matching described in Section 2.1.1, the model I developed in this chapter mostly has an advantage in terms of disambiguation. LDA-ER does not use the text of the document for disambiguation and so it can be easy for different authors with the same name to be inferred to be the same entity. LDA-ER also require the prior for the number of authors to be learnt in advance on a separate training set. The model I described in this chapter samples from the posterior over the concentration parameters and so a sufficiently broad prior means those parameters do not need to be learnt from a training set. Finally, LDA-ER uses a domain specific name variation model with no conjugate prior. The use of the conjugate prior in my name variation model allows the model parameters to be integrated out so that mixing is much faster. Their name variation model is also only for author names while the *n*-gram model used in my model can be used for any string that becomes corrupted or has variation at the character level such as titles, addresses or dates.

Finally, parallels can be drawn between my model and the Gaussian-multinomial LDA (GM-LDA) model of David M. Blei and Jordan (2003). The goal of the GM-LDA model is associating different regions in an image with words in the image caption. Each word or region in an image is allocated to a cluster where the cluster for both types of data points is chosen from a multinomial distribution. The model I described in this chapter can be thought of as a similar model where the image regions are replaced with author names and image captions with document text. My model is similar in that the true cluster for the image regions and the image captions are unknown unlike in the author-topic model or LDA-ER model. However, they focus on using their model to model the probability of the caption given a set of regions whereas the focus in my model is on disambiguating a set of authors rather than modelling text. They also use a simple Gaussian model for the image regions rather than the string-level variation model I use for author names. Additionally, their model requires specifying the number of topics in advance since it is a parametric model.

4.7 EVALUATION METRICS

4.7.1 B^3 *metric*

The approach most commonly used for evaluation in coreference resolution is the B^3 scoring algorithm (Bagga and Baldwin, 1998a). This algorithm can be used to calculate precision and recall based on the presence or absence of authors in the equivalence class of author entities averaged across all the author entities. The B^3 algorithm was proposed as an alternative to the MUC algorithm for scoring clusters. An important reason is that the B^3 algorithm gives credit for separating singletons compared with pairwise scoring or similar methods. The scoring algorithm is defined as follows where an entity in this case refers to an instance of an author name:

- For each entity $i = 1, \ldots, N$, let
  - nCE be the number of correct entities in the cluster containing entity i,
  - nE be the number of entities in the cluster containing entity i,
  - nTE be the number of entities in the true cluster containing entity i,
  - $\text{Precision}_i = \frac{\text{nCE}}{\text{nE}}$,
  - $\text{Recall}_i = \frac{\text{nCE}}{\text{nTE}}$.
- Then
  - B^3 precision $= \frac{1}{N} \sum_{i=1}^{N} \text{Precision}_i$,
  - B^3 recall $= \frac{1}{N} \sum_{i=1}^{N} \text{Recall}_i$.

For example, suppose author *Al* appeared in a corpus 4 times with the names *Al*, *Ai*, *Aj* and *A1* along with other authors. Then imagine that the output of the algorithm was 2 clusters, one with the names *Al*, *Ai*, *Aj* and *Bal* and the other with just *A1*. The calculation for the first entity *Al* would be: $nCE = 3$, $nE = 4$, $nTE = 4$. So $\text{precision}_i = 3/4 = 75\%$ and $\text{recall}_i = 3/4 = 75\%$ for this entity. The B^3 precision and B^3 recall is then the average of the individual recall and precisions for each of the other 5 entities: *Al*, *Ai*, *Aj*, *A1*, *Bal*.

The F1 score is the harmonic mean of precision and recall: $2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$.

4.7.2 *Pairwise clustering score*

The pairwise clustering score is also used to evaluate clusterings by examining pairs of author names that are clustered together.

For each possible pair of observed author names, where the true cluster is the true author entity, define:

- a, the number of author pairs that are in the same cluster and the same true cluster,
- b, the number of author pairs that are in the same cluster and different true clusters,
- c, the number of author pairs that are in different clusters but the same true cluster,
- d, the number of author pairs that are in different clusters and different true clusters.

Then

- Recall $= \frac{a}{a+c}$,
- Precision $= \frac{a}{a+b}$.

The F1 score is the harmonic mean of precision and recall as above: $2\frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$.

4.8 EXPERIMENTS

I used a set of documents from the CiteSeer database that have been hand labelled with ground truth identities for the authors. This is a citation database which contains papers on various computer science topics, often the papers and paper metadata are extracted from scanned-in versions of the physical documents. The dataset includes author names, some of which are corrupted due to errors in the OCR process. I removed punctuation and other non alphanumeric characters from the documents. The dataset was created by Giles, Bollacker and Lawrence (1998) and cleaned by Bhattacharya and Getoor (2006). I retrieved the abstracts for each of the documents in the dataset from the CiteSeer database. This set of documents contains 852 documents, 867 unique names, 706 underlying author entities and 1,680 author references. The abstracts contain 42,507 words with a vocabulary size of 5,695 words. This is the same dataset used in one of the experiments in Chapter 5. The words are lowercased, stemmed and a standard stoplist is applied*. The 10 words appearing in the most documents are discarded since these are usually function words or words that are standard to the domain of the corpus. The n-grams extracted from author names are lowercased and include beginning and end of name markers. These markers allow n-grams at the ends of a name to be distinguished from those in the middle of a name.

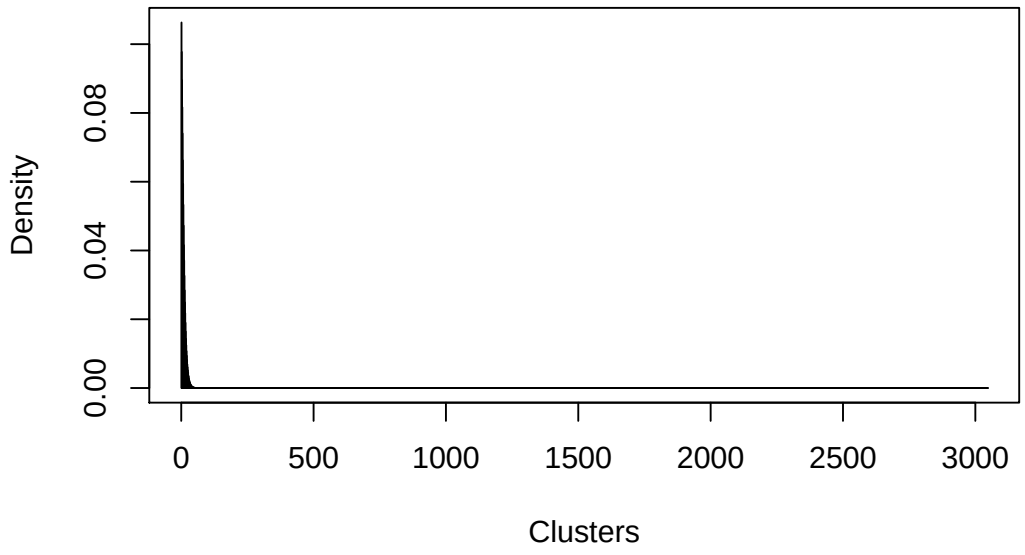
* Downloaded from `http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words`

Since the CiteSeer dataset has very little ambiguity and most of the authors can be identified by an exact string match on their name, I created another separate dataset where some of the authors are conflated together. This was done by discarding the last names of the authors in the CiteSeer dataset. The process resulted in a dataset with 362 unique names or roughly 2 author entities for every unique author name. I will refer to this as the 'conflated CiteSeer dataset'.

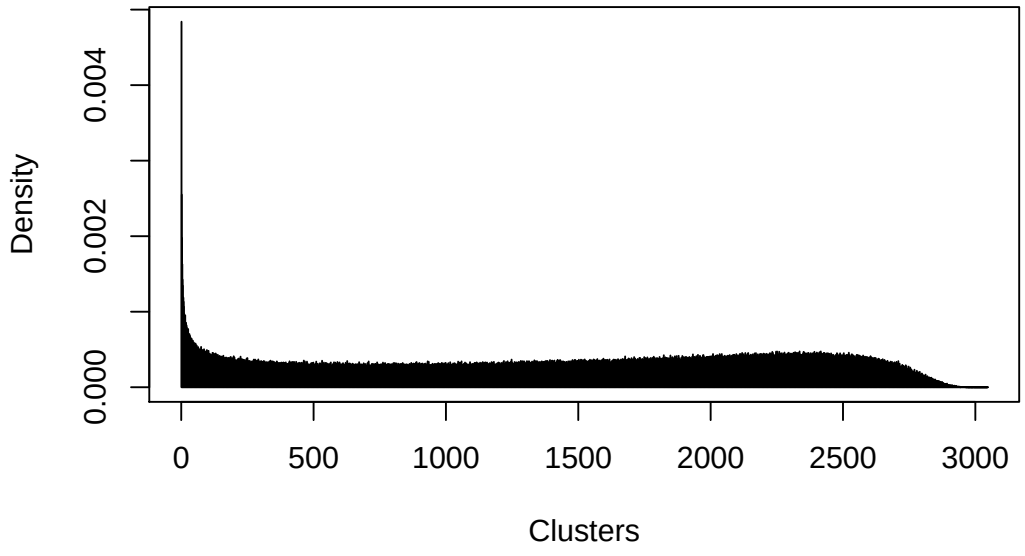
The results of the experiments were evaluated with the commonly used B^3 algorithm, described in Section 4.7.1, used to evaluate coreference resolution systems. Ground truth is available for the CiteSeer dataset and the scores are measured against that ground truth.

I performed experiments with the two inference algorithms, the CRF sampler and the direct sampler to compare their performance and convergence speed. These samplers were used to infer the author entities and topics in the corpus. Each author and word was initialised in its own cluster. The concentration parameter for the symmetric Dirichlet prior for the n-gram distributions was fixed at 10^{-9} for the generative bigram model and 0.001 for the BOW trigram model, found from optimising the B^3 F1 score by testing different orders of magnitudes for the concentration parameter on 10% of the dataset. The parameter for the symmetric Dirichlet prior for the topics, α^w, was set to 0.01 in common with the author-topic model and other topic models. A Gamma$(0.4, 0.0002)$ prior was placed on the initial concentration parameters for the DP γ and a Gamma$(1, 1)$ prior on τ. The concentration parameters were then sampled from their posteriors during every iteration. The standard vague prior was used for τ because there are likely to be very few tables or document-level clusters per document as many of the data points in a document are expected to belong to the same cluster. Changing the prior parameters by an order of magnitude do not significantly affect the results. The prior for γ was found by minimising the KL-distance between the number of prior clusters and a uniform distribution as in Section 3.8.4, seen in Figure 4.2b. Using a standard Gamma$(1, 1)$ prior for γ produced poorer results with lower precision since there is a very high prior probability of having very few clusters as seen in Figure 4.2a. This causes data points to be heavily overclustered so somewhat reduces the deduplication performance as measured by F1 by a few percent and causes unrelated data points to be clustered together. The prior effectively puts an unreasonable assumption for the number of author identities.

Experiments were run on the remaining held-out dataset with the algorithms implemented in C++. The diagnostic methods by Gelman and Rubin described in Section 3.4.1 were used to assess whether the Markov chains have converged.



(a) The prior distribution for the number of topics for the CiteSeer dataset given a standard $\text{Gamma}(1, 1)$ prior on γ.



(b) Prior distribution for the number of topics for the CiteSeer dataset given a $\text{Gamma}(0.5, 0.0002)$ prior on γ found by minimising the KL-distance between the prior number of clusters and the uniform distribution.

Figure 4.2: Different prior distributions for the number of topics with different priors on γ.

Results were collected after 100,000 iterations of sampling. This appears to be sufficient for the direct sampler from examining the Gelman-Rubin potential scale reduction factor (PSRF) plots for various posterior quantities including the number of entities, the number of topics, the size of the biggest cluster, the hyperparameter values and the B^3 and pairwise scores. These plots are in Figure 4.3. The plots show that using the direct sampler, the PSRF for the B^3 F1 score appears to converge by 5,000 iterations as it is close to 1.0, however, after that the PSRF grows to be 2.4 at 100,000 iterations. This shows that the chains will have falsely appeared to converge if less than 10,000 iterations have been performed, which is typical in other experiments.

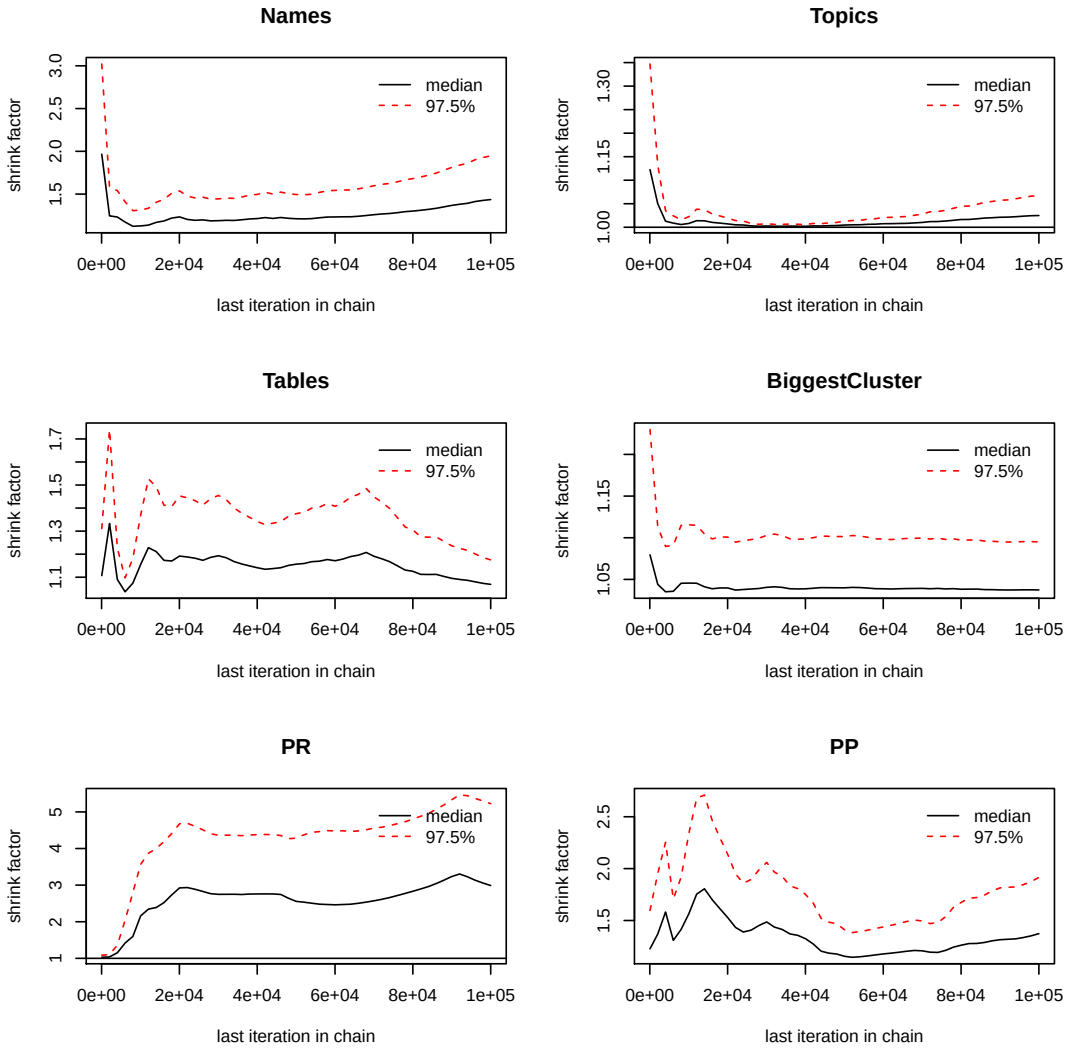
On the other hand for the CRF sampler in Figure 4.3, there is a similar dip in PSRF at around 7,000 iterations, however, the PSRF grows and then declines to be around 1.5 at 100,000 iterations. This indicates that the CRF sampler is converging faster than the direct sampler. An explanation for this is that since the global random measure G_0 is integrated out in the CRF sampler rather than sampled, the chain can mix faster. This could be because the weights in G_0 are only sampled after a round of sampling the entity indicators for the corpus. Since the data points in the corpus can be frequently reassigned, this slow updating of G_0 can slow down the mixing of the chain.

Even though the PSRF for the B^3 scores shows the chains have not fully converged by 100,000 iterations, the trace plot in Figure 4.3 shows that running the sampler for more iterations only improves performance by 0.5% in terms of the B^3 F1 score. Thus, due to the long time required for such long simulations, the results are given are for chains that are run for 100,000 iterations. Using different initial states where the words are all allocated to one entity cluster or all to individual clusters also resulted in very similar results after 100,000 iterations showing that the initial state is not important.

4.8.1 *Citation dataset*

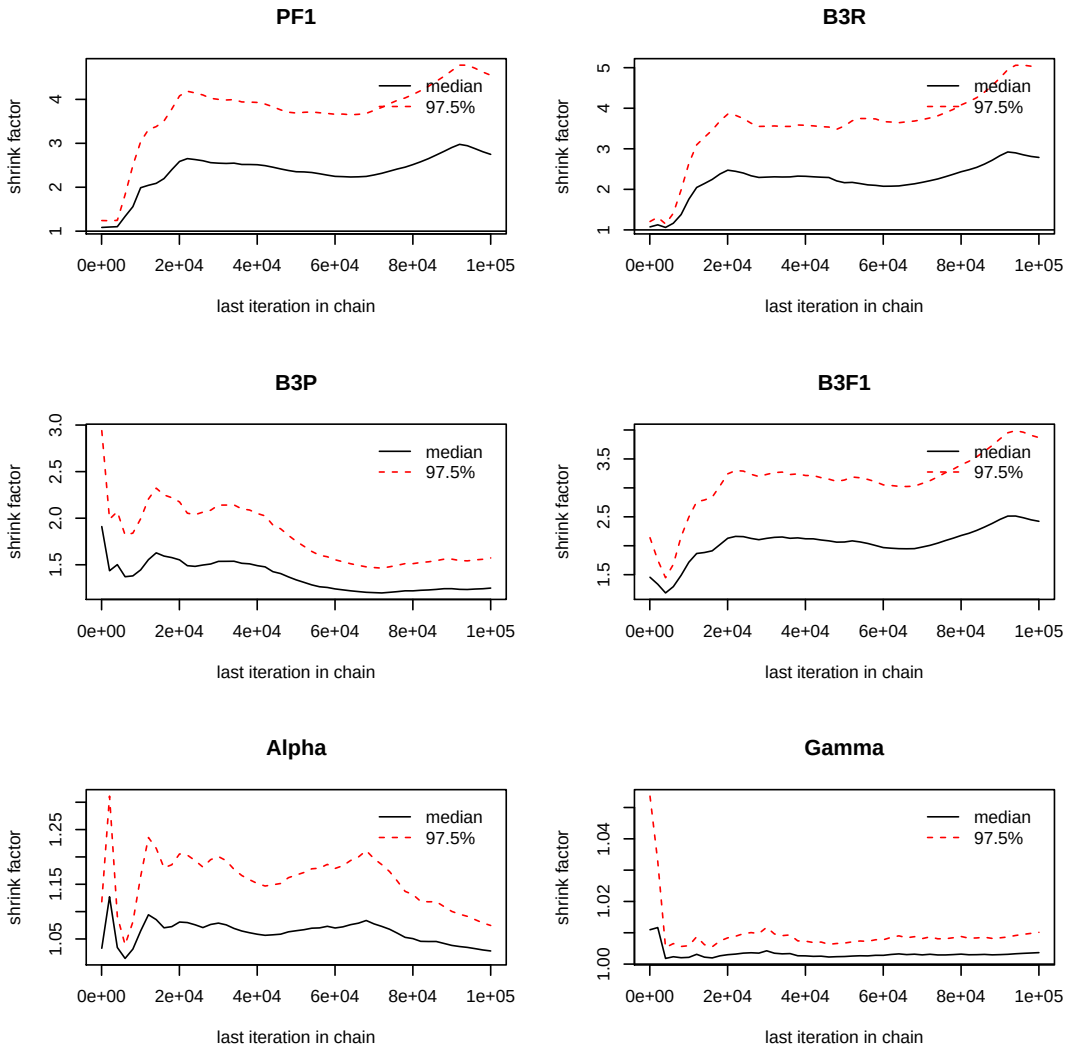
The density of some of the main posterior quantities for the direct sampler across the second half of 5 chains are shown in Figure 4.4. The densities are generally unimodal and the density for the scores are generally Gaussian showing that the chains have mixed well.

The performance of each inference method on the CiteSeer dataset is given in Table 4.1. The models generally have slightly higher scores when using the CRF sampler. This can be explained by the faster convergence of the chains in the CRF sampler so that they are less affected by the initialisation of the sampler. The difference, however, is very small, and the increase in computation time and inference complexity is signific-



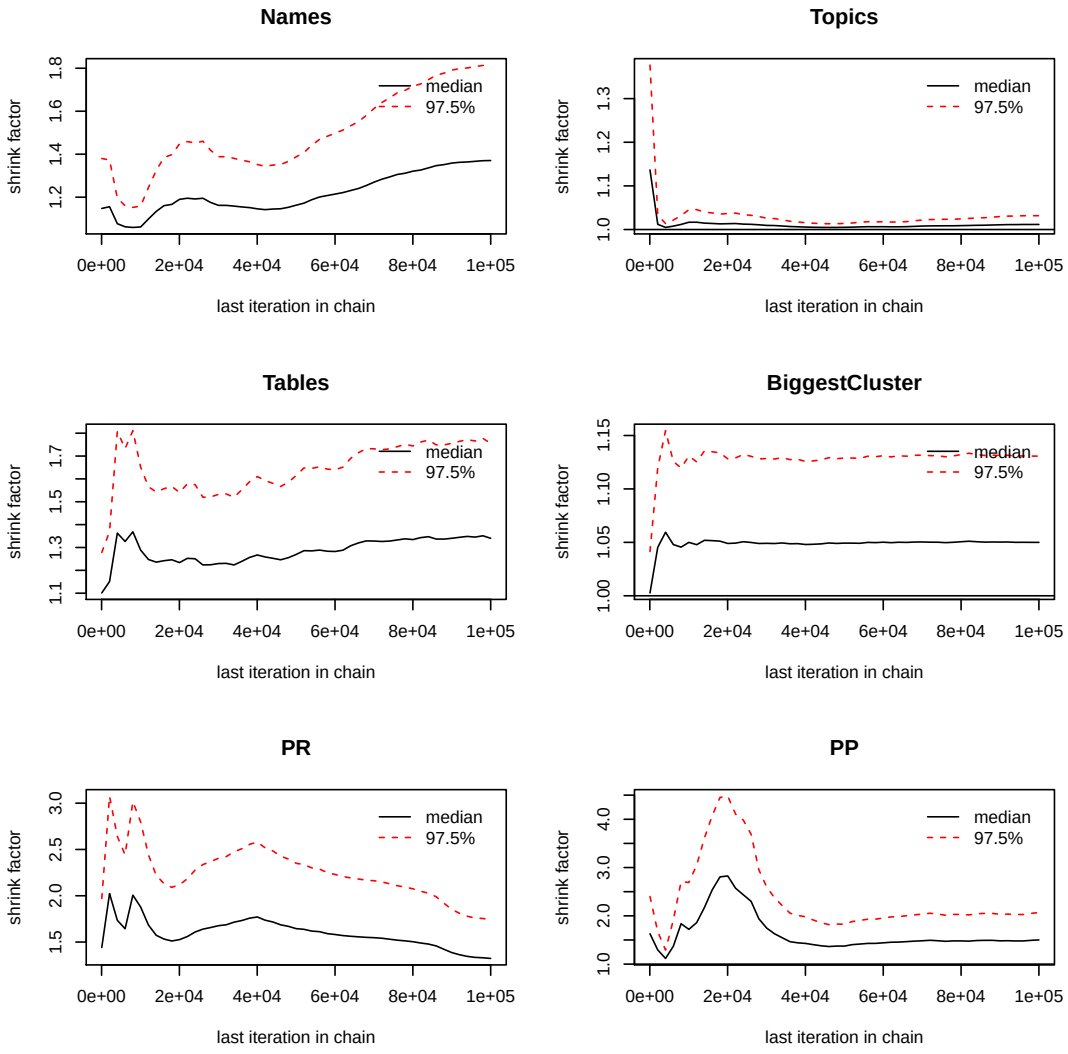
(a)

Figure 4.3: First half of Gelman-Rubin scale reduction factor plots calculated over 5 chains. This is for the generative bigram model with the direct sampler. *Names* is the number of latent entities, *topics* is the number of topics, *tables* is the number of tables, *biggestcluster* is the size of the biggest cluster, *PR* is pairwise recall, *PP* is pairwise precision.



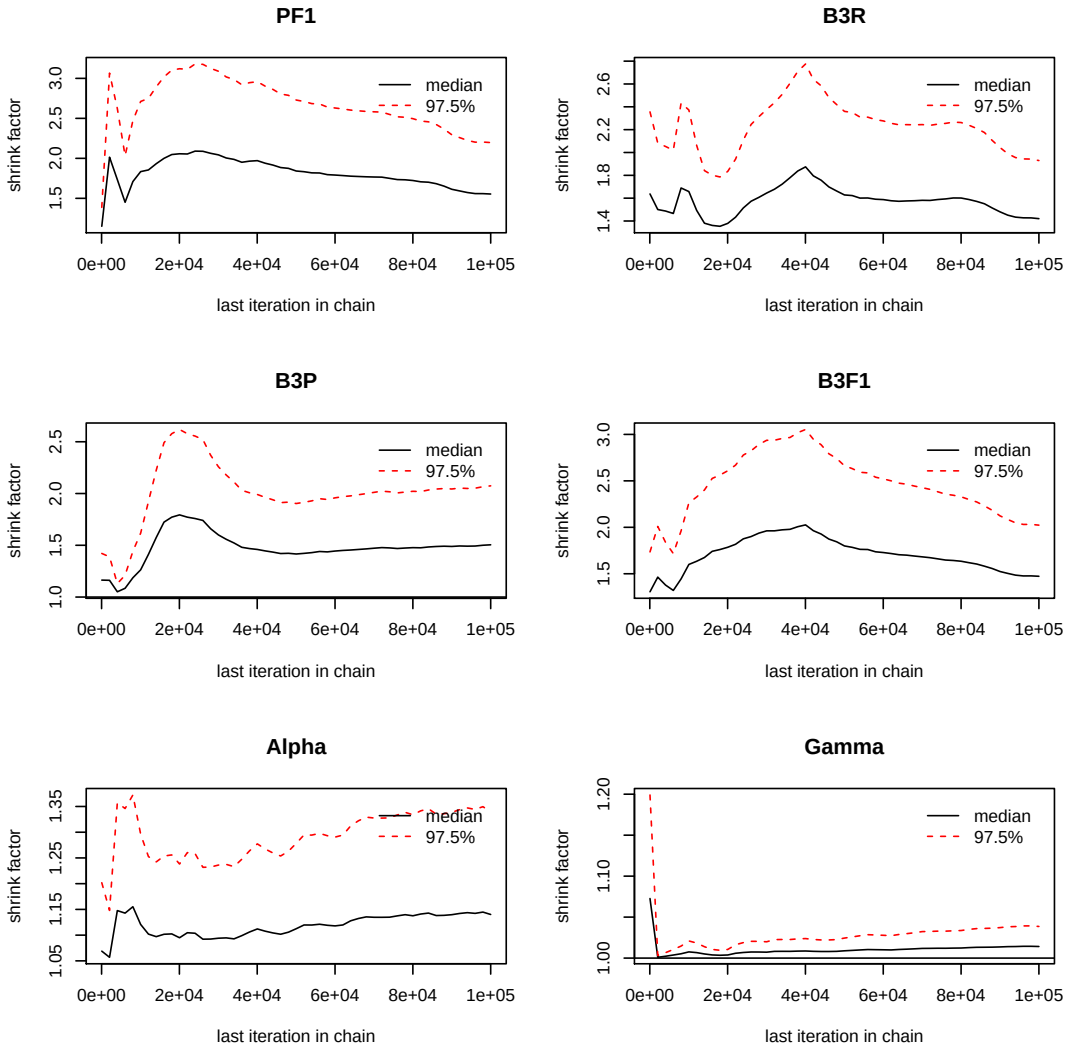
(b)

Figure 4.2: Second half of Gelman-Rubin scale reduction factor plots calculated over 5 chains. This is for the generative bigram model with the direct sampler. *PF1* is pairwise F1, *B3R* is B^3 recall, *B3P* is B^3 precision, *B3F1* is B^3 F1, *alpha* is α and *gamma* is γ.



(a)

Figure 4.3: First half of Gelman-Rubin scale reduction factor plots calculated over 5 chains. This is for the generative bigram model with the CRF sampler. The quantity labels are the same as those in Figure 4.3.



(b)

Figure 4.2: Second half of Gelman-Rubin scale reduction factor plots calculated over 5 chains. This is for the generative bigram model with the CRF sampler. The quantity labels are the same as those in Figure 4.3. It can be seen that B^3 F1 appears to be converging after 100,000 iterations whereas it has not yet converged for the direct sampler.

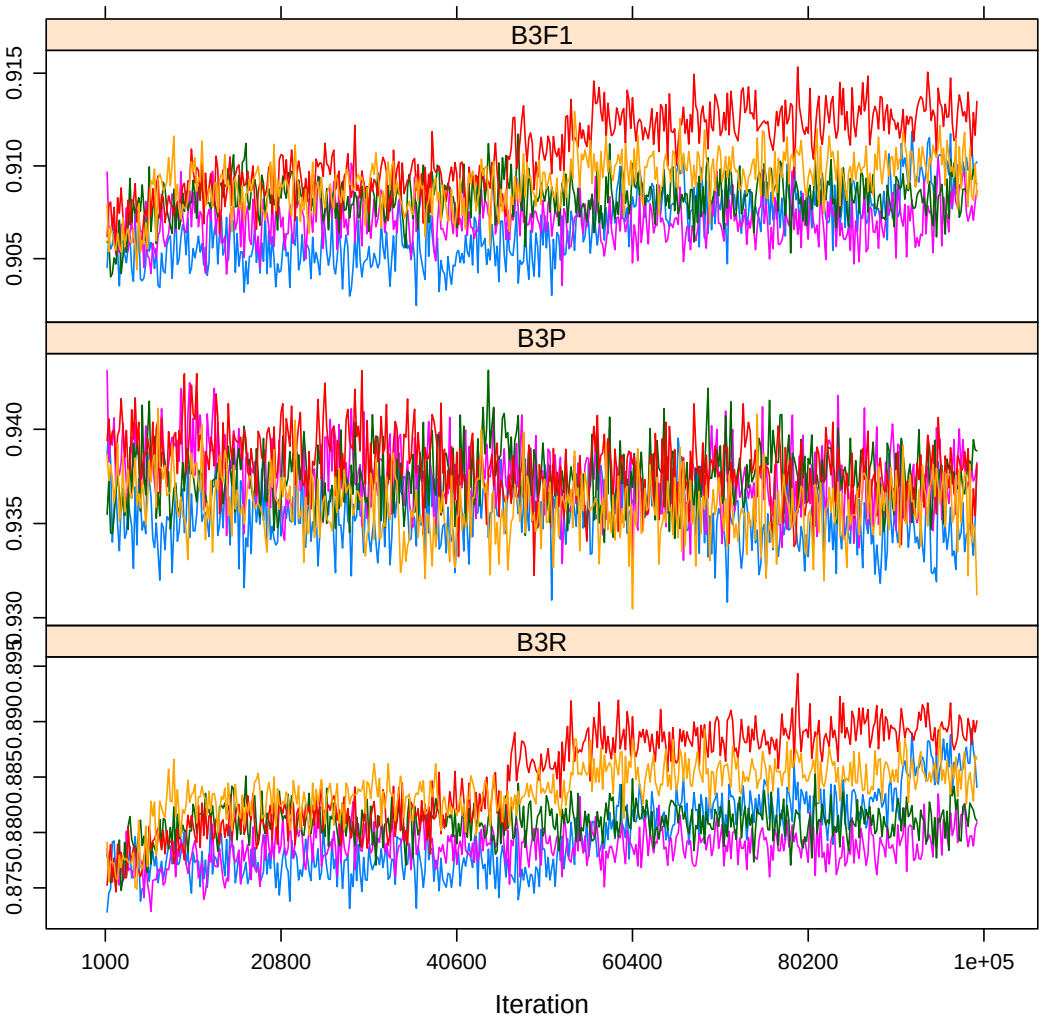
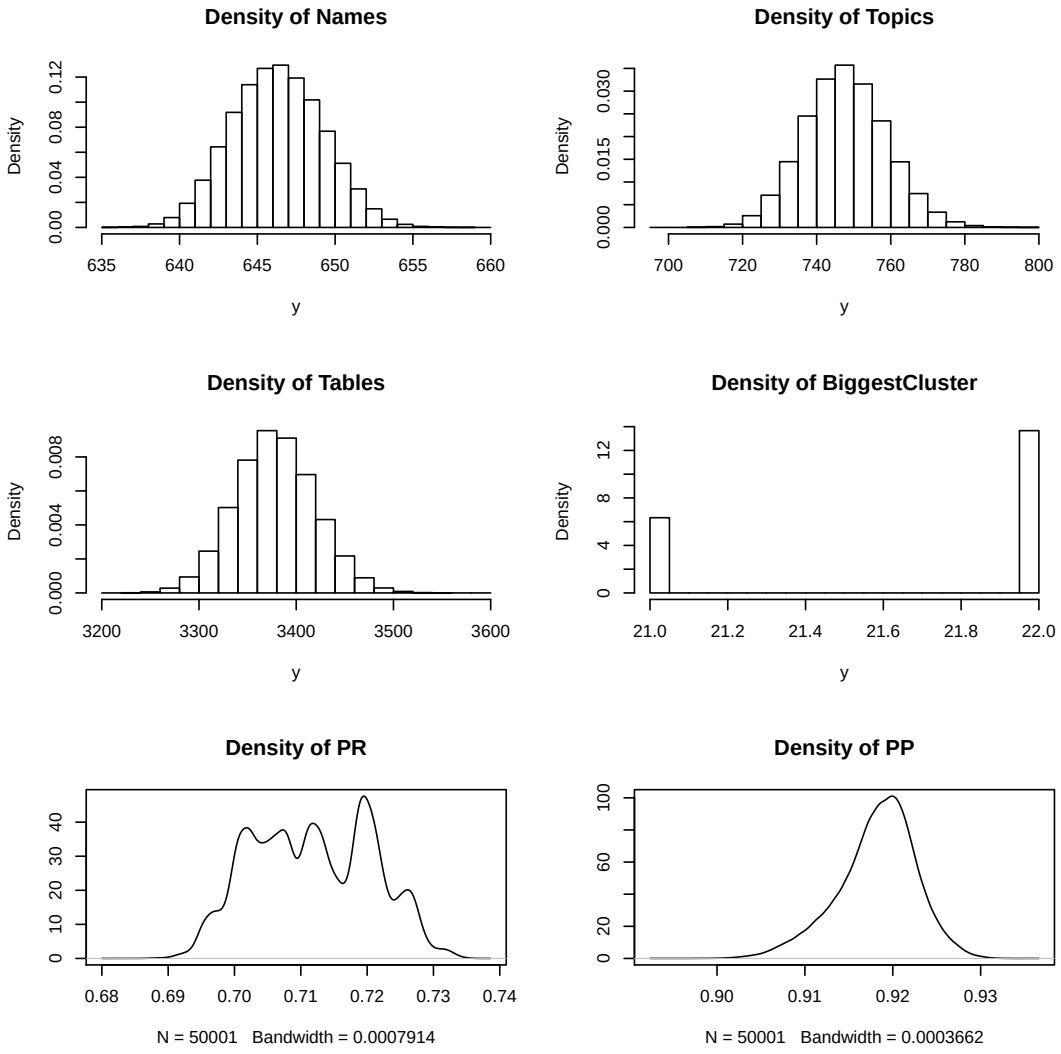
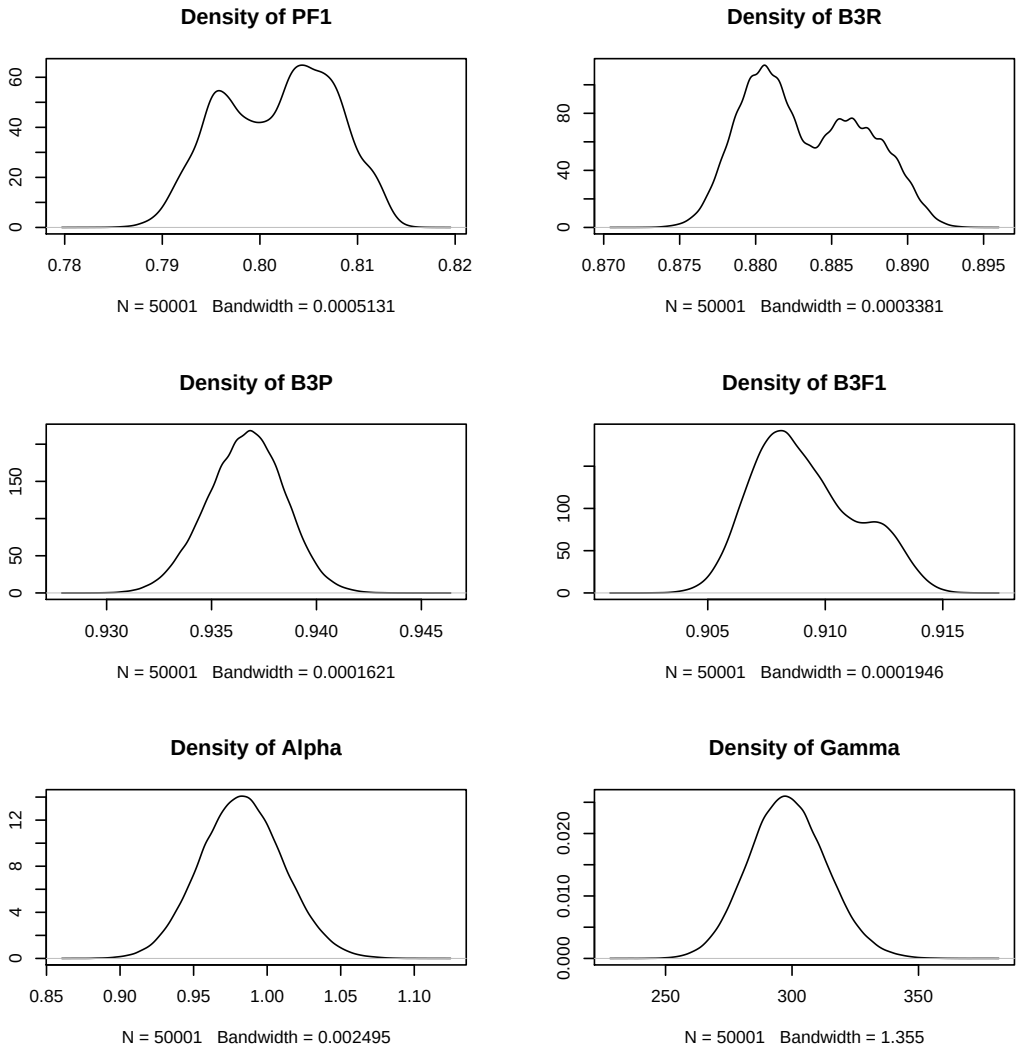


Figure 4.3: A trace plot of the B^3 F1 (B3F1), precision (B3P) and recall (B3R) over 5 chains each with 100,000 iterations. This is for the generative bigram model with the direct sampler. It can be seen that the scores do not change significantly with more iterations though there is a small jump in B^3 F1 after around 50,000 iterations.



(a)

Figure 4.4: First half of density plots of posterior quantities averaged over the last half of 5 chains each with 100,000 iterations. This is for the generative bigram model with the direct sampler. The quantity labels are the same as those in Figure 4.3.



(b)

Figure 4.3: Second half of density plots of posterior quantities averaged over the last half of 5 chains each with 100,000 iterations. This is for the generative bigram model with the direct sampler. The quantity labels are the same as those in Figure 4.3.

ant compared with the direct sampler. For practical purposes, it may still be preferable to use the direct sampler.

Table 4.1: Inference performance of the CRF and direct sampler with BOW trigrams and the generative bigram model. These are measured against the hand-annotated gold standard for the CiteSeer dataset. Means and standard deviations are calculated from samples from the last half of 5 chains, each with 100,000 iterations. The number of entities are clusters that have at least one author allocated to them.

	CRF Trigram	CRF Bigram	Trigram	Bigram
B^3 Recall	93.1 ± 0.3	88.4 ± 0.2	93.0 ± 0.3	88.3 ± 0.4
B^3 Precision	79.3 ± 0.4	93.8 ± 0.2	78.8 ± 0.5	93.7 ± 0.2
B^3 F1	85.7 ± 0.2	91.0 ± 0.1	85.3 ± 0.3	90.9 ± 0.2
Pairwise Recall	80.0 ± 0.6	71.2 ± 0.5	79.7 ± 0.8	71.2 ± 0.9
Pairwise Precision	81.2 ± 0.6	91.6 ± 0.4	80.7 ± 0.8	91.8 ± 0.4
Pairwise F1	80.6 ± 0.3	80.1 ± 0.4	80.2 ± 0.7	80.2 ± 0.6
Number of clusters	516 ± 11	759 ± 11	499 ± 10	748 ± 11
Number of entities	433 ± 5	648 ± 3	429 ± 5	647 ± 3
Time per iteration (s)	1.2	2.8	0.9	2.3

Table 4.2: Inference performance of baseline models on the CiteSeer dataset.

	LDA-ER	Exact string match	Truth
B^3 Recall	84.1 ± 0.7	80.3	—
B^3 Precision	100.0 ± 0.0	100.0	—
B^3 F1	91.3 ± 0.4	89.1	—
Pairwise Recall	68.7 ± 2.3	61.5	—
Pairwise Precision	100.0 ± 0.0	100.0	—
Pairwise F1	81.4 ± 1.6	76.1	—
Number of entities	785 ± 5	817	706

The slower convergence of the direct sampler is somewhat surprising since, in the CRF sampler, the sampling between documents is coupled together. This means that the allocation of entities in a given document is dependent on the counts for tables in other documents being assigned to entities. However, one benefit is that the data points on a table can all be reassigned to a different entity in one step, and this allows for larger

moves in the state space than the individual reallocations in the direct sampler. On the other hand, the coupled sampling in the CRF sampler makes it difficult to extend the model whereas sampling from G_0 allows for more flexible models in the future. The CRF also takes significantly more CPU time per iteration due to the extra level of latent variables it needs to sample.

The generative bigram model for the author names has a significantly better B^3 precision score than the BOW trigrams model, and this is likely due to the generative model being more restrictive in terms of the name variants that can be assigned to a single entity. As a result there is less overclustering and so the generative bigram model has a more reasonable number of inferred author entities. Another contribution to the higher precision score is that the generative bigram model is a better model for character sequences as it takes account of the marginal frequencies of each letter. On the other hand, since the trigram concentration parameter is much higher than the bigram one, there is more smoothing occurring in the BOW trigram model. Hence, the BOW trigram model has a higher recall score though at a large cost to precision.

The BOW trigram model slightly outperforms the generative bigram model using pairwise scoring where larger clusters have a much bigger impact on the score than smaller clusters. Since author entities with many observed name variants must author more documents, the lower recall score for the generative bigram model has a larger effect when using pairwise scoring. This allows the BOW trigram model to have a significantly better recall recall so that its F1 score is higher than that of the generative bigram model.

Due to the better precision, there are more entities in the posterior of the generative bigram model than in the BOW trigram model. During training I also observed even with a range of magnitudes for the trigram concentration parameter, the precision for the BOW trigram model never reached that for the generative model. I also found during training that compared with the BOW trigram model, the generative bigram model is much less sensitive to the concentration parameter. Setting the concentration parameter to other values within two orders of magnitude did not significantly affect the results whereas the results for the BOW trigram model rapidly degraded within one order of magnitude.

Since the CiteSeer dataset contains very few ambiguous names in that the vast majority of identical names refer to only one entity, it is expected that my model does not have a very high precision score. Since I use a uniform prior distribution for the number of clusters, identical names will occasionally be split up into multiple entities because of a strong association with their topics. This effect can be varied by adjusting the prior for γ, which controls the number of clusters across the corpus. In the CiteSeer dataset,

the majority of authors are singleton authors so clustering any of those with another author would lower the precision score. The size of each document can cause author entities to be associated with highly popular topics depending on how well represented an author is in the dataset. As a result, authors that have been associated with popular topics could be more likely to be assigned to other documents' author fields even if they do not share many *n*-grams. This is a result of the Dirichlet process prior and likelihood weighting the words and author names equally for each author entity cluster. This implies that the words and author names need to be weighted separately instead of jointly to prevent the likelihood of the words dominating that of the author names in the author entity clusters.

The results can be compared against a simple baseline of using exact string match to assign identical author names to the same author entity. The results of an exact string match and the leading LDA-ER model applied to this Citeseer dataset can be seen in Table 4.2. This shows that the bigram model improves on the exact string match model whereas the trigram model doesn't. This is also likely due to the generative bigram model being a better model for name variation than the BOW trigram model. The However, even though the bigram and trigram models approach the performance of LDA-ER, since neither of them use the domain specific name variation model in LDA-ER they are not able to outperform it. I also performed preliminary experiments using a simple baseline of clustering together names that were similar to each other by edit distance within a certain threshold and found that this also didn't perform as well as the generative name variation model.

My model successfully assigns similar names to the same cluster along with a reasonable topic as seen in Table 4.3. The table also shows the five words that are most allocated to each entity. Each author entity (separated by a horizontal line) is represented by a set of author names and words from the dataset. The names have various errors such as a word being separated into two words when the small space between characters is detected by an OCR algorithm as a full space. The topic that is inferred for each author contains some function words that are not usually specific to a field such as *recognize*, *study* and *achieve*. However, the author might prefer to use these words to describe their findings over other more general function words so it may help in disambiguating the author. Some words that are specific to the author's field also appear prominently in the author's topic such as *database*, *dynamic* and *vector*, which can disambiguate authors that work in different fields.

Below is an example of author/topic assignments to an abstract authored by *David Poole*, *Randy Goebel* and *Romas Aleliunas* in the CiteSeer dataset. Each of the authors in the author fields was allocated to the correct entity. These allocations are from a single

Table 4.3: Some of the inferred author entities for some name variants and the associated top words drawn from their topic.

Names	Topics
A. Lansky A.L. Lansky Amy L. Lansky	robot, rational, plans, intentions, goals, environment, desire, complex, autonomous, agent
E.H. Shortcliffe E. H. Shortliffe E. Shortliffe	efficient, deal, window, units, sharing, requiring, redundant, reallife, ontologies, map
Henry A. Kautz Henry Kautz H. Kautz	sound, sentence, retrieval, mundane, fundamental, decompose, combining, atomic
R. Kozierok Robyn Kozierok	users, special, set, methods, interactive, good, capable, article, aim, www
A. Moore A. W. Moore	robots, restricted, continuous, applies, algorithm, planning, mazes, twodimensional, non-linear, state-spaces
Klaus–robert Muller Klaus-robert M Uller Klaus-robert Muller	vector, dependent, adopt, generate, achieve
Lj. Trajkovic L. Trajkovi Lj. Trajkovi C Lj. Trajkovi	point, prove, performance, process, recognize

sample of the posterior assignments after 10,000 iterations in the generative bigram model using the direct sampler.

> We provide[3] an introduction[1] to Theorist[1], a logic[1] programming[1] system that uses a uniform[2] deductive[3] reasoning[1] mechanism[1] to construct[1] explanations[3] of observations[3] in terms[1] of facts[1] and hypotheses[1]. Observations[3], facts[1], and possible hypotheses[1] are each sets[1] of logical[1] formulas[1] that represent[1], respectively, a set[3] of observations[3] on a partial[3] domain[3], a set[3] of facts[1] for which the domain[3] is a model[1], and a set[3] of tentative[3] hypotheses[1] which may be required[1] to provide[3] a consistent[1] explanation[3] of the observations[3]. Theorist[1] has been designed[3] to reason[3] in a fashion[3] similar[1] to how we reason[3] with and construct[1] scientific[3] theories[1]. Rather than monotonically[1] deduce[3] theorems[1] from a fixed[3] logical[1] theory[1], theorist[1] distinguishes[3] facts[1] from hypotheses[1] and attempts[3] to use deduction[1] to construct[1] consistent[1] theories[1] for which the observations[3] are logical[1] consequences[1]. A prototype[1], implemented[4] inProlog[1], demonstrates[3] how diagnosis[2], default[1] reasoning[1], and a kind[3] of learning[4] can all be based[4] on the Theorist[1] framework.

The text indicated with a superscript 1 is assigned to the same cluster as the first author who has the inferred name variants *David Poole, D. Poole*. The text indicated with a superscript 2 is assigned to the same cluster as the second author who has the inferred name variants: *R. Goebel, Randy Goebel*. The text indicated with a superscript 3 is assigned to the same cluster as the third author who has the inferred name variants: *R. Aleliunas, Romas Aleliunas*. Finally, the text indicated with a superscript 4 is assigned to a cluster that does not appear in the author list and who has the name: *F. Rossi*. The assignment of words to these clusters indicate that these are function words in this document, especially since the words *implemented, learning, based* likely appear frequently throughout this computer-science based corpus. The unlabelled words were stripped from the dataset, either because they were in the stop word list or they only appeared in one document.

In this document, *Romas Aleliunas* writes using general, non topic-specific words such as *provide*, *set* and *demonstrates*. However, since he only authors one other document in this corpus, the posterior distribution for his topic is quite broad and tends to be over more general areas of expertise. This can be seen in Table 4.4. On the other hand, *David Poole*'s topic in this document is specialised as he writes about them in three other papers in the corpus. This seems a good indication of a topic that he specialises on. His homepage indicates that he works on knowledge representation, computational logic, reasoning about actions, similar to the inferred topic in his cluster. *F. Rossi*, who is not an actual author of the paper but is allocated to some of the words in the document, has a topic that is very broad and covers a number of function and general words used

in the corpus such as *paper* and *problems*. This is because his topic is the topic that is allocated to the most words in the corpus. Thus in this sample and in other documents, his topic serves as a function word topic that is allocated to words that are not specifically related to the topics of the authors for the document. Ideally, this function word topic would only be allocated to general function words and would not be allocated to any authors at all, but building a more complex model with additional restrictions such as that of David M. Blei and Jordan (2003) may slow down mixing. Instead, it is reasonable to assume that for a document, words which are allocated to clusters that are not allocated to any author fields are function words.

Table 4.4: The most probable words sampled from the corpus for each of the clusters that appeared in the example CiteSeer abstract. The last row indicates the number of words allocated to that entity across the corpus.

David Poole	Randy Goebel	Romas Aleliunas	F. Rossi
logical	uniform	set	paper
logic	incorporated	scientific	problems
default	diagnosis	reason	learning
theory		provide	reasoning
theorist		partial	based
hypotheses		observations	algorithms
consistent		fashion	problem
theories		explanations	systems
sets		domain	present
represent		designed	approach
152 words	4 words	35 words	4530 words

Decreasing the α^r hyperparameter in the Dirichlet base measure prior for the n-grams is sufficient to separate out individual author entities into differing clusters. However, this parameter needs to be tuned to specify how similar two names need to be to likely correspond to the same author entity. As $\alpha^r \to \infty$, the probability mass corresponds to an even proportion of all the different n-grams since the effect of large counts is reduced and so a high number of unique n-grams per entity is favoured. However, when $\alpha^r \to 0$, the n-gram counts are squeezed to become zero or one so increasing the effect of large counts. This results in the probability mass favouring a few number of unique n-grams per author entity. As a result, decreasing α^r increases the similarity needed between two strings to be assigned to the same author entity (i.e. *Manuela M. Veloso* and *Manuela Veloso* would not be allocated to the same entity) whereas increasing α^r lowers the

similarity (i.e. *Robert Veloso* and *Manuela Veloso* may be allocated to the same entity). In the given results, α^r is tuned on 10% of the dataset.

Reducing the base DP concentration parameter γ, as described in Section 3.8.4, reduces the number of clusters (author entities and topics) and so can allocate names that are less similar to the same entity. On the other hand, it causes fewer words to be assigned to the actual authors of the document and more of them to be treated as function words, which are assigned to no author.

The document level concentration parameter τ is important in determining the number of tables or number of document-level clusters to which a document is assigned. Ideally, this would be related to the number of real authors in a document. However, extra clusters are used for allocating function words and words not associated with any particular author. In general, experiments suggest setting concentration parameters based on the expected variance for author-topic relationships. For example, to associate author entities with more specific topics, γ should be increased and to associate author entities with more documents, τ should be increased. The experiments I performed with the model used only a small portion of the dataset for training the name variation model and no training was used for the DP concentration parameters, τ and γ. The prior for these concentration parameters were set to result in a relatively uniform prior distribution for the number of clusters.

When using n-grams to model author names, some of the names in my corpus are not clustered correctly. The Dirichlet prior and multinomial likelihood does not capture some intuitions with name matching. Names which have an abbreviated first name and especially those which share the same last name, such as *I. Fischer* and *R Fischer* have difficulties in my method and are commonly allocated together even though they are likely different entities. This is probably a result of *R Fischer* having many more documents in the corpus than *I. Fischer* so *I. Fischer* has too broad of a topic distribution to be identified as a separate entity. This could be mitigated by reducing the variance in the start of name bigrams. An alternative approach is to model the spaces in names separately and utilise domain knowledge that a name is split up into different parts. Using a HDP over the full name and the parts of the name is one way to better model the name variation in different parts of the name. In Section 5.4.2, I describe a domain-specific name variation that solves some of these problems.

4.8.2 *Conflated citation dataset*

The results for the conflated dataset given in Table 4.5 show that the generative bigram model has an even larger improvement over the BOW trigram model than in the

standard CiteSeer dataset. This is because since last names are discarded, the remaining portion of the name is more important for separating out authors and thus two names with minor differences are more likely to refer to two entities rather than being two variants of a given name. In this dataset, the generative bigram model performs significantly better than an exact string match on author names whereas the low precision score for the BOW trigram model means it is not sufficiently separating out entities based on author name. The generative bigram model has higher precision than the exact string match, indicating that topics are successfully being used to separate out authors with identical names to their real entities. Both the trigram and generative bigram models have a better pairwise precision and F1 score than the baseline.

Table 4.5: Inference performance of the direct sampler with BOW trigrams and the generative bigram model. These are measured against the conflated CiteSeer dataset. A burn-in of 5000 iterations were used and the results from the last 5000 iterations averaged together. The number of entities only counts clusters that have authors allocated to them.

	Trigram	Bigram	String match	LDA-ER	Truth
B^3 Recall	79.1 ± 1.1	73.3 ± 1.1	80.8	82.3 ± 0.4	—
B^3 Precision	34.6 ± 0.6	46.9 ± 0.7	42.4	36.4 ± 0.7	—
B^3 F1	48.2 ± 0.6	57.2 ± 0.8	55.6	50.5 ± 0.6	—
Pairwise Recall	60.3 ± 1.8	51.8 ± 2.4	62.6	65.9 ± 1.3	—
Pairwise Precision	11.5 ± 0.5	15.4 ± 0.5	8.7	9.0 ± 0.2	—
Pairwise F1	19.3 ± 0.6	23.8 ± 0.5	15.2	15.8 ± 0.3	—
Number of topics	328 ± 10	466 ± 18	—	—	—
Number of entities	207 ± 3	376 ± 6	362	259 ± 7	706

4.9 CONCLUSIONS

I developed a nonparametric generative Bayesian model, the author-topic space model for author disambiguation. Although there are related models that deal with cross-field dependencies, they do not integrate a name variation model with a model of document text as well as have the benefits of a nonparametric and generative model. This type of model improves flexibility as the model complexity scales with the size of the dataset and also potentially allows use of unlabelled data. The model is hierarchical and succeeds in sharing both topics and their authors among papers. Author entities are inferred that correspond to real author identities and each entity is matched with a unique topic

or distribution over the vocabulary. This allows authors with distinctive styles of writing to be disambiguated with the aid of differences in vocabulary in documents. The model is more general than similar citation matching models by allowing both authors to have aliases and for authors to have a unique distribution over words in a document.

Experiments performed on a set of papers from CiteSeer with gold-standard hand annotations used two different Gibbs sampling methods and yielded good results. The experiments also showed that the CRF sampler, which is a result of integrating out the global random measure G_0, mixed faster than the direct sampler.

A deficiency of this model is that overclustering occurs due to the author names and words being equally weighted in the likelihood. The high dimensionality of the words also result in a much more peaked distribution compared to the authors so that the words dominate in the posterior. An alternative model where the authors and topics are generated using different processes is a different approach that would be interesting to explore, however, sharing and inferring of topics may be more complex in this model. Simply associating authors with multiple topics would not solve the problem since the topics would be overweighted relative to the author names.

Another drawback is that the name variation model is restrictive in terms of the degrees of name variation it can represent since the name model needs to have a conjugate prior for efficient inference. This makes it difficult to use domain-specific name variation models, for example, in languages where names are written using a very small number of symbols from a vast dictionary, such as Chinese, a name variation model based on the Pinyin or strokes of the name is more appropriate. Modeling a name based on the different parts of a name separately, such as the first, middle and last name should also improve performance. The name model also cannot represent names where the first and last names have been switched and other similar problems, which require a more domain-specific model.

The model is also unable to capture the co-occurrence relationships of document co-authors, which can be a strong indicator of an author's identity especially since most papers have more than one author. For example, a graduate student often writes papers with their supervisor as a co-author. Finally, the model also requires associating every word in the corpus with exactly one author. This is needed even when a word is central to the document and would more accurately be modeled as originating from all authors of the document or when a word is a stopword and is not predictive of an author. This contributes to the overclustering problems as described earlier and can also result in some authors being associated with fewer topics than they should be as they are underrepresented in the corpus.

I explore a model in Chapter 5 that is better able to associate topics with authors and addresses some of the drawbacks of the model in this chapter such as the limitations of the name variation model and name overclustering.

For future work, an extension of the model to more accurately model the relationships between authors in a single document could be to have a DP generate only the primary author of a document allowing the modelling of common relationships such as student-supervisor or theorist-experimenter. A more complex model for author names such as a pair hidden Markov model to model string edit distances (Hall, Sutton and McCallum, 2008) would give a higher degree of control over the author clusters. However, these models have the cost of not having a conjugate prior so inference would be relatively inefficient. A dedicated 'function word' author could also be added to each document, which is shared across all documents to explicitly cluster together function words.

5

THE GROUPED AUTHOR-TOPIC MODEL

In this chapter, I present the necessary changes to the model described in Chapter 4 to take advantage of author co-occurrence relationships in documents and to better model author name variants. The new model uses the additional concept of a *latent group* that captures correlations among the latent entities. For each document, an abstract and an author list are generated conditioned on a given group. As in Chapter 4, an author reference, as it appears in an author list, is an author name that may be a variant of the full name. An author *identity* is a real-world individual author and an author *entity* is the latent representation of an author identity in the model. The task is then to infer author entities from a corpus containing documents with author references.

As outlined in Chapter 2, the problem involves associating different references to a real underlying identity. In the case of a corpus, the references are the literal names in the author list for each document. The references may be written in differing forms or using different methods of initialling; thus differing references may refer to the same person (e.g. C. K. I. Williams or Chris Williams). The references may also have changed when transliterating from the original language. Finally, there may be typographical errors in the reference where letters may be exchanged or dropped altogether or misread due to glyphs. As for ambiguity, identical references may refer to different identities. Many people have identical names, and since people rarely use unique identifiers on their papers or homepages, very little information is available to distinguish them.

5.1 INTRODUCTION

The grouped author-topic model I present in this chapter captures author interdependence relations between authors for each document in a corpus, and models the generation of words in documents resulting from this authorial structure.

The model that I develop in this chapter improves on the model in Chapter 4 in a number of ways. It integrates both topic and co-author information for tackling the task of unsupervised identity resolution. Co-author information is captured through a concept of research groups that forms part of the generative model. Topics are distributions over the vocabulary and are described in detail in Section 3.5. Authors and topics are associated with the latent research groups. Each research group also has a number of topics

on which they write. This integration of both topic and research group information enables improved performance over methods that only consider individual information sources. I also use a better name variant model and extend the inference algorithm to handle inference for a wider range of name variant models. Like the model described in Chapter 4, I make no assumptions regarding the equivalence of authors with names that have the same transcription in the corpus. It compares well to the state of the art generative unsupervised models and can both combine different references that refer to the same identity as well as separate identical references that refer to different identities. I associate groups of authors, rather than individual authors, with topics and generate an entire abstract from one group thereby eliminating the difficult problem of matching authors with topics when data is limited.

The remainder of this chapter is set out as follows. In Sections 5.2 and 5.3, I cover the details of the nested Dirichlet process (NDP), the extension of it to the hybrid NDP-HDP and my hierarchical extension of the hybrid NDP-HDP. The NDP (Rodriguez, Dunson and Gelfand, 2008) and hierarchical Dirichlet process (HDP), are both hierarchical extensions to the Dirichlet process (DP), and enable structured forms of sharing for grouped data such as documents. Section 3.8 covers the DP and the HDP in more detail. Section 5.4 develops my framework to tackle this problem with a description of the generative process. In Section 5.5, I describe inference in this framework. I then describe results on toy data in Section 5.6.1 and on real world datasets in Section 5.6.2. I conclude in Section 5.7 with a discussion of the benefits and disadvantages of my framework.

The generative model I describe in this chapter is based on a hybrid NDP-HDP mixture model. In the context of this entity disambiguation problem, a cluster is a set of author names that belong to one entity. The DP suits the fact that an author may write numerous documents in a corpus, and even though a variation of the author's name may not have been observed with that author, it should still have non-zero probability. The HDP then allows entities in each latent group to be shared across the corpus. Two HDPs are used, one for author entities and one for topics, for each group. This allows authors to be allocated to entities and words to topics within the same latent group. The HDP structure with the addition of groups is an extension of the NDP, where a draw from one DP is used as the distribution over an infinite number of other DPs. With the NDP, different documents can share the same distribution over author entities (and the same distribution over topics) whereas in the HDP, different documents would always have different distributions over entities and topics. Thus the NDP allows additional clustering at the document level where two documents may come from the same latent group and so share the same distributions over entities and topics. Finally in the NDP

each author entity can only be a member of one latent group and the hybrid NDP-HDP removes that restriction.

5.2 THE NESTED DIRICHLET PROCESS

Here I set out the nested Dirichlet process (NDP) as proposed by Rodriguez, Dunson and Gelfand (2008). The nested Dirichlet process is a model for grouped data, such as documents, and essentially is a hierarchical model that allows different groups of data to share the same distribution. This avoids the alternatives of assuming each group has the same distribution or that each group has a different distribution. By avoiding this, the model allows information to be shared between the groups, similar to the goal of the hierarchical Dirichlet process (HDP). However, in the HDP, groups do not share distributions only the underlying cluster locations. I will describe the model as applied to a text corpus throughout to make the explanations clearer. To avoid confusion with latent groups, each group of data will be referred to as a document. The observations within each document will be words even though the model can be used with both discrete and numerical data points.

The HDP uses a corpus-level random measure G_0 that is distributed as a Dirichlet process. This corpus-level random measure is used as the base measure for a document-level Dirichlet process from which document-level random measures G_i are drawn for each document. In the NDP, the document-level random measure G_i is instead chosen from an infinite set of random measures G^* so that two documents can share the same random measure.

Assume a set of observations or words x_{ij}, where $i = 1, \ldots, D$ and $j = 1, \ldots, N_i$, are grouped into D documents with words indexed by j within documents indexed by i and with N_i words in document i. Also assume that the words are exchangeable within each document. We represent samples from the Dirichlet process in stick-breaking form consisting of masses (weights) and atoms (cluster locations) as described in Section 3.8.2.

Then a set of distributions F_i, where $i = 1, \ldots, D$, follows a nested Dirichlet process mixture if

$$x_{ij} \sim F_i \tag{5.1}$$

$$F_i(\cdot) = \int p(\cdot|\theta)\, \mathrm{d}G_i(\theta) \tag{5.2}$$

$$G_i(\cdot) \sim \sum_{k=1}^{\infty} \rho_k \delta_{G_k^*(\cdot)} \tag{5.3}$$

$$G_k^*(\cdot) = \sum_{s=1}^{\infty} w_{ks} \delta_{\phi_{ks}}(\cdot) \tag{5.4}$$

where the weights and cluster locations are defined as

$$\phi_{ks} \sim H \qquad \text{(per cluster)} \tag{5.5}$$

$$w_{ks} = u_{ks} \prod_{l=1}^{s-1} (1 - u_{kl}) \tag{5.6}$$

$$u_{kl} \sim \text{Beta}(1, \beta) \tag{5.7}$$

$$\rho_k = v_k \prod_{s=1}^{k-1} (1 - v_s) \tag{5.8}$$

$$v_k \sim \text{Beta}(1, \alpha). \tag{5.9}$$

where α and β are concentration parameters, θ depends on the application, H is the base measure and $\mathbf{w}, \mathbf{u}, \boldsymbol{\rho}, \mathbf{v}$ are positive random variables and correspond to weights from the stick-breaking process. From the stick-breaking process, $\sum_{s=1}^{\infty} w_{ks} = 1$ and $\sum_{k=1}^{\infty} \rho_k = 1$. For each cluster s, the parameters ϕ_s determine the distribution of words from that cluster. Given a cluster, θ is the parameter for the distribution of each word and is equal to one of the ϕ. For example, in a model for continuous observations instead of words, the model for the observations could be a Gaussian distribution with unknown mean and variance where $p(\cdot|\theta) = \text{N}(\cdot|\mu, \sigma^2)$ and $\theta = (\mu, \sigma)$. H could then be a Gaussian-inverse-gamma distribution. In topic modeling, each cluster is referred to as a topic, which is essentially a multinomial distribution over the vocabulary. The graphical model is given in Figure 5.1.

As a result $\mathbb{P}(G_i = G_{i'}) = 1/(1 + \alpha) > 0$ so the probability of two documents sharing a distribution over clusters or topics is non-zero. Any two documents which don't share that distribution also don't share any of the topics. This induces clustering in the distributions for documents, when $G_i = G_{i'}$ for some i and i'. In contrast to the NDP model, in the HDP model, each document has its own distribution over topics

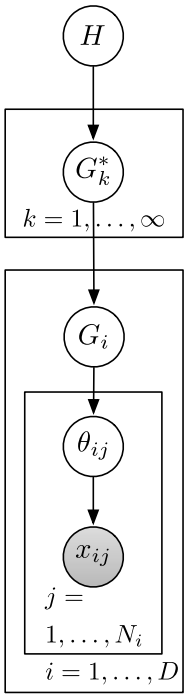


Figure 5.1: A nested Dirichlet process mixture in plate notation. x denotes data points, θ the parameters for the data points, G denotes the document-level random measures, G^* denotes the corpus-level random measures from which the document-level random measure is chosen and H denotes the base measure.

where the topics are shared between all the documents and there is no latent structure to model the case when two documents share the same distribution over topics.

The marginal distribution for the words in each document can be seen to be $G_i \sim \text{DP}(\beta H)$. In the context of the grouped author-topic model, the structure that is induced by the NDP where two documents share the same distribution over topics is advantageous as it can model the case where two documents originate from the same research group, where a research group is a distribution over topics.

The set G_i, where $i = 1, \ldots, D$, which is the mixing distribution, follows a NDP with parameters α, β and H which is written as $\text{nDP}(\alpha, \beta, H)$. * The above model can thus be rewritten as

$$x_{ij} \sim p(x_{ij}|\theta_{ij}) \tag{5.10}$$

$$\theta_{ij} \sim G_i \tag{5.11}$$

$$\{G_1, \ldots, G_D\} \sim \text{nDP}(\alpha, \beta, H). \tag{5.12}$$

We say that two *documents* i and i' are members of the same *cluster* when $G_i = G_{i'} = G_k^*$ in (5.3) for some k so that two documents i and i' share the same distribution and

* An alternative way of writing the NDP is as $G_i \sim Q$, where $Q \sim \text{DP}(\alpha\, \text{DP}(\beta H))$.

are allocated to the same latent cluster k. Two *words* x_{ij} and $x_{i'j'}$ are members of the same *cluster* when $G_i = G_{i'} = G_k^*$ and $\theta_{ij} = \theta_{i'j'} = \phi_{lk}$ in (5.11) for some l so that the words x_{ij} and $x_{i'j'}$ share the same distribution and are allocated to the same lower-level latent cluster l in latent cluster k. The first type of cluster of documents is referred to as the *distributional cluster* and the second type of cluster of words as the *observational cluster* in Rodriguez, Dunson and Gelfand (2008). In the following descriptions, I refer to distributional clusters as research groups and observational clusters as topics to put them in the context of the author-topic modeling problem.

5.2.1 *Issues during inference*

The NDP only allows words to be members of the same cluster when their respective documents are clustered together. This creates problems for inference and restricts the models that can be used to those where clusters need to be partitioned and separated. For inference, this means that when sampling the research group allocation for a document, the topic allocations for all the words within that document must be re-sampled. This is because observational clusters are restricted to one research group. In the grouped author-topic model, this would mean that a topic in the corpus could only be associated with at most one research group, which is a very limiting assumption.

This assumption can slow down mixing and result in a high number of topics since they will be partitioned into different research groups. In addition, many of the topics may be very similar, as the partitioning induced by the research groups prevents the sharing of information between the topics. As a result, a standard collapsed Gibbs sampler is inefficient and instead truncated sampling must be used.

Truncation approximates a DP using a finite-mixture approximation of the Dirichlet process. Truncation puts an upper limit on the number of clusters in the Dirichlet process model so that as long as the posterior number of clusters does not approach that limit, a good approximation to the posterior can be inferred. The good approximation is a result of the geometric drop in cluster weights from the stick-breaking process. The posterior under the truncation converges in distribution to the true posterior as the upper limit tends to infinity. In the NDP, there are two levels of clusters, and for inference to be tractable, the research groups must be truncated. This is called a top-level truncation and results in replacing (5.3) with $G_i(\cdot) \sim \sum_{k=1}^{K} \rho_k \delta_{G_k^*(\cdot)}$ for finite K. As long as K is chosen to be significantly higher than the number of research groups then a good approximation will result. The topics can also be truncated to improve inference performance further. A problem with truncated sampling, however, is that it can be difficult to choose the correct truncation levels (K) in advance.

5.3 THE HYBRID NDP-HDP

In this section, I describe the hybrid NDP-HDP, which generalises upon the NDP. This idea was briefly mentioned in comments made by Lancelot on Rodriguez, Dunson and Gelfand (2008). The hybrid NDP-HDP model allows topics to be members of multiple research groups. This allows for a wider range of models as any words in the whole corpus can be members of the same topics. It also allows for simpler inference with collapsed Gibbs sampling without the need for truncation. This is all achieved by using a draw from a Dirichlet process as the base measure in the NDP. In stick-breaking form, the model is

$$F_i(\cdot) = \int p(\cdot|\theta)\,\mathrm{d}G_i(\theta) \tag{5.13}$$

$$G_i \sim \sum_{k=1}^{\infty} \rho_k \delta_{G_k^*} \tag{5.14}$$

$$G_k^* = \sum_{s=1}^{\infty} w_{ks}\delta_{\phi_s} \qquad \text{(per research group)} \tag{5.15}$$

$$G_0^* = \sum_{s=1}^{\infty} \pi_s \delta_{\phi_s} \tag{5.16}$$

where the weights and cluster locations are defined as

$$\phi_s \sim H \qquad \text{(per topic)} \tag{5.17}$$

$$\pi_s = v_k^* \prod_{s=1}^{k-1}(1 - v_s^*) \tag{5.18}$$

$$v_k^* \sim \text{Beta}(1, \gamma) \tag{5.19}$$

$$w_{ks} = u_{ks} \prod_{l=1}^{s-1}(1 - u_{kl}) \tag{5.20}$$

$$u_{kl} \sim \text{Beta}\left(\beta\pi_l, \alpha\left(1 - \sum_{t=1}^{l}\pi_t\right)\right) \tag{5.21}$$

$$\rho_k = v_k \prod_{s=1}^{k-1}(1 - v_s) \tag{5.22}$$

$$v_k \sim \text{Beta}(1, \alpha) \tag{5.23}$$

where α, β and γ are concentration parameters and H is the base measure. With respect to the NDP, the changes to the formulas are in (5.15) and the additional (5.16) used

to allow topics to be members of multiple research groups. There are also additional stick-breaking weights ($\boldsymbol{\pi}$ and $\mathbf{v}^*$) for each of these topics across the corpus so that $\sum_{s=1}^{\infty} \pi_s = 1$. The changes for the weights are in (5.17), (5.18), (5.19) and (5.21).

The above can be rewritten concisely without using stick-breaking weights as

$$G_0^* \sim \text{DP}(\gamma, H) \tag{5.24}$$

$$\{G_1, \ldots, G_D\} \sim \text{nDP}(\alpha, \beta, G_0^*) \tag{5.25}$$

Since the atoms ϕ in G_k^* are limited to those in G_0^*, words in two documents can still be members of the same topic even if the documents are members of different research groups. In essence, topics can be members of multiple research groups. Documents are clustered together when $G_i = G_{i'} = G_s^*$ for some s so that the two documents i and i' share the same distribution and are allocated to the same research group s. The words x_{ij} and $x_{i'j'}$ can be clustered together when $\theta_{ij} = \theta_{i'j'} = \phi_k$ for some k so that the words share the same distribution and are assigned to the same topic k, even if their documents are not clustered together. The indicators $g_i = s$ and $z_{ij} = k$ will be used when $G_i = G_s^*$ and $\theta_{ij} = \phi_k$ respectively to indicate cluster membership during inference. A cluster then consists of documents with identical g_i values or words with identical z_{ij} values. Research groups will also be referred to as latent groups in the remainder of the chapter.

5.3.1 *Hierarchical extension*

Finally, I propose a hierarchical extension to the hybrid NDP-HDP that can model a document as a mixture of research groups. A hierarchical structure can be used not only for the base measure of the nested Dirichlet process but also for the random measure over the documents. In this case, a document is associated with a mixture of research groups and allows for an even more flexible clustering. Each word is allocated to both

a research group and a topic. Briefly, in stick-breaking form without writing out the weights, this model is written as

$$G_i = \sum_{k=1}^{\infty} \psi_{ik} \delta_{G_k^*} \tag{5.26}$$

$$G_0 = \sum_{k=1}^{\infty} \rho_k \delta_{G_k^*} \tag{5.27}$$

$$G_k^* = \sum_{s=1}^{\infty} w_{ks} \delta_{\phi_s} \tag{5.28}$$

$$G_0^* = \sum_{s=1}^{\infty} \pi_s \delta_{\phi_s} \tag{5.29}$$

where the additional stick-breaking weights ψ are defined similarly to (5.20). Compared to the hybrid NDP-HDP, there is an additional corpus-level DP G_0 and corresponding changes to the model in (5.26) and (5.27). The graphical model is given in Figure 5.2.

To generate each observation in a document, first a research group is sampled and then a topic is sampled conditional on the research group; finally the observation is sampled given the topic's parameters. Allowing a document to be allocated to multiple research groups allows the document to have a broader distribution over observations by combining the different research groups. This can be useful when it may not be the case that a single research group is sufficient to model all the observations in a document.

5.3.2 *Summary*

In summary, the HDP, NDP, hybrid NDP-HDP and hierarchical extension to the HDP-NDP are all models for grouped data and exhibit different sharing properties. The similarities and differences between the mixture models can be seen more clearly through thinking of the topics (atoms or cluster centres) and weights (cluster weights) that are being shared.

For a corpus of documents, each document consists of a variable number of words. The words in each document arise from a mixture, and so each word is allocated to a topic. The distribution of the document can then be thought of as a series of topics and weights from which the observations in the document arise.

Table 5.1 compares and contrasts the HDP and NDP mixture models and their extensions.

The model I describe in Section 5.4 is a model based on the hybrid NDP-HDP mixture model with modifications for author-topic modelling with latent groups. The main

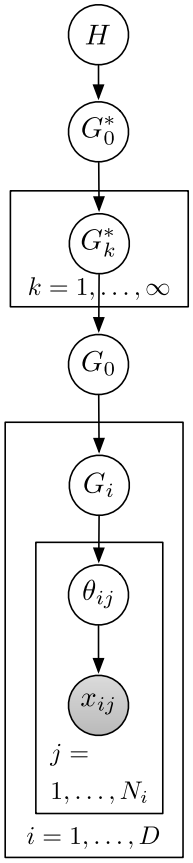


Figure 5.2: A hierarchical extension of the hybrid NDP-HDP mixture in plate notation. See Figure 5.1 for the notation. The additional notation G_0^* is the corpus-level shared distribution over topics and G_0 is the corpus-level shared distribution over research groups.

Table 5.1: A comparison of different HDP and NDP mixture models and their extensions. The observation source indicates where the observation (or word) originated from or was allocated from. A ? indicates that the property depends on the observation's research group allocation.

Model	Observation source	Topics shared	Topic weights shared
HDP mixture	Same document	Yes	Yes
	Different documents	Yes	No
NDP mixture	Same document	Yes	Yes
	Different documents	?	?
	Same research group	Yes	Yes
	Different research groups	No	No
Hybrid NDP-HDP mixture	Same document	Yes	Yes
	Different documents	Yes	?
	Same research group	Yes	Yes
	Different research groups	Yes	No
Hierarchical hybrid NDP-HDP mixture	Same document	Yes	?
	Different documents	Yes	?
	Same research group	Yes	Yes
	Different research groups	Yes	No

change is for modelling the two types of observations, words and authors, by mirroring the NDP-HDP structure. This model's graphical model is Figure 5.4b. I also perform experiments with my hierarchical extension to the hybrid NDP-HDP mixture model where a document is allocated to multiple research groups and group allocations are made at the word level. This model's graphical model is Figure 5.4a.

5.4 THE GROUPED AUTHOR-TOPIC MODEL

In this chapter, I aim to use as much of the commonly-shared information as possible that is available for the purposes of entity resolution. This information is typically the words in a title and abstract, as well as the author lists. This information is organised via the concept of a research group (which characterises which authors might be co-authors) along with topic information associated with each group (which helps disambiguate authors which could be members of a number of research groups). This leads to a model which I call the grouped author-topic model. The rest of this chapter will refer to a latent group of authors and topics as a group.

Each real-world author identity will be represented by a latent author entity. Although a single entity, a real-world author may have a number of different names by which he or she is referred. These are known as references or variants, as described in Chapter 4. Different variants of the author's name occur due to variation in initialling, transcription errors, typographical errors, transliteration differences etc. These varying forms can be viewed as being generated by a name variation process which, for each author, modifies an underlying canonical name associated with that particular author. I tested a nonparametric generative *n*-gram model and a bag-of-words *n*-gram model in Chapter 4, and in this chapter, I test a previously-used domain-specific name variation model (Bhattacharya and Getoor, 2006). This is a string metric based model that uses domain knowledge that author names are often written with first or middle names initialled or middle name removed to calculate the probability of a name corruption or name variant.

5.4.1 *Combining modalities using groups*

The problem being tackled is an unsupervised problem, and we wish to combine different modalities of data (author name and topics). Hence we need to combine the likelihood of the author entities with the likelihoods of the topics. This makes the problem similar to that of image annotation (David M. Blei and Jordan, 2003) in which the problem is to combine image region and topic likelihoods in the same model. However,

models such as the author-topic model and the model in Chapter 4 are models where entities are directly assigned to individual words in a document. This causes problems in these models because in the posterior distribution of the authors, authors that are assigned to many words will have a posterior that tends to be dominated by the topic assignments. As a result, the evidence from the author names themselves is missed.

I explore several models for reducing the effect of abstract length on authors in Section 5.4.3. The models explored in this chapter also allow the posterior distribution over entities conditional on the group to be independent of the length of text in each document. This is a result of associating topics with groups rather than individual entities.

An alternative model is to have topics be associated with entities and entities be associated with groups. Since this model uses DPs where the cluster weights are independent of whether the observation is a word or an author, known as single-p dependent DPs, the posterior weight of an entity will be equally affected by the likelihood of all observations rather than just those associated with authors or just words (Griffin and Steel, 2006; MacEachern, 1999). The result of this is that the word counts in the posterior would be overweighted relative to the author counts, and so topics would have an overly large effect compared with author names for the entities.

5.4.2 *Name variation model*

The domain-specific name variation model (Bhattacharya and Getoor, 2006) is similar to a string metric and calculates the probability of one string being transformed from another string. The model uses domain knowledge that names are commonly split into three different parts (first, middle and last name) and that the first and middle names are often initialled to assign different errors in different parts of the name varying levels of significance. This ensures a better model of author name variation than simple string based methods that do not distinguish name parts, such as the ones described in Chapter 4.

The model gives the probability of an author's canonical or full name being transformed to a reference name. In this model, first, middle and last names are assumed to be independent. The first name is initialled with probability p_{FI}, missing with probability p_{FD} and fully kept with probability p_{FR} where $p_{\mathrm{FI}} + p_{\mathrm{FD}} + p_{\mathrm{FR}} = 1$ and similarly for the middle name. The first and middle initials can also have the wrong initials with a certain probability. Since the last name is always present, it is modelled by characters being inserted with probability p_{I}, replaced with probability p_{R} and deleted with probability p_{D}. The minimum number of character insertions, n_{I}, replacements, n_{R}, and deletions, n_{D}, needed to transform the author's canonical name to the reference name

is calculated using a minimum edit distance algorithm. The probability of the author's referenced name given the author's canonical name is then $p_{\mathrm{I}}^{n_{\mathrm{I}}} p_{\mathrm{R}}^{n_{\mathrm{R}}} p_{\mathrm{D}}^{n_{\mathrm{D}}}$.

Compared to the name variant models in Chapter 4, this domain-specific model does not have a conjugate prior so inference is more complicated. This model is also domain-specific whereas the string-based models can be used in a wide range of areas.

5.4.3 *Model description*

To describe the model I need to introduce two concepts, that of group and that of topic. The idea of topic is described in Section 3.5, where a topic is a mixture component defining a distribution of words. The content of an individual abstract will only contain a small number of topics out of the total possible number. Intuitively, the idea of a group conceptualises authors who work/publish together and the associated topics on which they publish. The number and frequency of each group is not fixed but is given by a stick breaking prior from the GEM distribution, which is defined in Section 3.8.2.

For each particular group, we sample from a Dirichlet process over author entities (to capture the authors that work together), and over topics (to capture the topics the group publishes on). This Dirichlet process is hierarchical with a base measure based on a draw from a global author and topic DP. Hence author entities and topics can be shared between groups so that an author entity has non-zero probability of occurring in multiple groups, and similarly for the topics.

The grouped author-topic model can be written in stick-breaking form and is given as a graphical model in Figure 5.4b.

$$\phi_s^a \sim H^a \qquad \phi_t^w \sim H^w \tag{5.30a,b}$$

$$E_0 = \sum_{s=1}^{\infty} \pi_s^a \delta_{\phi_s^a} \qquad T_0 = \sum_{t=1}^{\infty} \pi_t^w \delta_{\phi_t^w} \tag{5.31a,b}$$

$$E_k = \sum_{s=1}^{\infty} \beta_{ks}^a \delta_{\phi_s^a} \qquad T_k = \sum_{t=1}^{\infty} \beta_{kt}^w \delta_{\phi_t^w} \tag{5.32a,b}$$

$$G_g \sim \sum_{k=1}^{\infty} \rho_k \delta_{E_k \times T_k} \tag{5.33}$$

where $\boldsymbol{\pi}$ and $\boldsymbol{\beta}$ denote the weights from the stick-breaking construction (which is described in Chapter 3) for the draw from the entity-level DP (superscript a) and the topic-level DP (superscript w). $\boldsymbol{\rho}$ denotes the weights for the group-level DP. k ranges over the groups and G_g is the mixing distribution for document g. E and T denote the ran-

dom measures for the entities and topics respectively. H^a denotes the base measure over the author entities and H^w denotes the base measure over topics. The correspondence between this model and the hybrid NDP-HDP can be seen by matching up the formulas, (5.30a,b) matches with (5.17), (5.31a,b) matches with (5.16), (5.32a,b) matches with (5.15) and (5.33) matches with (5.14).

To complete the generative model, I need to describe the process of generating the actual abstracts. Each abstract is associated with a group (again drawn from a multinomial drawn from the group-level random measure). The group associated with the document is used to determine which authors are potentially represented in a document and which topics are written about (i.e. those given significant probability by the associated group). Intuitively, this can be thought of as a document being authored by a single research group, which has a number of particular topics which they may choose to publish on and which may be represented in the current document. Each group is associated with an entity random measure and a topic random measure, which results in a group having a distribution over topics and entities, and this gives rise to a nested structure. Note that, under this model, only one group can contribute to a given document. A variant of this model is to allow multiple groups to contribute to a document as in the case of large-scale collaborations.

In the hybrid NDP-HDP mixture model, clusters can be members of multiple groups so that an author entity may be a member of multiple groups. For ease of understanding, I will subsequently represent the random measure over groups for each document in terms of its stick-breaking weights, ρ_k, which weight each of the group-level random measures over entities and topics (E_k, T_k).

All the author entities and topics in the document are generated conditional on this group. Since there is no knowledge a priori of how many components are in the group, then a DP mixture model is an appropriate model.

Intuitively, a research group has a number of authors who share interests. In the grouped author-topic model, in contrast to the author-topic model, the authors themselves are not associated with topics directly. As an illustrative example in Figure 5.3, the authors D. Blei and T. Griffiths are members of the same group and so it is likely that they co-author papers together. Since the co-authors and topics are so different between the two groups, then given more examples, it is reasonable to infer that the two T. Griffiths are different people. In the grouped author-topic model, $H^w = \text{Dirichlet}(\eta)$ denotes a prior distribution for the topic parameters where η is usually the parameter for a symmetric Dirichlet distribution. A topic is parameterised by a distribution over the vocabulary of the corpus. Since this is conjugate to the likelihood (a multinomial distribution), during inference θ^w can be integrated out. For the author name variation

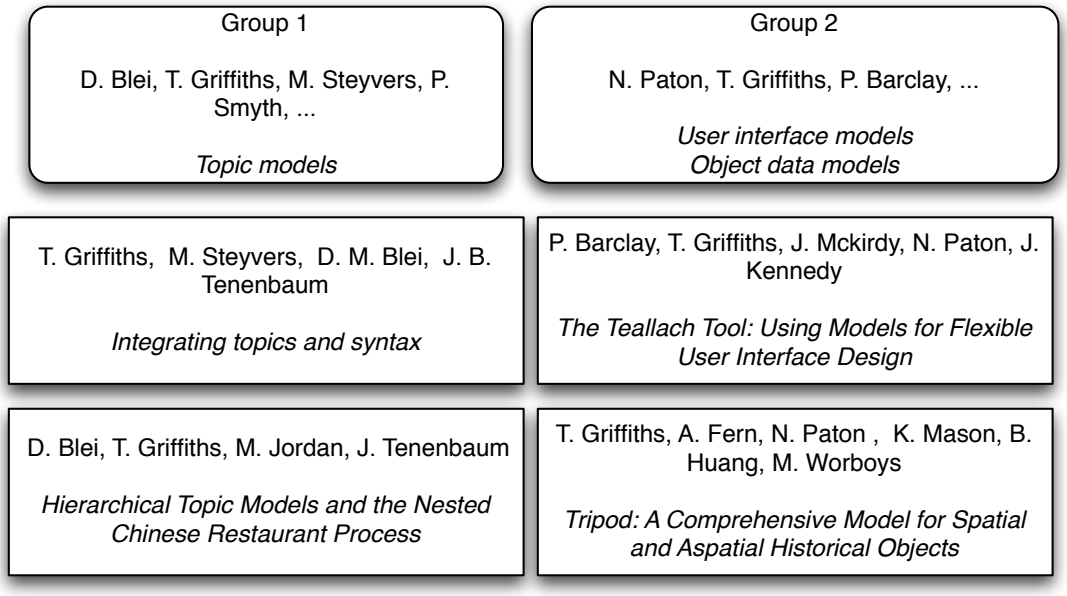


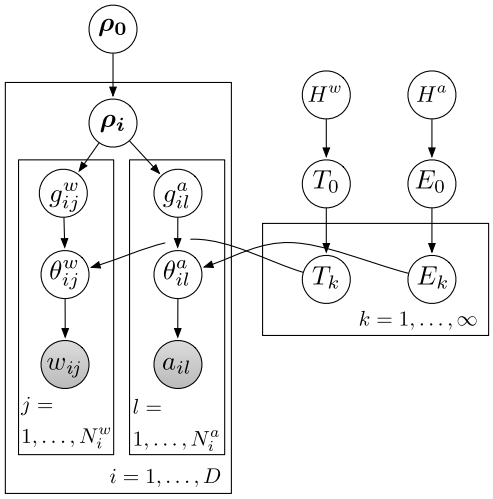
Figure 5.3: An example of four papers from CiteSeer that were generated by the two groups of authors above them. As can be seen from the list of co-authors and topics in each group, the T. Griffiths in each group are likely to be different people.

model, there are multiple possible models but I use a relatively simple generative process for changing an author name given its true name as described in Section 5.4.2.

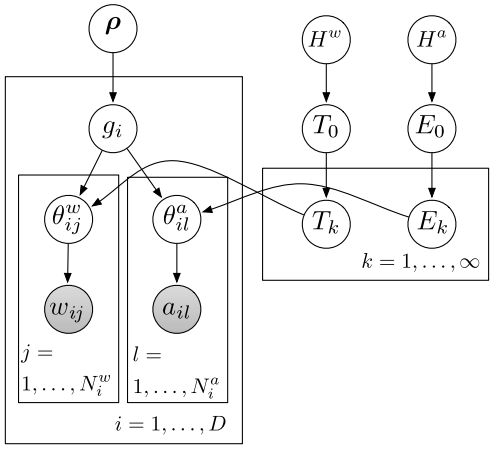
The generative process for a whole corpus is as follows, where γ and α denote concentration parameters for the global and lower level DPs respectively, the superscripts w and a denote the parameters or distributions for the topics and the author entities respectively. H^w denotes the base measure for the topics and H^a denotes the base measure for the author entities. GEM represents the distribution from the stick-breaking construction. ρ_k denotes the weight from the stick-breaking construction for each group $k = 1, \ldots, \infty$. These weights determine the group random measure over entities E_k and topics T_k. Finally, θ denotes the parameters for the likelihood models for the authors and topics and $f(a|\theta^a)$ is the probability the name reference a is a variant of the canonical (true) name θ^a under the name variation model. For brevity, I do not write out the conditioning of $f(\cdot)$ on the name variation model parameters $(p_\mathrm{I}, p_\mathrm{R}, p_\mathrm{D})$.

1. Draw (from their prior distributions) the concentration parameters for the global DPs, γ^w, γ^a, τ for the topics, authors and groups respectively. Likewise, draw the concentration parameters for the lower-level DPs, α^w, α^a from their priors.

2. Draw a global distribution over topics $T_0 \sim \mathrm{DP}(\gamma^w, H^w)$ and author entities $E_0 \sim \mathrm{DP}(\gamma^a, H^a)$. Draw a distribution over groups $\boldsymbol{\rho} \sim \mathrm{GEM}(\tau)$.



(a) A model with separate research groups collaborating and authoring a document.



(b) A unified research group, where all components of a document belong to one research group.

Figure 5.4: The grouped author-topic model in plate notation, where the number in the corner of the plate indicates the number of times it is replicated. The observed variables are the words, w, and author references, a. D is the number of documents in the corpus, N_i^w and N_i^a are respectively the number of words and authors in document i, k ranges over the groups. H denotes the base measures and θ denote the topic and author name parameters for an observation. $\boldsymbol{\rho}$ denotes the weights for the latent groups from the stick-breaking construction. g denotes the group allocation variable. E and T denote the random measures for the entities and topics respectively. The concentration parameters for the DPs have been omitted for clarity.

3. For each group k, draw a distribution over topics $T_k \sim \text{DP}(\alpha^w, T_0)$ and author entities $E_k \sim \text{DP}(\alpha^a, E_0)$.

4. Now for each document i,

   a) Draw a group to generate the document $g_i \sim \boldsymbol{\rho}$.

   b) For each word w_{ij},

      i. Draw a topic $\theta^w_{ij}|g_i \sim T_{g_i}$.

      ii. Draw a word $w|\theta^w_{ij} \sim \text{Multinomial}(\theta^w_{ij})$.

   c) For each author name a_{ij},

      i. Draw an author entity $\theta^a_{ij}|g_i \sim E_{g_i}$.

      ii. Draw a (possibly modified) author's name from the name variation model $a|\theta^a_{ij} \sim f(\theta^a_{ij})$.

I also explore a similar model where each data point in the document is separately associated with a group drawn from a document-level random measure over groups. In this model, words and authors in a document can be allocated to different research groups and are not restricted to one group. To investigate if there are significant differences to assigning multiple research groups to a document, I also perform experiments on this model, my hierarchical extension of the hybrid HDP-NDP. In this model, a document no longer needs to be associated with one research group. A hierarchical structure is used for groups and every document is composed of a mixture of groups.

5.5 INFERENCE

Since calculating the exact posterior under DP models is intractable as described in Section 3.8.6, I use approximate algorithms. Due to having a non-conjugate base measure and the ease of implementing and verifying a MCMC approach, I use collapsed Gibbs sampling based on the Pólya urn scheme for inference in this model.

Inference can be done using a Chinese restaurant process representation similar to that described by Teh, Jordan et al. (2006) and involves Gibbs sampling while integrating over conjugate distributions and the random measures E_k, T_k, G. E_0 and T_0 are instantiated and sampled from. I avoid integrating over the global random measures, E_0 and T_0, since as shown in Chapter 4, doing this can slow down mixing of the Markov chain as it would couple the sampling of multiple groups together.

For inference, the group allocations can be sampled given the word and author allocations, and then the word and author allocations can be sampled given the group. As

noted earlier, I integrate out the parameters for each topic, which are the multinomial distributions over the words.

The true names in the corpus are represented in latent variables in the grouped author-topic model. However, for practical purposes, to avoid the search over all possible canonical names, I make the computationally simplifying assumption that the true name can be sufficiently well represented by one of the references in the corpus. Every unique author name that appears in the corpus is, therefore, given an equal probability, η^a of being the canonical name for an entity so that $H^a = \text{Multinomial}(\eta^a)$. This is equivalent to using an empirical prior for the space of canonical names. However, in reality it is unlikely for every unique name in the corpus to be equally likely to be a canonical name since it is known that longer names are more likely to be full canonical names. It is also probable that many canonical names do not exist in the corpus and ideally these should be represented in H^a. This is because H^a represents the probability of any author entity having a particular canonical name. Therefore, if H^a only put non-zero probability on names observed in the corpus, then that assumes many entities in the corpus will share the same name.

As a result, to fit the full name variant model, η^a is learnt on a small training corpus and a non-zero probability is put on names that are unobserved in the corpus. To ease computation, these unobserved names will never be assigned as the canonical name of an author in the corpus. In effect, this serves to reduce the number of final entities in the model that share identical names and is necessary due to the use of this name variant model and discrete base measure. In addition, since the base measure for the author names is not conjugate to the name variation model, I use the non-conjugate auxiliary variable sampling algorithm as described in Section 3.8.6 for the entity allocations. This allows the name parameters for all names that are assigned to a single entity to be resampled in a single step.

w_{ij} denotes the jth word in the ith document and a_{ij} similarly denotes the jth author reference. Latent indicator variables are used to make inference easier. g_i denotes the group assigned to the ith document. z_{ij} denotes the author entity from which a_{ij} is drawn and t_{ij} denotes the topic from which w_{ij} is drawn. So that $z_{ij} = z_{i'j'} = k$ when $\theta_{ij} = \theta_{i'j'} = \phi_k$. γ and α denote the concentration parameters as described in the previous section. $\boldsymbol{\phi}$ denotes the parameters for the distribution over words conditional on each topic or over author references conditional on each author name. The conditional distributions used for Gibbs sampling are given below.

The full process for sampling from the conditional posterior is enumerated below. n^g_s is a count for the number of documents allocated to group index s. n^a_{iks} is a count for the number of authors in the ith document being allocated to entity index k and the group

index s. n^w_{its} is a count for the number of words in the ith document being allocated to topic index t and group index s. Finally, n^v_{tw} is a count for the number of w words allocated to topic index t. In addition, $\cdot$ indicates marginalising over an index so that

$$n^w_{\cdot\cdot s} = \sum_{i=1}^{D}\sum_{t=1}^{N_T} n^w_{its} \qquad n^w_{i\cdot\cdot} = \sum_{t=1}^{T}\sum_{s=1}^{N_G} n^w_{its} \tag{5.34}$$

$$n^w_{it\cdot} = \sum_{s=1}^{N_G} n^w_{its} \qquad n^w_{\cdot ts} = \sum_{i=1}^{D} n^w_{its} \tag{5.35}$$

$$n^a_{\cdot ks} = \sum_{i=1}^{D} n^a_{iks} \qquad n^a_{ik\cdot} = \sum_{s=1}^{N_G} n^a_{iks} \tag{5.36}$$

$$n^v_{t\cdot} = \sum_{w=1}^{W} n^v_{tw} \tag{5.37}$$

where W is the size of the corpus vocabulary, N_T is the current number of instantiated topics and N_G is the current number of instantiated groups. A superscript $-ij$ indicates that particular observation should be ignored. β_k is the stick-breaking weight for the respective component in the higher-level random measure and β_{new} is the weight for a new component. ϕ^a_k is the parameter for author entity k.

1. For each document i,

    a) Sample the group allocation g conditional on the topics and names in the document. This indicates the group from which the document is generated. After removing the current allocation g_i from the counts n, the conditional distribution is

$$p(g_i = s|\mathbf{g}^{-i}, \mathbf{z}, \mathbf{t}) \propto \begin{cases} n^g_s \frac{\Gamma(n^w_{\cdot\cdot s}+\alpha^w)}{\Gamma(n^w_{i\cdot\cdot}+n^w_{\cdot\cdot s}+\alpha^w)} \prod_u \frac{\Gamma(n^w_{iu\cdot}+n^w_{\cdot us}+\alpha^w\beta^w_u)}{\Gamma(n^w_{\cdot us}+\alpha^w\beta^w_u)} \prod_k \frac{\Gamma(n^a_{ik\cdot}+n^a_{\cdot ks}+\alpha^a\beta^a_k)}{\Gamma(n^a_{\cdot ks}+\alpha^a\beta^a_k)}, & \text{if } s = g_{i'} \text{ for some } i \neq i' \\ \tau \frac{\Gamma(\alpha^w)}{\Gamma(n^w_{i\cdot\cdot}+\alpha^w)} \prod_u \frac{\Gamma(n^w_{iu\cdot}+\alpha^w\beta^w_u)}{\Gamma(\alpha^w\beta^w_u)} \prod_k \frac{\Gamma(n^a_{ik\cdot}+\alpha^a\beta^a_k)}{\Gamma(\alpha^a\beta^a_k)}, & \text{otherwise.} \end{cases} \tag{5.38}$$

    where u indexes into the topics and k indexes into the entities. The values for u where the topic is not associated with words in the current document and values for k where the entity is not associated with references in the current document can be ignored to improve performance. The probability of sampling a group is proportional to the number of times that group has been chosen for an author or topic in the corpus.

b) For each author a_{ij}, sample the entity allocation z for each author name. After removing the current allocation z_{ij} from the counts n, the conditional distribution is

$$p(z_{ij} = k|\mathbf{z}^{-ij}, \mathbf{g}) \propto \begin{cases} (n^a_{\cdot kg_i} + \alpha^a \beta^a_k) f(a_{ij}|\phi^a_k), & \text{if } k = z_{i'j'} \text{ for some } (i'j') \neq (i,j) \\ \alpha^a \beta^a_{\text{new}} f(a_{ij}|\phi^a_{\text{new}}), & \text{otherwise} \end{cases} \quad (5.39)$$

The entity is chosen proportional to the number of times that entity is associated with the group that this author is currently assigned to. A new canonical name ϕ^a_{new} is sampled from H^a, however, when the existing allocation z_{ij} is not shared with any other data points (i.e. z_{ij} is a singleton), ϕ^a_{new} is set to be equal to $\phi^a_{z_{ij}}$.

When a new entity k^{new} is sampled, then also draw $b \sim \text{Beta}(1, \gamma^a)$, set the new weight $\beta^a_{k^{\text{new}}} = b\beta^a_{\text{new}}$ and set the new β^a_{new} to $(1-b)\beta^a_{\text{new}}$. b corresponds to the weight of the new atom that is instantiated from the Dirichlet process.

c) For each word w_{ij}, sample the topic allocation. After removing the current allocation t_{ij} from the counts n, the conditional distribution is

$$p(t_{ij} = k|\mathbf{t}^{-ij}, \mathbf{g}) \propto \begin{cases} (n^w_{\cdot kg_i} + \alpha^w \beta^w_k) \frac{n^v_{kw_{ij}} + \eta}{n^v_{k\cdot} + W\eta}, & \text{if } k = t_{i'j'} \text{ for some } (i'j') \neq (i,j) \\ \alpha^w \beta^w_{\text{new}} \frac{1}{W}, & \text{otherwise} \end{cases} \quad (5.40)$$

Similarly, the topic is chosen proportional to the number of times that topic is associated with the group that this word is currently assigned to. If a new topic is sampled in this step then draw the new weight for the topic as for the authors.

d) Sample the document-level cluster counts via the Chinese restaurant process where m_{ik} represents the number of 'tables' in document i allocated to cluster k, and accounts for the possibility that different allocations at the document-group level can be assigned to the same entity at corpus level. This represents the extra level of clustering induced by the HDP mixture model at the document level. Sample m^a conditional on the entity and group. Sample m^w similarly conditional on the topic and group.

Each m count can be sampled by generating $n_{i\cdot k}$ uniformly distributed random variables $u_1, \ldots, u_{n_{i\cdot k}}$ in $[0, 1]$, so that:

$$m_{ik} = \sum_{m=1}^{n_{i\cdot k}} \mathbf{1}\left[u_m \geq \frac{\alpha\beta_k}{\alpha\beta_k + m}\right]. \tag{5.41}$$

where $\mathbf{1}$ is the indicator function.

2. Sample the canonical string for each author entity:

$$p(\phi_k = S|\mathbf{z}) \propto \prod_{z_{i'j'}=k} p(a_{i'j'}|S) \tag{5.42}$$

where $p(a_{i'j'}|S)$ is calculated according to the name variation model. For alternative models, any likelihood can be used here.

3. Sample $\boldsymbol{\beta}^a$ from $(\beta_1^a, \ldots, \beta_K^a, \beta_{\text{new}}^a) \sim \text{Dirichlet}(m_{\cdot 1}^a, \ldots, m_{\cdot K}^a, \gamma^a)$ where K is the current number of instantiated author entities. Sample $\boldsymbol{\beta}^w$ equivalently.

The concentration parameters are updated by sampling from their posteriors after every round of Gibbs sampling following the methods in Section 3.8.4 and Section 3.8.7 adapted to sample different parameters for the entities, topics and groups following their respective numbers of tables.

In identity resolution problems, the mode of the posterior can be more interesting than the full posterior distribution that is sampled by the Gibbs algorithm. The simulated annealing schedule of Haghighi and Klein (2007) can be used, which samples entities proportional to the entity posteriors exponentiated according to the current cycle number. At the end of the preset number of cycles, a mode is found. In my case, I am calculating the expected author identity for each author reference and the full posterior distribution will be more useful as it can describe the confidence in the number of entities, groups and topics.

5.6 EXPERIMENTS

The best parameters for the name variation model were found with a parameter sweep on a 10% subset of the CiteSeer dataset. The model was then evaluated on both a small set of generated toy data and on real-world datasets.

The diagnostic methods by Gelman and Rubin as described in Section 3.4.1 were used to assess whether the Markov chains have converged.

Table 5.2: Mean values and standard deviations of the posterior quantities for the toy dataset across 10 chains, each with 100,000 iterations and discarding the first half of the chains.

Names	Groups	Topics	Pairwise F1	B3 F1
8.2 ± 1.0	3.0 ± 0.1	14.7 ± 2.5	91.4 ± 0.05	94.2 ± 0.03

5.6.1 *Toy data*

I initially tested the model on a synthetic corpus by checking the model posteriors. The synthetic corpus was created using the generative procedure described in the previous section and consisted of 50 documents, generated from 8 authors, 3 research groups, 9 topics and a vocabulary of 100 words. Two of the authors have exactly the same names. Each author name had between 0 and 2 name variants. Each document consisted of 60 words and 2 authors drawn from one research group. A Gamma(1, 0.1) prior was placed on α^a while the other concentration parameters were given Gamma(1, 1) priors, η set to 0.1. The Gibbs sampler was used to infer the parameters used to generate the corpus. The concentration parameters were updated by sampling from their posteriors during inference.

Table 5.2 shows the posteriors for the inferred number of authors, groups and topics. Figure 5.5 shows the posterior density of various inferred quantities for this toy problem. The final inferred number of names and research groups is close to those used during data generation; however, the number of topics is slightly overestimated. This indicates that inferring the true number of topics is difficult due to the high dimensionality of the problem, especially as each topic has its own distribution over 100 words to be estimated.

The plot of the scale reduction factor over a number of iterations in Figure 5.6 show that the number of names converges by around 2,000 iterations. The scale reduction factors for the other posterior quantities also indicate that the chain has mixed within 2,000 iterations, as best as these diagnostics can assess.

5.6.2 *Citation dataset*

To experiment with real-world data, I tested the grouped author-topic model on the author lists and abstracts from several publicly available citation datasets. Since their ground truth is publicly available, I chose the the real-world CiteSeer and Rexa databases. The CiteSeer dataset was created by Giles, Bollacker and Lawrence (1998) and

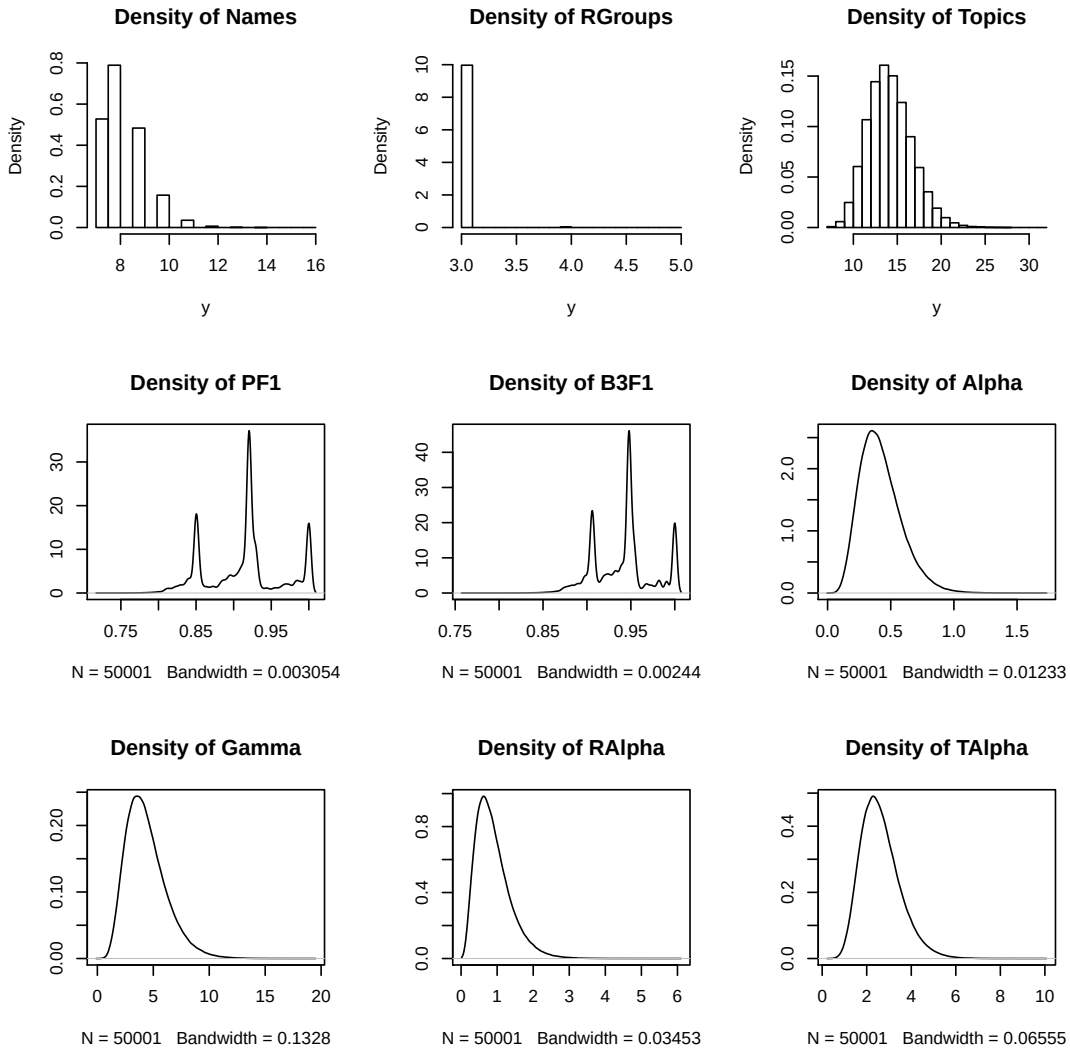


Figure 5.5: A density plot of the posterior quantities for the toy dataset. *Names* is the number of inferred author entities, *RGroups* is the number of inferred latent groups, *topics* is the number of inferred topics, *PF1* is the pairwise F1 score, *B3F1* is the B^3 F1 score, *Alpha* and *Gamma* are the values of α^a and γ^a, while *RAlpha* is the value of τ and *TAlpha* is the value of α^w. The inferred number of names and research groups is close to that used to create the dataset, however the number of topics is slightly overestimated.

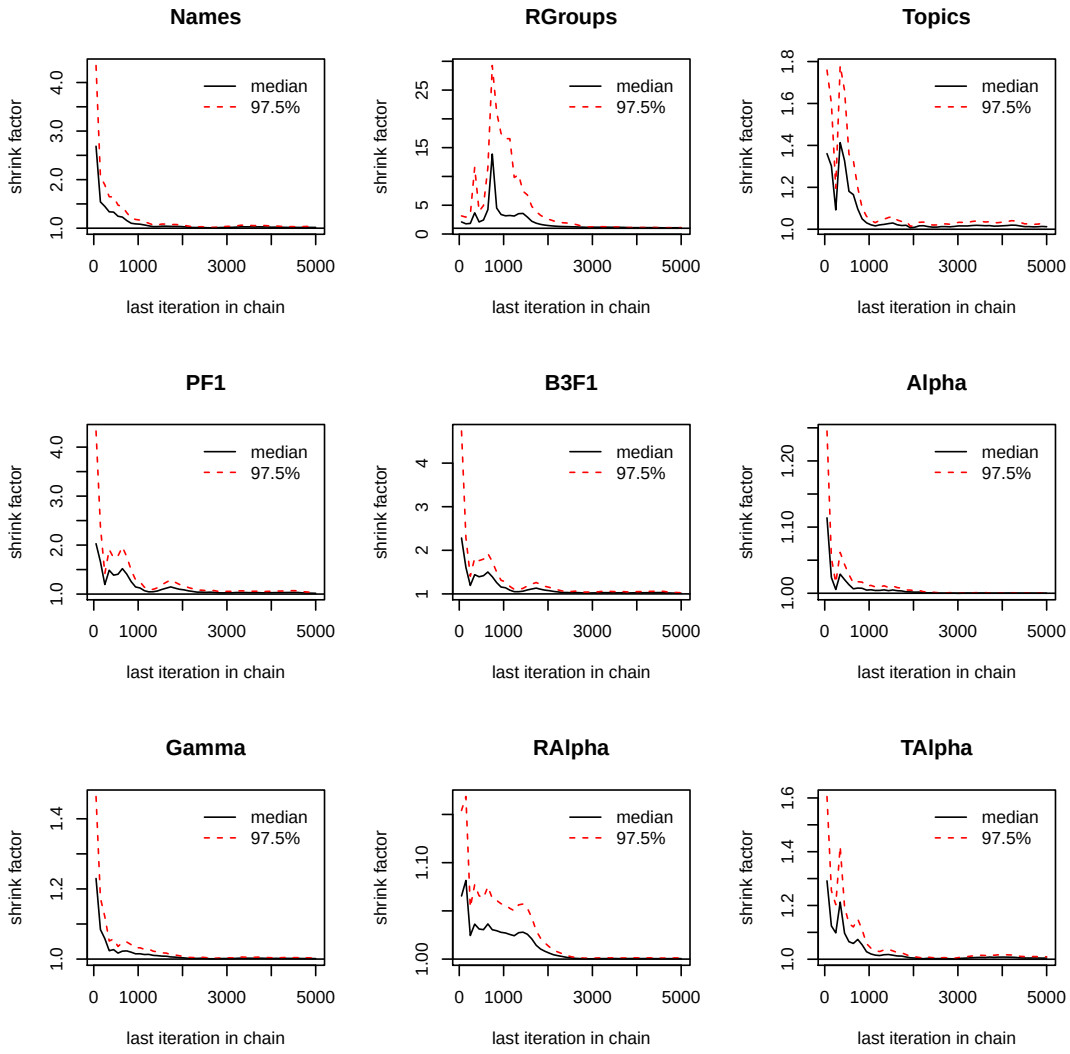


Figure 5.6: A plot of the Gelman and Rubin diagnostic scale reduction factors for the toy dataset. The quantity labels are the same as those in Figure 5.5. The plot shows that the chain has appeared to converge by around 2,000 iterations.

cleaned by Bhattacharya and Getoor (2006) and consists of citations to four areas in machine learning. The ground truth was compiled by Culotta and McCallum. This is the same dataset used in the experiments in Chapter 4. I extracted abstracts for the documents in the CiteSeer dataset from the CiteSeer website and after removing documents where the abstract could not be found, the dataset contained 1,680 references to 706 authors across 852 documents. There were a total of 42,507 words in the abstracts for the documents. The Rexa dataset (Peng and McCallum, 2006) contains documents from the Rexa citation database and is partially labelled by Culotta. Experiments are run on the entire dataset with both labelled and unlabelled authors but only evaluated using the labelled authors. The Rexa dataset contains 9,366 author references in total with 1,972 of those labelled to 105 author identities and 20,600 words across 2,697 abstracts. Since only 20% of the references are labelled, I expect that the results will be biased towards models that cluster many names together since each labelled author will generally have many occurrences. On the other hand, the CiteSeer dataset contains many more singleton author entities, authors that only appear once in the corpus. For all datasets, I applied a standard stoplist and stemmed the words. For the Rexa dataset, words which appeared less than 9 times were also removed. This resulted in a vocabulary size of 790 for Rexa and 5,695 for CiteSeer.

I compare my model, the grouped author-topic model, with other similar approaches:

ONE GROUP PER DOCUMENT The grouped author-topic model where a document is allocated to one group. This is the model shown in Figure 5.4b.

MULTIPLE GROUPS PER DOCUMENT The grouped author-topic model where a hierarchical model is used for the groups, so that a document is associated with a mixture of latent groups. Words and authors can be allocated to different groups within a document. This is the model shown in Figure 5.4a.

NO GROUPS A model where words are allocated to author entities directly rather than to groups. This is similar to the model in Chapter 4 with a non-conjugate base measure and better name variant model. Figure 4.1 shows the model closest to this one.

LDA-ER A reimplementation of the LDA-ER model (Bhattacharya and Getoor, 2006), which uses the concept of groups to perform disambiguation but does not use any of the abstract or free-text and requires the number of groups to be fixed in advance. This is one of the best performing unsupervised generative models used for entity resolution.

EXACT STRING MATCH This method assigns identical names to the same entity.

I sampled the entity, topic and group allocations in rounds. η was set to 0.1 (similar to the HDP-LDA model) as this was found to be the best setting after training on 10% of the CiteSeer dataset. η^a and the parameters for the name variant model were also trained on the training dataset where the best value of η^a was found by testing values at different orders of magnitude. For the entities, I used an uninformative Gamma$(1, 0.005)$ prior for γ and τ and a Gamma$(1, 1)$ prior for α and updated by sampling from their posteriors. These priors and similar priors for concentration parameters were chosen via minimising KL-distance to give a uniform prior distribution for the number of clusters, as in Section 3.8.4. Changing the priors by an order of magnitude did not significantly influence the results. However, using a standard Gamma$(1, 1)$ prior for γ produced poorer results with lower deduplication performance as measured by F1 by a few percent as overclustering occurs. This is because the standard prior for γ puts a very high prior probability on a very small number of clusters which is an unreasonable assumption for the number of author identities. I set the hyperparameters for the LDA-ER model to be those that are used for the other models to ensure a uniform prior distribution for the number of clusters.

I used the B^3 algorithm to calculate precision, recall and F1 as defined in Section 4.7.1, which is the standard algorithm used in coreference system evaluations. I also calculate the pairwise duplicate recall, precision and F1 scores and these are displayed in Table 5.3. However, pairwise scores underweight the performance of a model on entities that have few occurrences.

From the Gelman and Rubin potential scale reduction factor (PSRF) plots in Figure 5.7, it can be seen that the chains have appeared to converged after 100,000 iterations. The model is implemented using C++ and it takes around 4.5 hours to run 100,000 iterations running on a single core of a 3.00GHz Intel Xeon CPU for the CiteSeer dataset.

My results show that the grouped author-topic model performs better than other unsupervised approaches including LDA-ER. The grouped author-topic model with only one group per document performs best on the CiteSeer dataset, whereas the variant model where multiple groups are allocated to each document performs best on the Rexa dataset. This shows that which of the grouped author-topic models is better may depend on the characteristics of the corpus. In the case of the variant that allows for multiple groups per document, this may imply that there are weaker research group and coauthor relationships in the Rexa dataset. Another difference between the two variants of the grouped author-topic model is that the model with multiple groups per document learns fewer topics. This is most likely because each word and corresponding topic in a document is more likely to be allocated to a group that already has instances of that

Table 5.3: Results on Rexa and CiteSeer datasets. Means and standard deviations are across 5 parallel chains averaged over the last half of each 100,000 iteration chain. The two models that are based on the grouped author-topic model perform the best with the model that allows for a document to be allocated to multiple groups performing slightly better than the one where a document can only be allocated to one group on the Rexa dataset and vice versa for the CiteSeer dataset.

	Rexa			CiteSeer		
	B^3 results					
Model	Recall	Precision	F1	Recall	Precision	F1
One group/doc	94.6 ± 1.2	**99.7** ± 0.0	97.1 ± 0.6	97.5 ± 0.5	**99.6** ± 0.2	**98.5** ± 0.3
Multiple groups/doc	95.1 ± 0.6	**99.7** ± 0.0	**97.4** ± 0.3	97.1 ± 0.8	99.5 ± 0.2	98.3 ± 0.4
No groups	92.2 ± 1.4	99.4 ± 0.1	95.7 ± 0.7	97.4 ± 0.4	96.8 ± 0.4	97.1 ± 0.3
LDA-ER	60.2 ± 2.2	99.6 ± 0.0	75.0 ± 1.7	84.1 ± 0.7	100.0 ± 0.0	91.3 ± 0.4
Exact string match	57.4	99.6	72.9	80.3	100.0	89.1
	Pairwise results					
Model	Recall	Precision	F1	Recall	Precision	F1
One group/doc	94.7 ± 3.0	99.9 ± 0.0	97.2 ± 1.7	93.7 ± 1.6	99.6 ± 0.2	96.6 ± 0.9
Multiple groups/doc	95.5 ± 0.7	99.9 ± 0.0	97.7 ± 0.4	92.1 ± 2.6	99.6 ± 0.2	95.7 ± 1.4
No groups	94.6 ± 1.3	99.8 ± 0.1	97.1 ± 0.7	93.2 ± 1.7	95.2 ± 1.1	94.2 ± 1.0
LDA-ER	55.6 ± 4.8	99.9 ± 0.1	71.3 ± 3.7	68.7 ± 2.3	100.0 ± 0.0	81.4 ± 1.6
Exact string match	52.6	99.9	68.9	61.5	100.0	76.1
	Posterior counts					
Model	Topics	Entities	Groups	Topics	Entities	Groups
One group/doc	206 ± 5	1710 ± 4	385 ± 5	125 ± 5	674 ± 4	220 ± 6
Multiple groups/doc	134 ± 7	1719 ± 5	546 ± 30	89 ± 5	674 ± 5	285 ± 10
No groups	43 ± 8	1626 ± 9	—	699 ± 10	645 ± 6	—
LDA-ER	—	2377 ± 8	332	—	785 ± 5	96
Exact string match	—	1972	—	—	817	—
Ground truth	—	—	—	—	706	—

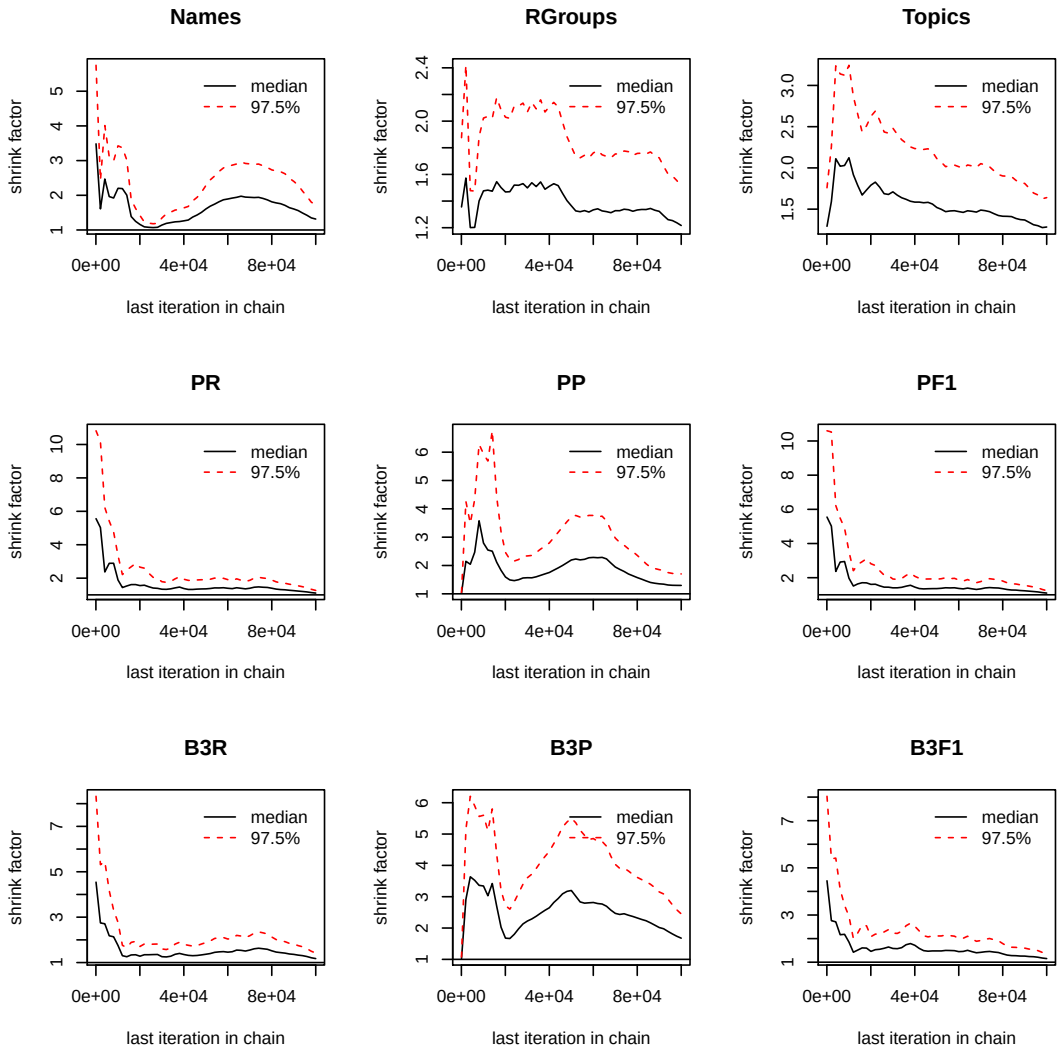


Figure 5.7: A plot of the scale reduction factors for the Gelman and Rubin diagnostics for the CiteSeer dataset. *Names* is the number of latent entities, *RGroups* the number of latent groups, *topics* the number of topics, *PR* the pairwise recall score, *PP* the pairwise precision score, *PF1* the pairwise F1 score, *B3R* the B^3 recall score, *B3P* the B^3 precision score and *B3F1* the B^3 F1 score. The plot shows the chains have appeared to converge by around 100,000 iterations.

topic allocated to it. As a result, it is less likely for a topic to be allocated to a group that does not have any similar topics and so it is less likely for new topics to be instantiated. The increased flexibility of allowing multiple groups to be allocated to a document also means that there are more groups in general. These differences most likely contributed to the differing results for the two datasets so that the best variant of the grouped author-topic model for a dataset depends on how flexible the model needs to be.

The LDA-ER model assumes that identical author references always refer to the same author identity. Applying this assumption to my model requires that identical references are assigned to the same entity. This effectively putting an upper limit on the number of entities equal to the number of unique names in the dataset. Inference then involves assigning an entity to each set of identical references simultaneously. However, this would result in a model that would be unable to handle ambiguous names, a significant advantage of my model over LDA-ER.

LDA-ER does not perform as well in the CiteSeer dataset as seen in Table 5.3, and it can be seen that the posterior has more entities than the other models. The results show that with the domain-specific name variant model, naively adding abstract information results in worse performance than not using abstracts at all. My model succeeds in integrating abstract and co-author information as seen in the results as compared with results for the models which do not (LDA-ER and the no groups model). The model without groups exhibits poorer performance than the model with groups, most likely because the posterior of the DP overweights author entities with many assigned words so that the name of the author has less effect. This can be seen by the lower precision in the model without groups than the models with groups. The model without groups also has significantly fewer topics and entities than the other models. The small number of topics again indicate that some author entities are overweighted due to the number of words allocated to them, making it unlikely that new entities are instantiated.

The grouped author-topic model also has better precision than the other models. This is because the co-author information aids in separating out similar entities more than the topics do. This can be valuable in certain applications where precision is more important than recall when it would be a major error if the system confused two similar but different entities. The F1 score in the results equally weights the recall and precision scores.

The results for these models also improve on the results from Section 4.8. Much of the improvement likely comes from using a better name variation model that is domain-specific whereas in Chapter 4 a name likelihood model was used that had a conjugate prior so allowing the name variation parameters to be integrated out. This allowed for simple inference while allowing the name model to be quite general. These models also

Table 5.4: Example of an inferred research group from the Rexa dataset spread across 20 documents.

Names	Topics
N. Cristianini, Taylor J. Shawe, J. Kandola, J. Platt, H. Lodhi, P. L. Montgomery	spectral, clustering, classification, semantic, kernel, method, extension

significantly improve on the simple baseline of using an exact string match to assign identical names to the same author identity. This is a reasonable approach to take when the number of author identities are unknown and when there is no name variation model.

The inferred topics in Table 5.5 seem reasonable and fit the theme of the corpus, which focused on computer vision, logic and reinforcement learning. An example of an inferred research group is given in Table 5.4. This appears to be a reasonable group consisting of people who collaborate on some of the papers in the corpus on various clustering and kernel methods.

5.6.3 *Conflated CiteSeer dataset*

The CiteSeer and Rexa datasets have very little ambiguity. As seen in the results, performing an exact string match on the author name results in 100% precision, which means that two references of the same author name will always refer to the same entity. Thus to better show the capabilities of my model in disambiguating entities, I created another dataset where some of the authors from the CiteSeer dataset are conflated together. This is also the same conflated dataset used in Chapter 4. This is done by taking the CiteSeer dataset and discarding the last names from the authors. The process results in a dataset with 362 unique names or roughly 2 author entities for every unique author name. η^a was retrained on 10% of this conflated dataset and all the other hyperparameters were set to the same as those in the full CiteSeer dataset.

The results on this dataset are shown in Table 5.6. They show that the grouped author-topic model where only one group is allowed per document performs significantly better at disambiguating entities than the other models. In addition, with this dataset the abstract information gives better precision than the model that does not use abstracts. However, the model where multiple groups are allowed per document performs worse than the other approaches. Looking more closely, there are many more groups in the posteriors for this dataset than in the original CiteSeer dataset. A likely reason for this is that, in this dataset, author names are more similar and provide less information about author identity. As a result, having multiple groups per document leads to more groups

Table 5.5: The top 10 inferred topics in the CiteSeer dataset.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
system	reasoning	interface	reinforcement	reinforcement
present	representation	user	learning	learning
problem	logic	system	environment	robot
paper	knowledge	language	algorithm	control
base	qualitative	design	programming	learn
approach	inference	programming	markov	action
algorithm	temporal	application	dynamic	signal
describe	spatial	graphical	agent	state
result	property	management	policy	architecture
show	relations	interactive	reward	controller
Topic 6	**Topic 7**	**Topic 8**	**Topic 9**	**Topic 10**
constraint	reasoning	constraint	surface	face
satisfaction	case-based	logic	image	recognition
problem	knowledge	programming	reconstruction	feature
search	effort	abstract	object	image
hard	share	program	orientation	human
show	reasoner	interpretation	boundary	facial
difficulty	retrieval	language	motion	classification
binary	organization	analysis	geometric	training
variable	integration	semantics	arbitrary	scale
backtrack	utility	concurrent	shape	add

being used to explain the variation in the document abstracts. On the other hand, in the original CiteSeer dataset, the groups were used to explain the variation in author references and co-authors. This can be seen in the posterior where there are twice as many groups as entities in the conflated dataset, whereas there are twice as many entities as groups in the original dataset.

The results for these models compared to Section 4.8 have a much smaller advantage from the name variation model since names are a smaller distinguishing factor in this conflated dataset. This can be seen since only the model with one group per document surpasses the best performing model in Section 4.8. This shows that modeling research groups and having a model of one group per document is good for modeling corpuses with high name ambiguity. The one group per document model also significantly improves on the simple baseline of using an exact string match to assign identical names to the same author identity, whereas the other models do not improve on this baseline.

5.6.4 *John Smith ambiguity dataset*

This dataset is one that LDA-ER cannot tackle due to its inability to disambiguate authors with identical names. I ran experiments on the 'John Smith' corpus to determine if the model can disambiguate authors based solely on document text without a name variation model. The task is to disambiguate the name references using text in the document. The corpus and hand-labelled ground-truth was provided by Bagga and Baldwin (1998b). The corpus is an ambiguous set of 197 articles extracted from the 1996 and 1997 editions of the New York Times. The criteria for including an article was the presence of a string that matched the '/John.*?Smith/' regular expression. This ensures that the matched articles have name references for people with the first name John and last name Smith but does not capture name variants. The dataset consists of 35 different John Smiths, 24 of these were only mentioned in one article whereas the remaining 11 were mentioned in the other 173 articles. The background and profession of each of the John Smiths varies greatly from a CEO to the former head of the Labor party.

Following Gooi and Allan (2004), I extracted a 55-word window of text centered around the 'John Smith' reference in each document and used these in the experiments. I removed the authorial part of the model so that each group is used to represent an author entity and so an author entity is now a distribution over topics. I perform experiments with uninformative priors on the concentration parameters and evaluate using the posterior group allocations. The results are evaluated using the B^3 score and compared with a vector space model in Table 5.7. The agglomerative vector space model (Bagga and Baldwin, 1998b) requires setting a similarity threshold above which two clusters

Table 5.6: Results on the conflated CiteSeer dataset. Means and standard deviations are across 5 parallel chains averaged over the last half of each 10,000 iteration chain.

	B^3 results		
Model	Recall	Precision	F1
One group/doc	69.2 ± 0.8	**91.5** ± 0.7	**78.8** ± 0.6
Multiple groups/doc	63.6 ± 2.1	36.3 ± 2.6	46.2 ± 2.5
No groups	72.5 ± 0.7	45.0 ± 0.8	55.5 ± 0.7
LDA-ER	82.3 ± 0.4	36.4 ± 0.7	50.5 ± 0.6
Exact string match	80.8	42.4	55.6

	Pairwise results		
Model	Recall	Precision	F1
One group/doc	38.3 ± 1.7	78.8 ± 2.6	51.5 ± 1.6
Multiple groups/doc	37.3 ± 4.0	16.8 ± 2.3	23.1 ± 2.9
No groups	48.4 ± 1.7	17.7 ± 1.3	25.9 ± 1.5
LDA-ER	65.9 ± 1.3	9.0 ± 0.2	15.8 ± 0.3
Exact string match	62.6	8.7	15.2

	Posterior counts		
Model	Topics	Entities	Groups
One group/doc	391 ± 11	843 ± 11	327 ± 7
Multiple groups/doc	337 ± 10	336 ± 16	732 ± 63
No groups	86 ± 10	386 ± 10	—
LDA-ER	—	259 ± 7	96
Exact string match	—	1526	—
Ground truth	—	706	—

Table 5.7: Disambiguation B^3 results on John Smith dataset. Vector space is a model that uses a vector space model to compare entity similarity and the results are given for the best similarity threshold, this acts as an upper-bound. All separate is a method that assumes every John Smith is a unique individual.

Model	Recall	Precision	F1
Unsupervised grouped Author-Topic	79.4	56.5	**66.0** ($\pm$ 5.8)
Vector space model with best threshold	65.9	81.7	72.9
All separate	21.8	100	35.8

are considered to refer to the same entity. The best result over a range of thresholds is found on the test set and given in the table. As a result, this is not a baseline and acts more as an upper-bound since the test dataset was used to set the parameters for the vector space model. However, even though my unsupervised model is run fairly with no tuning of the parameters, my model performs well compared with the best performing agglomerative vector space model. Putting a smaller prior on the concentration parameters may reduce the overclustering that happened during inference causing different people to be assigned to the same entity. This could be done by training η^a or the concentration parameters on a training dataset.

5.6.5 *WePS 2 people clustering dataset*

Finally, I show results on another disambiguation task. I ran experiments on the dataset from the WePS 2 (Artiles, Gonzalo and Sekine, 2009) people clustering task. The goal of the task is to disambiguate person names in web search results. 30 randomly chosen names were searched for on an Internet search engine. The top 150 search results were retrieved, and each document was hand annotated to match with a real identity. The dataset is highly ambiguous with an average of 18 different people per name.

I extracted the words from each webpage, removed stopwords and ran the result through the Stanford named entity recogniser (Finkel, Grenager and C. Manning, 2005). I used the extracted named entities in place of the author references in my model and used the Jaro-Winkler distance metric as the name variation model. This flexible model was chosen to allow different ways of writing names, locations and organizations to be matched together. I used the non-entity words as the observed words for each document. I then performed experiments with priors on the concentration parameters that were scaled logarithmically in proportion to the given real-world frequency of that name. Since annotations are labelled at the document level, I evaluate my model using the

Table 5.8: Macro-averaged B^3 disambiguation results on the WePS 2 dataset.

Model	Recall	Precision	F1
Bag of words with best threshold	83	89	85
Unsupervised grouped Author-Topic	50	82	**56**
Supervised bag of words	48	95	59
Each document in individual cluster	24	100	34

posterior group assignments. The results in Table 5.8 show that my unsupervised model almost matches the performance of the supervised bag of words approach. The supervised bag of words approach represents documents as a bag of words weighted with TF-IDF. The similarity between documents is calculated by cosine distance, and pairs of documents above a set similarity threshold are assumed to refer to the same entity. The supervised approach has the advantage of learning a similarity threshold from the training set, which is fully labelled with the true identities. On the other hand, my model does not make use of any of the training set. Results where the best threshold is found on the test set is also given and acts as an upper bound. Many of the approaches used by other participants in the task are reliant on learning a cutoff threshold from the training data or using a supervised approach using additional features based on person attributes or web search queries. The topics and inferred groups that are inferred by my model could also be used as additional features in a supervised algorithm.

5.7 CONCLUSIONS

My grouped author-topic model models the authorship of a document through a hierarchical model that combines a topic model with a multiple authorship model. This allows information that comes from a document having multiple authors and the content in a document to be leveraged to usefully disambiguate the authors that are represented in the corpus. In this chapter, I have evaluated my unsupervised model against both toy and real world data and shown that it performs well in the task of identity resolution compared against other state of the art approaches. The model shows significant improvement over ignoring groups or abstracts in the citation database examples and shows that it can perform well at disambiguating entities with identical names in a set of documents. The results show that the best-performing method for disambiguation is one in which the entities and topics in a document are explained through a single latent variable, a latent group that gives a distribution over both entities and topics. This al-

lows the posterior distribution over entities conditional on the group to be independent of the length of text in each document. However, in cases where there is little or no ambiguity in the dataset then the variant of the model that allows for multiple groups in the document performs better.

Each latent group in the grouped author-topic model essentially merges the documents allocated to that group into one super-document. This can aid in analysing short documents, such as tweets and text messages where messages have strict character length limits. Standard topic models are typically unable to learn coherent topics from the few words available in such short documents. Document merging can be done but then it can be difficult to decide which documents should be merged. The grouped author-topic model allows good topics to be learnt from the automatically merged super-document. The super-document would be represented by the latent group, which could be a collection of some of the related topical tweets by a group of related users or by one user. This can also lead to detecting which messages have been split into a set of smaller messages due to the message length limits. Finally, the model can also be reduced to a typical HDP model by forcing each document to its own latent group.

My model is versatile in that it can disambiguate identical name references that refer to different entities, as well as combine differing references to the same entity. The model is fully automated in that it does not require pre-specification of numbers of entities, research groups, topics, etc. This is a result of the model taking a Bayesian nonparametric approach to the problem and allowing broad uninformative priors for the number of entities, etc. while more informative priors for the number of entities can be chosen if needed.

To accomplish inference, a Gibbs sampler is used to sample from the posterior. Although the base measure for the entities is non-conjugate, using an auxiliary variable Gibbs sampler still resulted in good performance. With the aid of various convergence diagnostics, the sampler is also shown to converge quickly and mix well. The name variation model is based on a domain-specific string metric, which allows the probabilities of characters to be changed, inserted or deleted in a name to be specified a priori or learnt from a training dataset.

There is room to explore other possibly faster sampling options such as a variational inference procedure (Teh, Kurihara and Welling, 2008) although the non-conjugate base measure complicates this. The name variation model could be changed to a *n*-gram model or a discriminative name model to simplify inference or to use the model in other areas. For example, the appropriate likelihood and base measure may allow the modelling of co-entity relationships to be used for word sense disambiguation. The dependent DP framework introduced in MacEachern (1999) could also be used to allow

for a set of dependent random measures that are marginally DPs, allowing the model to be extended dynamically or use more complicated types of base measure. In the context of my framework, this dependency can be viewed as an indicator variable indicating whether an observation in a document is an author reference or a word and then using the appropriate base measure. It would also be possible to model how research groups change over time in terms of their member composition and the topics they write about. This could be done by extending the model to integrate publication dates.

6

NONPARAMETRIC AND KEYWORD-SUPERVISED TOPIC MODELS

In this chapter, I develop two new supervised models that utilise topics, where topics are distributions over the vocabulary and are described in Section 3.5. In addition, I analyse their performance with experiments on *regression* and *classification* tasks. The problems of regression and classification are fundamental in a variety of tasks and settings from computer vision to natural language processing. These problems consist of labelled examples where each example is a pair consisting of a *predictor*, also known as input or independent variable, and a *response*, also known as output or dependent variable. These examples are then used to predict the responses for unlabelled test data.

The new models are the nonparametric supervised HDP (sHDP) model and the parametric keyword-supervised topic model (ksLDA). The sHDP model is a nonparametric extension of supervised latent Dirichlet allocation (sLDA) (David M. Blei and McAuliffe, 2007). The sHDP has an infinite number of topics, preventing the overfitting or underfitting that can result from a fixed number of topics that is unsuitable for the dataset. It is a supervised extension of the HDP mixture model described in Section 3.8.7. The ksLDA model is an extension of sLDA where different words can have a more direct effect on a document's response and the effect of the words can be perturbed by their context. In this chapter, I show that these new models perform better than sLDA.

6.1 INTRODUCTION

Topic models, as described in Section 3.5, are an unsupervised model of the text in the documents of a corpus. The latent topics that are learnt by the models are particularly important when modelling large document collections as they can reduce the dimension of the data. These models define a topic as a distribution over words in the vocabulary of the corpus. Each topic can be thought of as a group of semantically related words, and these inferred topics shed light on the common themes that run through the documents. The models have been successful in analysing collections of documents, including citation databases and newsgroup corpora. They can also be used for a wide range of applications including data exploration and information retrieval.

Recently, attention has turned to these models as ways of performing regression and classification on collections of documents, where each document possesses an associated response. The response can be categorical, continuous, ordered or other types. For example, the responses can be sentiment ratings or field categories. A simple approach to this problem is to use topic models as a dimensionality reduction method and then to regress on the lower dimension dataset. A set of topics is learnt for the corpus using a topic model (LDA) while ignoring the document responses. Then the document responses are regressed on the empirical topic distribution for each document. But this approach performs poorly in contrast to directly regressing on the empirical word distribution for each document. The topics that are learnt also often have no relation to the responses that need to be predicted. As a result, words that cause positive responses and words that cause negative responses end up being assigned to the same topic. This has spurred interest in supervised topic models that can learn topics that are both good models of document contents and are good predictors for document responses.

Supervised topic models such as supervised latent Dirichlet allocation (sLDA) (David M. Blei and McAuliffe, 2007) are an extension of topic models. Topics that are learnt are more useful than that obtained in unsupervised topic models such as LDA for predicting a document's response. This is because the learnt topics are oriented around words that relate to document responses. In contrast, unsupervised LDA learns topics that are in line with the general theme of the documents, but are often unrelated to the document responses. David M. Blei and McAuliffe (2007) found that the predictions made by sLDA of the responses for an unseen test set were better than the predictions made using the unsupervised topics inferred by LDA. However, they found that the sLDA model only performed slightly better than LASSO regression (Tibshirani, 1994) on the empirical distribution of words for each document.

Although these models perform well, they are limited as the number of topics in the model must be fixed in advance. This can lead to overfitting if there are too many topics or underfitting if there are too few topics where the learnt topics may poorly represent the documents. A number of methods can be used to choose the number of topics, including cross-validation and model comparison techniques; however, these are often unsatisfactory and slow. Bayesian nonparametric methods, as described in Section 3.7, have emerged as a good way to extend these models naturally to handle a flexible number of topics.

In this chapter, I present the supervised HDP (sHDP) model as a generative supervised model that has an infinite number of topics (or clusters) used to predict a document response and the keyword-supervised topic model (ksLDA) as an extension to the sLDA model that allows different words to have a more direct effect on the document

responses and for the words' context to perturb their effect. The rest of the chapter is organised as follows. In Section 6.2, I briefly review some existing work on tackling the supervised learning problem with nonparametric models and also approaches specifically for grouped data. Section 6.3 sets the problem and the form of the data for the models proposed in this chapter. Section 6.4 gives an introduction to generalised linear models, which are a class of models that generalise linear regression. I review the existing sLDA model in Section 6.5. I then introduce my two new models, the supervised HDP model in Section 6.6 and the keyword-supervised topic model in Section 6.7. Section 6.8 describes the Gibbs inference algorithms that are used to sample from the posteriors of the new models. Finally, Section 6.9 covers experiments with these models on real-world datasets consisting of both binary and continuous responses and compares the new models to existing models.

6.2 EXISTING WORK

Bayesian nonparametric models have more flexibility than that in parametric models. Nonparametric models allow the number of utilised parameters in the model to grow as more data is observed so that the structure of the model can adapt to the data. Dirichlet process (DP) mixtures are a popular type of nonparametric model that can have an infinite number of clusters. These models are commonly unsupervised and are used for problems that require a model that can better adapt to the data.

Due to their flexibility, there has been interest in supervised nonparametric models, such as the regression models of Gaussian processes (GPs) (Rasmussen and Williams, 2006) and Bayesian regression trees. Dirichlet processes have also been adapted for supervised problems. An example of this is the DP multinomial logit model (dpMNL) (Shahbaba and Radford M. Neal, 2009). In this generative model, each example is a covariate with an associated response. The relationship between the covariates and responses are then modelled jointly using DP mixtures. Although within each cluster the relationship is assumed to be linear, an overall nonlinear relationship occurs when the model has more than one cluster. A multinomial logit is used to model the responses conditionally on the covariates within each cluster. Thus, the regression parameters of the logit model are different for each cluster. The predicted responses are conditional on the parameters and the covariates. The dpMNL model was tested on protein fold classification, and compared with existing methods based on neural networks and support vector machines. The results showed that the dpMNL model performed significantly better.

The dpMNL has been extended to model additional response types with DP mixtures of generalised linear models (DP-GLM) (Hannah, D. Blei and Powell, 2011). Whereas the dpMNL only explicitly models discrete responses, the DP-GLM can generatively model both continuous and discrete responses using different generalised linear models. Again, the regression coefficients of the generalised linear models are different for each cluster. Priors are also placed on the coefficients, resulting in a model for the response that is similar to a regularised regression model. The model was shown to have weak consistency by Hannah, D. Blei and Powell (2011), and the performance was shown to be comparable to a Gaussian process model, which is a nonparametric model that is the state of the art for regression.

Both kinds of approaches, however, have not yet been applied to grouped data such as documents and the problem of predicting the responses of groups of observations. Supervised latent Dirichlet allocation (sLDA) is one approach to tackling this prediction problem for grouped data. sLDA learns topics that are able to model the document responses more accurately. For example, in sentiment analysis tasks, the topics learnt would consist of words that cause the document to have positive or negative sentiment. Similarly, for financial news, the topics would consist of words that have positive or negative effects in the market. The sLDA model has, however, limited flexibility since the number of latent topics must be fixed in advance leaving it at risk of overfitting or underfitting. The sHDP model I present in this chapter removes that limitation by extending the HDP mixture model to supervised problems.

Hierarchical Dirichlet process (HDP) mixture models, described in Section 3.8.7, are a type of Bayesian nonparametric model that can be used instead of LDA for topic modelling. They are commonly used as the nonparametric analog to LDA, allowing for flexible topic modelling without being restricted to a fixed number of topics. Though inference is more complex, Gibbs sampling and variational Bayes techniques can still be applied. Until now, HDP mixtures have not seen significant use in supervised problems and suffer the same problems as unsupervised LDA in that the topics learnt are not necessarily predictive of the responses. The sHDP model I present in this chapter extends the HDP mixture model to learn topics that are good predictors of document responses.

Another problem with supervised topic models is that they model the effect of the words on the document response only indirectly through topic allocations, and so they are unable to model the magnitude of different words' effect on the document response. For example, stronger words such as *funniest* would be clustered into the same topic as milder words such as *fine* and so end up having the same regression coefficient as other words in the topic. I propose the ksLDA model that allows different words to have a

more direct effect on the response and for their effect to be influenced by the context of the words. There has also been work on other methods of learning the regression coefficients or other response types such as DMR (Mimno and McCallum, 2008), DiscLDA (Lacoste-Julien, Sha and Jordan, 2008), MedLDA (Zhu, Ahmed and Xing, 2009) and labeled LDA (Ramage et al., 2009), however, these models still have a fixed number of topics and do not directly model the effect of different words.

6.3 PROBLEM DESCRIPTION

We assume that there is a set of data points divided into D groups. To reduce complexity, a bag of words model can be used for each group, amounting to assuming exchangeability among the observations within a group. Each group i consists of both a variable number of data points $x_{ij}, j = 1, \ldots, N_i$, which are the predictors, and a single response y_i. In the case of document modelling, D is the number of documents in the corpus, each word uses one-of-V encoding $x_{ij} \in \{1, \ldots, V\}$ where V is the size of the vocabulary of the corpus. y_i is the response for the document, such as a rating or a category. In the rest of this chapter, the models will be described in terms of documents and words, but the models can also be used on other kinds of grouped data. Given a set of training examples with predictors and associated responses, the task is to predict the responses on a separate test set.

6.4 GENERALISED LINEAR MODELS

Often when a response is not an unconstrained continuous variable, it is transformed into one and a normal linear model is used for it. However, this may not always be appropriate. A *generalised linear model* (GLM) expands the flexibility of linear models by being capable of analysing data where either there may not be a linear relation between the covariates x and the response y or where a Gaussian assumption for y is inappropriate. A generalised linear model is specified by a linear predictor which I denote in this section by θ, a link function $g(\cdot)$ that relates the linear predictor to the mean of the response $\mu = g^{-1}(\theta)$ and a probability distribution from the exponential family, described in Section 3.2, that gives the distribution of the response y with mean $\mathrm{E}(y|\cdot) = \mu$. In this chapter, I only consider canonical link functions though others can be used when needed. The canonical link function is the function of the mean parameter that is in the exponent of the exponential family form of the probability density. The

distribution of the response may also be an exponential dispersion family that has an additional dispersion parameter denoted as γ.

A generalised linear model that models a response y is

$$p(y|\theta,\gamma) = h(y,\gamma)\exp\left\{\frac{\theta y - A(\theta)}{\gamma}\right\}, \tag{6.1}$$

where the distribution of the response is modelled by an exponential dispersion family with natural parameter θ. $h(y,\gamma)$ is the base measure, y the sufficient statistic and $A(\theta)$ the log-normaliser. The canonical parameter is θ.

Different forms of responses can be modelled using different choices of h and A. In particular, there is a normal distribution on y,

$$p(y|\theta,\gamma) = \frac{1}{\sqrt{2\pi\gamma}}\exp\left\{-\frac{1}{2\gamma}(y-\theta)^2\right\} \tag{6.2}$$

when $h(y,\gamma) = \left(1/\sqrt{2\pi\gamma}\right)\mathrm{e}^{-y^2/2}$ and $A(\theta) = \theta^2/2$. This is a normal linear model with a mean of θ and variance of γ.

When y is binary, a binomial distribution can be used with the number of trials $n = 1$, so that y is distributed as

$$p(y|\theta) = \theta^y(1-\theta)^{1-y} \tag{6.3}$$

which uses the canonical logit link function $g(\theta) = \ln(\theta/(1-\theta))$ and the binomial distribution for y. This choice of distribution and link function results in a logistic regression model.

6.5 SUPERVISED LATENT DIRICHLET ALLOCATION

Supervised latent Dirichlet allocation (sLDA) (David M. Blei and McAuliffe, 2007) is an extension of LDA (described in Section 3.5) to supervised problems. It partially overcomes the problem that the topics that are learnt cannot be controlled in the LDA model. The learnt topics in LDA act to reduce the dimension of the data but may not be predictive of a document's response as they will correspond to the general themes of the corpus rather than be predictive of document responses. sLDA overcomes this problem by jointly learning topics and their regression coefficients for the document responses. The response for a document is predicted by averaging over the empirical topic allocations for a document.

The following generative process for each document i is followed. Let K be the fixed number of topics, N_i the number of words in document i, $\boldsymbol{\phi}_{1:K}$ the topics where each $\boldsymbol{\phi}$ is a distribution over the vocabulary, α a parameter for topic proportions, and η and δ the response parameters.

1. Draw topic proportions $\theta_i \sim \text{Dirichlet}(\alpha)$.
2. For each word,
   a) Draw a topic assignment $z_{ij} \sim \text{Multinomial}(\theta_i)$.
   b) Draw a word $w_{ij}|z_{ij} \sim \text{Multinomial}(\phi_{z_{ij}})$.
3. Draw the document response $y|z_{i,1:N_i}, \eta, \sigma^2 \sim \text{GLM}(\cdot|\bar{z}_i, \eta, \delta)$ where $\bar{z}_i = 1/N_i \sum_{j=1}^{N_i} z_{ij}$.

The linear predictor in the GLM model for the response is $\eta^\top \bar{z}$ where $\bar{z}$ are the empirical frequencies of the topics in the document and η are the regression coefficients. Since the GLM model conditions on the unobserved empirical topic assignments, the document's response is non-exchangeable with the topic assignments. As a result, the document's contents are assumed to be generated first and then the document's response is chosen conditional on those contents. This is a reasonable assumption to make as most labels for documents are chosen post-hoc after the document has been written, such as a document's category. An alternative to this model is one where y is regressed on the topic proportions for the document θ, which is the limit of $\bar{z}$ being averaged over an infinite number of iterations. However, this may result in some topics being estimated that just explain the response variables while other topics only explain the document words.

In the sLDA model, the parameters α, $\phi_{1:K}$, η and δ are treated as constants to be estimated. Approximate maximum-likelihood estimation is then performed with a variational expectation-maximisation (EM) method, similar to that for LDA.

The models I propose in this chapter solve two of the problems in this model. One problem with sLDA is that the number of topics must be fixed from the start. This can result in overfitting or underfitting if the number of topics is unsuitable for the dataset. Though the number of topics can be chosen based on a training set, it may be time consuming and the number is dependent on the size of the dataset. Another problem with sLDA is that the words in a document only have an effect on the model of a document's response through the words' topic allocations. The model effectively averages the effect on a document response across words in the same topic, thus the fact that some words have a stronger effect and others have a weaker effect on a document's response gets lost.

6.6 THE SUPERVISED HDP (SHDP) MODEL

The supervised HDP (sHDP) model proposed in this chapter can automatically learn the necessary number of topics to model the responses of documents on training data. It is a Bayesian nonparametric model so that a potentially infinite number of latent clusters can be used for prediction. The sHDP model extends the HDP mixture model to learn clusters that align with document responses. The relationship between the data points and the responses is modelled with a generalised linear model on the clusters to which the data points in a document have been allocated. A regression coefficient is associated with each cluster, and the document's response is regressed on the average of these coefficients.

In the sHDP model, unlike sLDA, the number of topics does not need to be fixed in advance. This is beneficial in supervised problems since it is unclear how many latent topics will be necessary to model the data and the response conditional on the document. The response is modelled by a generalised linear model (GLM) conditioned on the topics that have been assigned to the observations in the document. Since the number of instantiated topics can vary and each topic has a regression coefficient, the number of instantiated regression coefficients also varies given the current number of instantiated topics. In the generative process, a regression coefficient is sampled for each topic in addition to sampling a distribution over the vocabulary. In effect, a product base measure is used for the topics where one component is a prior over the vocabulary and the other is a prior for the regression coefficient. This treats the regression coefficients as random variables, whereas in sLDA, the regression coefficients are treated as constants. This modelling of the regression coefficients effectively results in a regularised regression model for the response variables. Each topic can also be assigned a vector of regression coefficients, which can be used when there are categorical responses.

The model is thus

$$\boldsymbol{G}_0 \sim \mathrm{DP}(\gamma \boldsymbol{H}) \tag{6.4}$$

$$\boldsymbol{G}_i \sim \mathrm{DP}(\alpha \boldsymbol{G}_0) \tag{6.5}$$

$$\theta = (\theta_{ij}^X, \theta_{ij}^Y) \sim \boldsymbol{G}_i \tag{6.6}$$

$$x_{ij}|\theta_{ij}^X \sim f(\theta_{ij}^X) \tag{6.7}$$

$$y_i|\theta_{i\cdot}^Y \sim \mathrm{GLM}(\cdot|\overline{\theta_{i\cdot}^Y}) \tag{6.8}$$

where $\overline{\theta_{i\cdot}^Y} = (1/N_i)\sum_j \theta_{ij}^Y$ is the linear predictor for the GLM, i ranges over each document, j ranges over each observation in that document, γ denotes the concentration parameter for the corpus-level DP and α denotes the concentration parameter for the

document-level DP. The base measure $H = H^Y \times H^X$ consists of a measure for the regression parameters $\theta^Y \sim H^Y$ and another measure for the topic parameters $\theta^X \sim H^X$. G_0 is the corpus-level random measure that acts as the base measure for the document-level random measure G_i.

Due to the clustering property of the DP, some data points will share the same parameters θ, which can be represented as those data points being assigned to the same topic. The prior density for the regression parameters, $\theta^Y \in \mathbb{R}$, is typically $H^Y = \mathrm{N}(0, \zeta)$ where ζ is the variance of the parameters. For topic modelling, the documents consist of words, and the prior density for the cluster parameters is $H^X = \mathrm{Dirichlet}(\alpha^w)$, where α^w is the parameter for a symmetric Dirichlet distribution and so $\boldsymbol{\theta}^X \in \mathbb{R}^W$ where W is the size of the vocabulary. I found that changing these priors by an order of magnitude did not significantly change the results. f is the likelihood of θ^X given the observations x. In a topic modelling problem, $f(\theta^X) = \mathrm{Multinomial}(\cdot|\theta^X)$. When coupled with its conjugate prior, the Dirichlet distribution, the topic parameters θ^X can be integrated out, allowing for collapsed Gibbs inference to be performed by just keeping track of the word to topic allocations and the regression coefficients for the topics. The GLM model for the responses allows the responses to be continuous, ordinal, categorical and other types depending on the form of the GLM. When the base measure for the coefficients H^Y is a Gaussian distribution, the maximum a posteriori (MAP) solution for the coefficients is similar to the solution for L_2 penalised logistic regression with a logistic GLM and for ridge regression with a Gaussian GLM. A graphical model is shown in Figure 6.1.

The generative process for the full model is:

1. Draw (from their prior distributions) the concentration parameters for the global DPs γ. Likewise, draw the concentration parameters for the lower-level DPs α from their priors.

2. Draw a global distribution over topics and their regression coefficients $G_0 \sim \mathrm{DP}(\gamma, H)$.

3. Now for each document i,
   a) Draw a distribution over topics $G_i \sim \mathrm{DP}(\alpha, G_0)$.
   b) For each word w_{ij},
      i. Draw a topic $(\boldsymbol{\theta}^X_{ij}, \theta^Y_{ij}) \sim G_i$.
      ii. Draw a word $w \sim \mathrm{Multinomial}(\boldsymbol{\theta}^X_{ij})$.
   c) Draw a response for the document $y \sim \mathrm{GLM}(\overline{\theta^Y_{i\cdot}})$ where $\overline{\theta^Y_{i\cdot}} = (1/N_i) \sum_j \theta^Y_{ij}$.

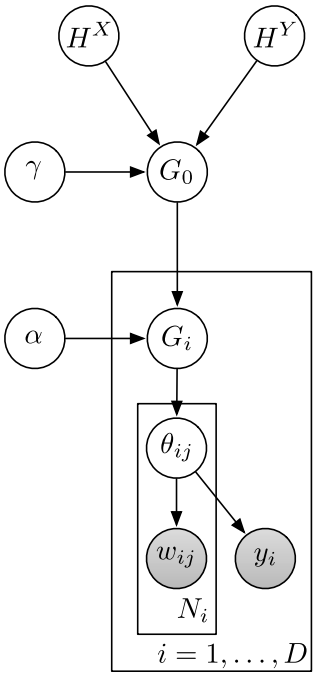


Figure 6.1: The supervised HDP model.

The sHDP learns topics that both model document contents well and are predictive of document responses without the need to choose a fixed number of topics beforehand. This structured approach to supervision allows the model to be easily extended to incorporate additional information from documents to aid in predicting the response such as the authors of a document or the research group which authored a document, which can be inferred through the model described in Chapter 5. For example, the problem of predicting the venue where a paper is published by learning the venues where the research group has previously published. Another example could be the problem of predicting a set of keywords or categories for a paper by learning which categories have previously been picked by the research group for those topics. Allowing for the topics to be supervised could also allow users of the model to control the types of topics that are learnt by the sHDP in case the unsupervised topics learnt are not along the lines of what the user would like to explore. Finally, the sHDP model allows for unlabelled data to be used as part of the training set in semi-supervised problems. This allows supervised topics to be learnt that take into account the content of unlabelled documents so that the learnt topics can better model the entire corpus instead of just the labelled documents.

6.7 THE KEYWORD-SUPERVISED TOPIC (KSLDA) MODEL

A problem with the supervised topic model is that it smooths over the effect of different words in a document on the document's response. A document's response has an indirect relationship with the words via the topic allocations in the document. As a result, it is possible for both strongly positive and weakly positive words to be allocated to the same topic resulting in them having the same regression coefficient in the model of the response. An alternative approach is to allow different words to have a more direct effect on a document's response using the keyword-supervised LDA (ksLDA) model proposed in this section.

The ksLDA model extends the sLDA model so that the response for each document is modelled both by the empirical distribution over topics in each document and by the actual words in each document. The response is modelled with a GLM that is conditioned on both the topics that have been assigned to the words in the document and the words in the document themselves. Experiments with a non-parametric variant of this model where the number of topics was not fixed showed results similar to the parametric model (as discussed in Section 6.9) so I present the parametric variant here for ease of understanding and implementation.

The following generative process for the full model is followed. Let z_{ij} denote the topic indicator for word j in document i, ϕ_k denote the parameters for topic k and $\boldsymbol{\eta} = (\boldsymbol{\eta}^t, \boldsymbol{\eta}^w)$ denote the regression parameters for the GLM.

1. Draw (from their prior distributions) the concentration parameter α for the topics.
2. Draw (from their prior distributions) the regression coefficients $\boldsymbol{\eta}^t$ for each topic and the regression coefficients $\boldsymbol{\eta}^w$ for each word.
3. Now for each document i,
    a) Draw a distribution over topics $\theta_i \sim \text{Dirichlet}(\alpha)$.
    b) For each word w_{ij},
        i. Draw a topic $z_{ij} \sim \text{Multinomial}(\theta_i)$.
        ii. Draw a word $w_{ij}|z_{ij} \sim \text{Multinomial}(w_{ij}|\phi_{z_{ij}})$.
    c) Draw a response for the document $y \sim \text{GLM}(\overline{\mathbf{z}_i}, \overline{\mathbf{w}_{i\cdot}}, \boldsymbol{\eta}^t, \boldsymbol{\eta}^w)$ where $\overline{\mathbf{w}_{i\cdot}} = (1/N_i)\sum_j w_{ij}$ and $\overline{\mathbf{z}_i} = (1/N_i)\sum_j z_{ij}$.

The prior density for the regression parameters is typically $\boldsymbol{\eta}^t \sim \text{N}(\mathbf{0}, \zeta^t\mathbf{I})$ and $\boldsymbol{\eta}^w \sim \text{N}(\mathbf{0}, \zeta^w\mathbf{I})$. In a topic modelling problem, the likelihood f, of the topic parameters is the multinomial distribution so the prior for the topic parameters can be chosen to be

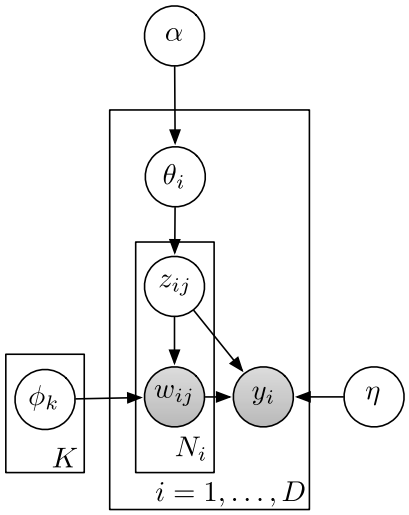


Figure 6.2: The keyword-supervised LDA model.

the conjugate prior $\phi_t \sim$ Dirichlet. This allows the topic parameters θ to be integrated out, allowing for inference to be done using cluster indicators. The GLM model allows the responses to be continuous, ordinal, categorical and other types depending on the form selected. The linear predictor for the GLM uses the word distribution of the document and is $(\boldsymbol{\eta}^t, \boldsymbol{\eta}^w)^\top(\overline{\mathbf{z}_i}, \overline{\mathbf{w}_{i\cdot}})$. During learning, $\boldsymbol{\eta}^w$ is estimated once at the start of the process and is then kept fixed. This is because re-estimating it during sampling can cause the effects of the words to be reassigned to the topics since the topics are collections of words. This decreases the regression parameters of the words and lowers the impact of strong words. $\boldsymbol{\eta}^t$ is re-estimated during sampling since the topics consantly change. When the prior for the coefficients is chosen to be a Gaussian distribution, the maximum a posteriori (MAP) solution is similar to the solution from a L_2 penalised logistic regression with a logistic GLM and from a ridge regression with a Gaussian GLM.

A graphical model is shown in Figure 6.2.

6.8 INFERENCE

Since posterior inference is intractable in both DP-based models and LDA, approximations must be used. Collapsed Gibbs sampling is the most common technique used to sample from the posteriors of these models, and it can also be applied to the models described in this chapter. For topic modelling problems and word-based datasets, the Dirichlet base measure for each topic is conjugate to the multinomial likelihood for the words. This enables the topic parameters, which are distributions over the vocabulary, to be integrated out. Thus at each iteration and based on the Chinese restaurant

process, collapsed Gibbs sampling can be used to sample the topic allocations. I estimate the maximum a posteriori (MAP) regression coefficients instead of sampling from their posteriors to reduce computation time. The following sections describe inference in both the proposed sHDP and ksLDA models. A latent indicator variable z indicates to which topic a word is allocated.

6.8.1 *The sHDP model*

Since the base measure for the topic regression coefficients will not in general be conjugate to the GLM response model, the non-conjugate auxiliary variable sampling algorithm (alg. 8) described by Radford M. Neal (2000) is used to sample the topic allocations. The main difference from inference for the HDP mixture model is in sampling the topic allocation variable and estimating the topic regression coefficients. The conditional distribution for the topic allocation has an additional term for the conditional likelihood of the topic parameters given the document response. Gibbs sampling proceeds as below.

1. For each document i,
   a) For each word w_{ij}, sample z_{ij} for the topic allocation, where n_{ik} is the number of words in document i allocated to topic k and a superscript $-ij$ for a variable denotes the variable while ignoring the current allocation z_{ij},

$$p(z_{ij} = k|\mathbf{z}^{-ij}, w_{ij}, \boldsymbol{\beta}) \propto \begin{cases} (n_{ik}^{-ij} + \alpha\beta_k) f_k^{-w_{ij}}(w_{ij}) p(y_i|\mathbf{z}^{-ij}, z_{ij} = k, \boldsymbol{\eta}), & \text{if } k = z_{i'j'} \text{ for some } (i', j') \neq (i, j) \\ \alpha\beta_{\text{new}} f_{\text{new}}(w_{ij}) p(y_i|\mathbf{z}^{-ij}, z_{ij} = k, \boldsymbol{\eta}^{\text{new}}), & \text{if } k = k^{\text{new}} \end{cases} \tag{6.9}$$

   where $\boldsymbol{\eta}^{\text{new}} = (\eta, \eta^{k^{\text{new}}})$, $\eta^{k^{\text{new}}} \sim \text{N}(0, \zeta)$, f_k is the distribution of the word given the other words allocated to topic k and f_{new} is the probability of the word in an empty topic.

   If a new topic k^{new} is sampled during one of the steps above, then draw $b \sim \text{Beta}(1, \gamma)$, set the new weight $\beta_{k^{\text{new}}} = b\beta_{\text{new}}$ and set the new β_{new} to $(1 - b)\beta_{\text{new}}$. b corresponds to the weight of the new atom that is instantiated from the Dirichlet process. Also, set $\boldsymbol{\eta}$ to the value of $\boldsymbol{\eta}^{\text{new}}$.

b) Sample m_{ik}, where k ranges over the topics, by generating n_{ik} uniformly distributed random variables $u_1, \ldots, u_{n_{ik}}$ between 0 and 1 and setting

$$m_{ik} = \sum_{m=1}^{n_{ik}} \mathbf{1}\left[u_m \geq \frac{\tau\beta_k}{\tau\beta_k + m}\right] \tag{6.10}$$

where $\mathbf{1}$ is the indicator function.

2. Sample $\boldsymbol{\beta}$ from $(\beta_1, \ldots, \beta_K, \beta_{\text{new}}) \sim \text{Dirichlet}(m_{\cdot 1}, \ldots, m_{\cdot K}, \gamma)$.

For a continuous response assuming $\gamma = 1$,

$$p(y_d|\mathbf{z}, \boldsymbol{\eta}) \propto \exp(-(y_d - \boldsymbol{\eta}^\top \bar{\mathbf{z}})^2) \tag{6.11}$$

and for a binomial response where $y_d \in \{0, 1\}$,

$$p(y_d|\mathbf{z}, \boldsymbol{\eta}) = (\boldsymbol{\eta}^\top \bar{\mathbf{z}})^{y_d}(1 - \boldsymbol{\eta}^\top \bar{\mathbf{z}})^{1-y_d}. \tag{6.12}$$

During prediction, the posterior of $\bar{\mathbf{z}}$ is needed over the test documents. This is calculated by removing the terms that depend on the response y from the conditional distributions so that inference on the test documents is identical to unsupervised sHDP. The posterior for the test samples can be sampled by replacing (6.9) with

$$p(z_{ij} = k|\mathbf{z}^{-ij}, w_{ij}, \boldsymbol{\beta}) \propto \begin{cases} (n_{ik}^{-ij} + \alpha\beta_k) f_k^{-w_{ij}}(w_{ij}), & \text{if } k = z_{i'j'} \text{ for some } (i', j') \neq (i, j) \\ \alpha\beta_{\text{new}} f_{\text{new}}(w_{ij}), & \text{if } k = k^{\text{new}} \end{cases} \tag{6.13}$$

and sampling the allocations and counts for the test documents.

6.8.2 *The ksLDA model*

For the keyword-supervised LDA (ksLDA) model, the coefficients for the words $\boldsymbol{\eta}^w$ are first estimated while the topic coefficients $\boldsymbol{\eta}^t$ are set to 0 and $\boldsymbol{\eta}^w$ is then fixed. During sampling, $\boldsymbol{\eta}^t$ is estimated after several rounds of sampling the topics. This allows the coefficients $\boldsymbol{\eta}^w$ to be learnt for the effect of the different words on the documents response and then for the topic coefficients $\boldsymbol{\eta}^t$ to be learnt to perturb the effects of the words according to the context of the document. The initial word coefficients can be learnt through a L_2 or LASSO regression of the document's response on the empirical word distribution. The topics can be learnt with either collapsed Gibbs sampling

or variational mean field inference. For a Gaussian model for the document's response this turns out to be equivalent to learning a regression model and then modeling the residuals using the topics. However, this may not be the case for other document responses.

Gibbs sampling proceeds as in unsupervised LDA with an additional term for the document's response.

1. For each observation j in document i, sample z_{ij} for the topic allocation, where T is the number of topics, V is the size of the vocabulary, n_{ik} is the number of words in document i allocated to topic k, n^{w}_{wk} is the number of times word w has been allocated to topic k, α^{w} is the concentration parameter for the prior Dirichlet distribution for the topic parameters and a superscript $-ij$ denotes the variable while ignoring the current allocation z_{ij},

$$p(z_{ij} = k|\alpha, \boldsymbol{\eta}, w, \mathbf{z}^{-ij}) \propto \frac{n_{ik}^{-ij} + \alpha}{n_{i\cdot}^{-ij} + T\alpha} \frac{n_{w_{ij}k}^{w,-ij} + \alpha^{w}}{n_{\cdot k}^{w,-ij} + V\alpha^{w}} p(y_d|\mathbf{z}^{-ij}, z_{ij} = k, \boldsymbol{\eta}, w) \quad (6.14)$$

For a continuous response assuming $\gamma = 1$,

$$p(y_d|\mathbf{z}, \boldsymbol{\eta}) \propto \exp(-(y_d - \boldsymbol{\eta}^{t\top}\bar{\mathbf{z}} - \boldsymbol{\eta}^{w\top}\overline{\mathbf{w}}_d)^2) \quad (6.15)$$

and for a binomial response where $y_d \in \{0, 1\}$,

$$p(y_d|\mathbf{z}, \boldsymbol{\eta}) = (\boldsymbol{\eta}^{t\top}\bar{z} + \boldsymbol{\eta}^{w\top}\overline{\mathbf{w}}_d)^{y_d}(1 - \boldsymbol{\eta}^{t\top}\bar{\mathbf{z}} - \boldsymbol{\eta}^{w\top}\overline{\mathbf{w}}_d)^{1-y_d}. \quad (6.16)$$

During prediction, as for the sHDP, the posterior of $\bar{z}$ is needed over the test documents. The inference process for these test documents is identical to unsupervised LDA. The posterior for the test documents can be sampled by replacing (6.14) with

$$p(z_{ij} = k|\alpha, \boldsymbol{\eta}, w, \mathbf{z}^{-ij}) \propto \frac{n_{ik}^{-ij} + \alpha}{n_{i\cdot}^{-ij} + T\alpha} \frac{n_{w_{ij}k}^{w,-ij} + \alpha^{w}}{n_{\cdot k}^{w,-ij} + V\alpha^{w}} \quad (6.17)$$

and sampling the allocations and counts for the test documents.

6.8.3 *Parameter estimation and prediction*

The topic regression coefficients are estimated after each round of sampling the topic assignments. I also performed experiments where the topic assignments were sampled for several rounds in between estimating the regression coefficients. But this made little

difference to prediction performance. The topic coefficients can be updated in the same way for both ksLDA and sHDP by regressing only on the topics that are allocated to at least one observation in the sHDP. I will describe cases for a Gaussian and binary response in this section, though other models for the response can be used too.

Gaussian model

To improve computation speed, instead of fully sampling from the posterior of the regression parameters, I optimise the parameters and find the maximum a posteriori (MAP) values. This can be found by rewriting the model response as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\eta} + \mathbf{c} \tag{6.18}$$

where $\mathbf{y}$ is a length-D vector of document responses, $\mathbf{X}$ is a $D \times \infty$ matrix of cluster to document allocation counts, $\boldsymbol{\eta}$ is a vector of regression parameters for each topic and $\mathbf{c}$ are the residuals. Let $\mathbf{X}$ be the matrix where row d is the empirical topic distribution for document d. Since only a finite number of topics have non-zero counts in the corpus, the columns in $\mathbf{X}$ that have zero counts and their corresponding $\boldsymbol{\eta}$ entries can be ignored, so making the optimisation tractable. For ksLDA, $\mathbf{X}$ is a $D \times K$ matrix where $K = T+V$. I found in preliminary experiments that in calculating $\mathbf{X}$ for ksLDA, averaging the empirical topic distribution for each document across M samples produces much better predictions of the document response than using the empirical distribution derived from a single sample.

The MAP solution for the parameters $\boldsymbol{\eta}$ while minimising $\mathbf{c}$ can be calculated numerically by

$$\boldsymbol{\eta} = \left(\mathbf{X}^\top\mathbf{X} + \zeta\mathbf{I}\right)^{-1}\mathbf{X}^\top\mathbf{y} \tag{6.19}$$

where ζ is the prior variance for the concentration parameters and $\mathbf{I}$ denotes the identity matrix. When $\zeta = 0$, this reduces to standard least squares. From preliminary experiments, I found that increasing M improves the estimation of the coefficients.

For prediction, topics are sampled for test documents as in (6.13) for sHDP and (6.17) for ksLDA. The empirical topic distribution is sampled over a number of iterations with any topics that are instantiated or any topics that are removed during this period ignored. The remaining empirical topic distributions for each document are averaged and used to calculate the expectation of the response.

For the sHDP model, this is calculated as

$$\mathrm{E}[\mathbf{y}|\mathbf{z}, \boldsymbol{\eta}] \approx \boldsymbol{\eta}^\top \mathrm{E}[\bar{\mathbf{z}}]. \tag{6.20}$$

For the ksLDA model, the empirical word distributions are also used for prediction:

$$\mathrm{E}[\mathbf{y}|\mathbf{z}, \mathbf{w}, \boldsymbol{\eta}] \approx \boldsymbol{\eta}^\top(\mathrm{E}[\bar{\mathbf{z}}], \overline{\mathbf{w}}). \tag{6.21}$$

Binomial model

For the logistic regression GLM model, we seek to find the parameters $\boldsymbol{\eta}$ which maximises the objective (likelihood function). The likelihood in the case of the sHDP model is

$$l(\boldsymbol{\eta}) \propto -\sum_{d=1}^{D} \log(1 + \exp(-y_d \boldsymbol{\eta}^\top \bar{\mathbf{z}}_d)) - \frac{\zeta}{2}\boldsymbol{\eta}^\top\boldsymbol{\eta}. \tag{6.22}$$

The gradient is

$$\nabla_{\boldsymbol{\eta}} l(\boldsymbol{\eta}) = \sum_d (1 - \sigma(y_d \boldsymbol{\eta}^\top \bar{\mathbf{z}}_d)) y_d \bar{\mathbf{z}}_d - \zeta\boldsymbol{\eta} \tag{6.23}$$

where $\sigma(\cdot)$ is the sigmoid function,

$$\sigma(x) = \frac{1}{1 + \exp(x)}. \tag{6.24}$$

A similar objective with the addition of the words and coefficients in the linear predictor is used for estimating the ksLDA parameters. Since the objective function is not analytically tractable, the limited-memory BFGS algorithm can be used to find the MAP estimate of the parameters (Minka, 2003). This is a quasi-Newton optimisation method that approximates the inverse Hessian matrix by analysing gradient vectors.

For prediction, topics are sampled for test documents as in (6.13) for sHDP and (6.17) for ksLDA as for the binomial model.

For the sHDP model, the distribution of the response is given by

$$p(y_d = 1|\mathbf{z}, \boldsymbol{\eta}) \approx \frac{\exp(\boldsymbol{\eta}^\top \mathrm{E}[\bar{\mathbf{z}}])}{1 + \exp(\boldsymbol{\eta}^\top \mathrm{E}[\bar{\mathbf{z}}])} \tag{6.25}$$

and similarly for the ksLDA model, it is given by

$$p(y_d = 1|\mathbf{z}, \mathbf{w}, \boldsymbol{\eta}) \approx \frac{\exp(\boldsymbol{\eta}^\top(\mathrm{E}[\bar{\mathbf{z}}], \overline{\mathbf{w}}))}{1 + \exp(\boldsymbol{\eta}^\top(\mathrm{E}[\bar{\mathbf{z}}], \overline{\mathbf{w}}))}. \tag{6.26}$$

6.9 EXPERIMENTS

I conducted experiments on four real-world problems. Firstly, the problem of classifying financial newswires based on their effect on the direction of change of the closing prices of a set of stocks. Secondly, the problem of classifying movie review sentences based on whether the review was classified as positive or negative. Thirdly, the regression problem of predicting a rating for a full movie review and finally the regression problem of predicting the popularity of a document. Following David M. Blei and McAuliffe (2007), the datasets were preprocessed to keep the words with the highest total TF-IDF score. TF-IDF is a measure of how important a word is for a document in a corpus. The score is calculated as $\mathrm{tf}(w) \times \log D/n_w$ where tf is the frequency of the word w in the document, D is the number of documents and n_w is the number of documents where the word w occurs. This is summed across all the documents for each word, and the highest scoring words are kept.

The newswire dataset consists of a set of real-world newswires extracted from Reuters about the stocks in the S&P 500 on different days over a year up to May 2011. The newswires were labelled with the companies that were mentioned in the wire. These labels were used so that only newswires whose stocks on days that had more than an 8% positive change or 3% negative change from the previous day were considered. These cutoffs were chosen so that the numbers of declining stocks were similar to the number of rising ones, and to ignore minor changes of prices due to other factors. This resulted in a dataset of 1,518 documents and a vocabulary of 1,895 words. The review snippet classification dataset (Pang and L. Lee, 2005) consists of reviews from the Rotten Tomatoes website with reviews that were marked as fresh labelled as positive reviews and reviews that were marked as rotten labelled as negative reviews. The dataset contains 5,331 positive snippets with the same number of negative ones and a vocabulary of 4,310 words.

The review snippet regression dataset (Pang and L. Lee, 2005) consists of reviews written by four film critics where the writer additionally assigned a rating to his or her review. The ratings were normalised to be between 0 and 1. Following David M. Blei and McAuliffe (2007), any words that appeared in more than 25% of the documents were removed as were any words that appeared fewer than 5 times. Only the remaining top 2,179 words by TF-IDF score were then kept. The ratings for each document were preprocessed to normalise the scores by applying a logit transform. There was a total of 5,005 documents with a vocabulary of 2,179 words. The document popularity regression dataset is a dataset of submission descriptions from the Digg website with the

associated number of votes that each submission received. The number of votes were again normalised by applying a logit transform.

Experiments were performed with the sHDP model, the keyword-supervised LDA model and the sLDA model.

The accuracy for classification problems and predictive R^2 for regression problems after five-fold cross-validation were calculated. Predictive R^2 is defined as

$$pR^2 = 1 - \frac{\sum_d (\hat{y_d} - y_d)^2}{\sum_d (y_d - \bar{y})^2}, \tag{6.27}$$

where y_d are the observed responses, with d ranging over the documents, $\hat{y_d}$ is the response predicted by the model and $\bar{y} = 1/D \sum_{d=1}^{D} y_d$ is the mean of the observed responses. This value gives the proportion of variability in the data set that is accounted for by the model and is often used to evaluate the goodness of fit of a model. A value of 1.0 is obtained when the regression line perfectly fits the data.

In the experiments, the prior standard deviation of the parameters ζ was tested with several values on the training set and the best one chosen. For ksLDA, the best value for ζ^w was found on the training set using a penalised regression of the response on the empirical word distributions. α for the ksLDA model was set to $50/K$, where K is the number of topics, similar to previous experiments with LDA. α^w for the sHDP model was set to 0.01 similar to previous experiments with HDP. In the sHDP, the standard prior Gamma(1, 1) was placed on α and γ and these are sampled during inference. For ksLDA, the regression coefficients were optimised after every round of sampling the topic allocations for a total of 20,000 iterations. From looking at the residuals and weights of the regression coefficients, the Markov chain appears to have converged after this many iterations. For sHDP, learning took place over 2,000 iterations with the coefficients being optimised every iteration. For predicting the responses of the test documents in ksLDA and sHDP, 500 iterations of topic sampling were used to allow the inferred topics to converge. The number of iterations was chosen by looking at the trace plots of the residuals, which appeared to converge by that number of iterations. To compare my models, I also carried out experiments using supervised LDA with 40 topics. This number of topics was found by splitting the training set in each cross-validation fold into further smaller training and testing sets. Different numbers of topics were tested on these smaller training and testing sets to avoid leaking information from the test sets. The models with 40 topics most often produced the best predictions on the training set. Since in the HDP and the sHDP, the posterior distribution over topics for each document is different than in LDA and sLDA, regression performance may be different between these models even if the posterior number of topics is the same.

6.9.1 *Results*

Table 6.1 shows that my supervised HDP (sHDP) model performs somewhat better than the sLDA model on the classification tasks. It performs roughly similar to sLDA on the regression tasks. The better performance of sHDP for classification is likely due to the increased flexibility of the model and faster mixing during inference. The increased flexibility comes from the model having an infinite number of topics to model the documents and responses. The faster mixing is because the inference process allows the model to instantiate clusters or remove unneeded clusters during sampling. Since newly instantiated clusters are empty, it is easier for words to change topic and be allocated to a new cluster. In contrast, in sLDA each topic almost always has a significant number of words allocated to it, making it difficult for the distribution of a topic to change. This has the effect of smoothing over word contributions for each topic. For regression, this is less important for sHDP as smoothing over word contributions affects performance less than the hard boundaries used in classification. Thus, the more specific topics in the sHDP model helps to improve classification performance.

From the combination of relatively low accuracy scores and large standard deviations, it can be seen that the newswire dataset is much harder than the movie review dataset. The standard deviations for the newswire scores imply that the data is much more noisy since newswires only indirectly influence stock movements. In addition, only closing stock prices were available, which means that it is possible there were changes in the stock price from the general movement of the industry or the market. The performance increase with ksLDA compared with sHDP is less for the newswires than with the movie reviews, implying that the words used in newswires are not very indicative of stock movement direction as compared with the words in movie reviews being indicative of the movie rating.

The keyword-supervised LDA (ksLDA) model performs much better than sLDA on both the regression and classification tasks. This is because the topics learnt by sLDA are too broad to pin down the contributions to document responses from individual words. Examining the coefficients for individual words shows a wide range between words that have the largest and smallest coefficients, whereas in sLDA, word contributions are smoothed out over the other words in the document that happen to be assigned to the same topic.

The prediction results of sLDA and ksLDA with different amounts of training data are shown in Table 6.2. Reducing the amount of training data significantly reduces the performance of sLDA, whereas there is a much smaller effect on the performance of ksLDA. In addition, ksLDA trained on a small subset of the training set can outperform

Table 6.1: 5-fold cross validation results from two movie review, one financial news and one document popularity dataset. The models used are the supervised HDP model, the supervised LDA model and the keyword-supervised LDA model. The number of topics for the LDA type models were set to 40.

Dataset	sHDP	ksLDA	sLDA
Classification		% accuracy	
Newswires	58.6 ± 3.2	60.2 ± 2.1	52.8 ± 7.3
Movie snippets	71.6 ± 1.5	76.0 ± 0.9	69.4 ± 1.0
Regression		Predictive R^2	
Movie reviews	0.324 ± 0.023	0.476 ± 0.026	0.341 ± 0.022
Document popularity	0.068 ± 0.019	0.097 ± 0.024	0.056 ± 0.022

sLDA trained on the full training set. This shows that many fewer documents are needed to learn the effect of words on document responses than are needed to learn a set of topics and the effect of those topics on document responses. This is likely because similar positive words and negative words are being used by the authors across the corpus allowing the effect of the words to be learnt from a relatively small dataset. On the other hand, a small dataset is insufficient to learn specific enough topics that are predictive of the document response. A dataset such as the document popularity dataset would probably have a greater effect on ksLDA as the size of the training set shrinks.

ksLDA improves faster than sLDA as the training set grows because a larger training set means that the model will observe more of the vocabulary in the corpus that affect document responses. As more of the relevant vocabulary is observed, the improvement in performance diminishes. On the other hand, sLDA's performance increases much more slowly since topic models typically require a large number of documents to learn good topics. This is a harder problem as every topic in the model has its own distribution over the vocabulary.

A comparison of the effect of different numbers of topics for the two parametric models is shown in Table 6.3. In ksLDA, as different words in a document have a direct effect on the document's response, fewer topics are required to explain the remaining variation in the response. This can be seen in the results since the number of topics affects the performance of sLDA more significantly than ksLDA. Decreasing the number of topics in the model affects the performance of both models more than reducing the amount of training data, showing that training set size is still much more important for model performance than model complexity. sLDA is much more prone to underfitting if there are insufficient topics to model the documents. The result of this is most

Table 6.2: Results comparing the use of different numbers of documents for the training set for the movie review dataset. The results are given as predictive R^2. The models used are the the supervised LDA model and the keyword-supervised LDA model. The number of topics was set to 20.

Training set size	ksLDA	sLDA
1000	0.368	0.165
2000	0.428	0.177
3000	0.465	0.183

Table 6.3: Results comparing different numbers of topics for the movie review dataset. The results are given as predictive R^2. The models used are the the supervised LDA model and the keyword-supervised LDA model.

# of topics	ksLDA	sLDA
10	0.458	0.184
20	0.465	0.183
30	0.460	0.247
40	0.476	0.341

likely that the words which have the biggest effect on the document response are mixed with words which have a small effect since there are not enough topics to separate them, causing the word effects to be smoothed over. Increasing the number of topics increases iteration time and increasing to more than 40 topics does not improve the results and increasing past that gradually degrades the results.

Additionally, I carried out experiments with the method of inference used in the original sLDA model, variational mean field inference, instead of collapsed Gibbs sampling. I found, however, that this produced slightly poorer results than Gibbs sampling.

6.9.2 *Analysis of strong topics and words*

For the sHDP model, the top positive and negative topics, in terms of their regression coefficients and their most frequent words for the movie review problem, are shown in Table 6.4. The topics do not generally correspond to themes such as film genre or style. Instead of this, the topics contain names and other unrelated words such as the function words *instead* and *appear*. This is because the flexibility of a nonparametric model means that the top positive and negative topics consist of very few words and are allocated to actors and directors that are consistently reviewed well or poorly. This flex-

ibility results in strong topics that are grouped along consistently performing actors or directors but which are less coherent since they are associated with so few documents. Topics that consist of more words, even if those are strong words, generally have smaller regression coefficients since the effect of the different words is averaged over other words in the same topic. Strong words are spread among the top positive and negative topics, for example, positive topic 5 contains the positive word *charming* and negative topic 2 contains many negative words such as *unfortunately*, *worse* and *problem*. Since many of the topics have actor and director names such as *Tom Hanks* in positive topic 2 and the *Coen brothers*: *Ethan* and *Joel* in positive topic 4, it can be seen that specific actors and directors are associated with consistenly better or poorer movie review scores.

The top positive and negative topics for the newswire problem and their most frequent words are given in Table 6.5. These topics are more cohesive than those for the movie review dataset. The top positive topic contains very strong positive words such as *higher*, *strong*, *rise* and *record*, which all imply good stock performance. The top negative topic also contains strongly negative words such as *cut*, *fall*, *decline* and *drop*, which clearly indicate bad performance. Similarly to the top topics for the movie review dataset, it can be seen that some industries consistently have better or poorer stock performance. For example, negative topic 2 consists of companies such as *prudential* and *metlife* along with words such as *insurers* and *insurance* that indicate this industry is performing badly. Positive topic 2 with words such as *defense*, *military* and *shareholders* indicates that companies involved with the military and defence are performing well.

For the ksLDA model, the words with the largest regression coefficients η^w for the movie review regression dataset are given in Table 6.6. These coefficients were estimated at the start of inference and then kept fixed. This categorisation of the words can be seen to be much clearer and more accurate than the topics learnt in the parametric models where word contribution is not directly modelled. For example, the strongest negative words *worst*, *awful*, *dull* and *unfortunately* are words that can typically be expected in bad reviews and the strongest positive words *perfect*, *hilarious*, *brilliant* and *stunning* are words that are typically expected in good reviews. The word *cinematographer:* unexpectedly appears as one of the strongest negative words but this is likely due to an error in the dataset. For example, negative reviews may be parsed wrongly and so include some metadata fields like cinematographer as part of the review text. Some words that are typically associated with a particular genre may have higher coefficients than expected when that genre has its own vocabulary for judging the merits of

Table 6.4: The most positive and negative learnt topics, in terms of their regression coefficients, from the movie review regression dataset with sHDP.

+ Topic 1 (8.3)	+ Topic 2 (6.5)	+ Topic 3 (6.1)	+ Topic 4 (6.1)
jeff	tom	philip	ethan
philip	hanks	calls	brothers
lane	roth	happiness	joel
write	tim	features	journey
miller	store	baker	singing
party	eric	helen	constant
cameron	speed	jane	blake
kate	rob	human	process
bus	wallace	hoffman	wonderfully
instead	appear	feelings	george
+ Topic 5 (5.8)	**− Topic 1 (-6.2)**	**− Topic 2 (-6.0)**	**− Topic 3 (-5.4)**
six	beneath	that's	dogs
aaron	child	least	aaron
neil	series	supposed	score
howard	son	watching	animal
matt	someone	unfortunately	golden
company	kills	lot	air
teacher	flaws	flat	martin
nature	winner	worse	ball
charming	onto	pretty	dog
buddy	record	problem	charles
	− Topic 4 (-4.1)	**− Topic 5 (-4.0)**	
	rachel	nelson	
	breaking	daughter	
	anthony	roger	
	ten	christopher	
	harry	con	
	warner	bergman	
	thinks	travolta	
	quinn	leslie	
	strikes	flashback	
	dog	simon	

Table 6.5: The most positive and negative learnt topics, in terms of regression coefficients, from the newswires dataset with sHDP.

+ Topic 1 (20.5)	+ Topic 2 (15.9)	+ Topic 3 (15.4)	+ Topic 4 (15.1)
prices	itt	fedex	north
higher	defense	raised	america
since	publicly	outlook	american
high	water	june	second
level	shareholders	monday	latin
strong	segments	bellwether	six
rise	military	pace	europe
above	aerospace	housing	china
record	announcement	foreign	avon
hit	holds	whole	september
+ Topic 5 (14.7)	**− Topic 1 (-16.4)**	**− Topic 2 (-11.9)**	**− Topic 3 (-10.8)**
wells	cut	insurers	include
fargo	fall	prudential	closed
friday	declines	cuomo	whose
effect	third-quarter	military	mln
between	ended	benefits	range
larger	volume	accounts	nasdaq
repurchase	third	richard	raise
accounting	drop	metlife	among
scheduled	decline	insurance	between
bonds	tuesday	yields	trades
	− Topic 4 (-10.6)	**− Topic 5 (-10.3)**	
	july	goldman	
	nvidia	sachs	
	exxon	buy	
	symantec	rates	
	motor	list	
	semiconductor	stocks	
	cut	prices	
	unemployment	adjusted	
	japan	demand	
	largest	result	

Table 6.6: The most positive and negative words, in terms of regression coefficient, from the movie review regression dataset using the ksLDA model.

Positive Words	Coefficient	Negative Words	Coefficient
perfect	10.0	worst	−10.8
hilarious	6.9	least	−8.6
powerful	6.3	unfortunately	−8.5
yet	6.2	cinematographer	−7.3
brilliant	5.8	awful	−7.0
simple	5.8	interesting	−6.8
easy	5.6	acceptable	−6.7
fascinating	5.4	dull	−6.6
complex	5.3	worse	−6.6
subtle	5.3	ridiculous	−6.5

a movie. An example is the word *hilarious*, which is the second strongest positive word most likely because it only appears in reviews for comedies.

An interesting distinction between these words and those in the top positive and negative topics in Table 6.4 is that both strong negative words such as *worse* and *unfortunately* and weaker negative words such as *problem* and *flat* are assigned to the same topic and therefore have the same coefficient in the parametric models. This shows that these models smooth over the effect of different words on the document response. Interestingly, the strongest positive and some of the strongest negative words do not appear in the top words for the top negative and positive topics, which again indicates that those word effects are being smoothed. On the other hand, the words in the top topics for sHDP seem to correspond to people with names such as *cameron* and *miller* with only one topic focusing on words that intuitively should have a strong contribution to a movie rating. This shows that the topics being learnt are divided into those that correspond to the content of the corpus and those that are more focused on general words that affect the rating of a movie.

The top topics and their most frequent words that are learnt from the ksLDA model are shown in Table 6.7. These topics are much more focused around movie genres rather than words that indicate a movie rating, except for negative topic 1. One reason for this is that the regression coefficients for these topics act as a perturbation on the effect of the words in the topic on the document's response. This means that a topic with a high regression coefficient does not indicate that the topic will predict a high positive response. This is because the response will also be affected by the word regression

coefficients for the different words in that topic. The effect of these topic coefficients can be thought of as perturbing the effect of the words on the response according to the context of the words. The top positive topic shows that movies about families adjust the rating positively compared to the individual words allocated to the topic. For example, words in the topic such as *children* and *father*, which are relatively neutral when taken across all contexts, have a much more positive effect on the review score in the context of a movie about a family, which are associated with words such as *lives*, *tragedy* and *powerful*. Negative topic 1 mostly consists of words that would be expected to indicate a bad movie, however, even in this case, normally neutral words such as *watch* and *say* have a much more negative effect on the review score in the context of other negative words such as *awful* and *worst*. This change in the effect of a word depending on context is not as obvious from the sHDP model where the most positive topic and negative topics have no association with film genres and is more concentrated on specific actors, which are more likely to perform consistently. On the other hand, the most negative topic learnt in ksLDA has no genre association. The other topics learnt by ksLDA have genre associations such as the drug culture in positive topic 2 and the horror genre in negative topic 2, indicating that these genres have a more positive and more negative effect on the review rating compared to the individual words in the topics. This shows that ksLDA was successful in learning both the effect of different words on the document responses and the context of words that push the rating in a direction such as those for certain genres.

The words with the largest learnt regression coefficients from the Reuters dataset are given in Table 6.8. It can immediately be seen from the coefficients that performance was poor in the third-quarter of the financial year and good in the fourth quarter and september. These words likely appear in the most negative and positive words as they are often used when a company publishes their quarterly reports, one of the most significant events that affect a company's stock price. Other negative words such as *fall*, *lower* and *weak* indicate a company is performing poorly whereas *raised*, *strong* and *jumped* indicate a company is performing well. More general words such as *bank* and *news* are also in the list of most positive and negative words indicating that stock price movements are relatively noisy as expected. In addition, news releases usually affect stock prices the most within an hour of when they are first released whereas only closing stock prices are available so often a newswire does not directly affect the movement of the stock price or may be reporting past movement of the price and the price may instead be affected by the general market or industry trends.

The topics that are learnt through ksLDA for the Reuters dataset shown in Table 6.9 are more general and less coherent than those in the movie rating dataset. The top posit-

Table 6.7: The strongest learnt topics, in terms of their regression coefficients, from the movie review regression dataset using the ksLDA model. The topic coefficients and +, − signs indicate a perturbation on the effect of the words in the topic.

+ Topic 1 (1.0)	+ Topic 2 (0.5)	+ Topic 3 (0.4)	+ Topic 4 (0.4)
mother	black	french	jack
father	white	sexual	cast
family	social	paris	scott
powerful	drug	sex	bill
emotional	culture	relationship	tom
child	group	affair	george
tragedy	case	english	james
tale	political	subtitles	joe
children	against	husband	larry
lives	drugs	women	jim

+ Topic 5 (0.4)	− Topic 1 (−2.9)	− Topic 2 (−0.7)	− Topic 3 (−0.7)
lawyer	least	horror	english
trial	actually	blood	subtitles
case	awful	special	cinematography
law	stupid	gore	beautiful
court	thing	body	tale
justice	watch	dead	french
judge	say	effects	spanish
murder	worse	evil	together
performances	that's	killed	named
compelling	worst	genre	language

− Topic 4 (−0.4)	− Topic 5 (−0.3)
comedy	women
funny	actress
humor	sex
jokes	mother
laughs	she's
comic	husband
gags	jane
joke	jennifer
amusing	sexual
fun	sister

Table 6.8: The most positive and negative words, in terms of their regression coefficients, from the Reuters dataset learnt with the ksLDA model.

Positive Words	Coefficient	Negative Words	Coefficient
raised	17.4	fall	−20.8
strong	16.0	bank	−12.6
rise	12.5	commission	−11.9
jumped	12.1	says	−11.7
fourth	11.3	lower	−11.2
above	11.2	tuesday	−10.8
high	10.5	under	−9.9
news	10.4	third-quarter	−9.9
morgan	10.0	statement	−9.6
september	9.8	weak	−9.5

ive topic consists of words that mention drugs and the drug industry such as *treatment* and *medical*. It also has words such as *people* and *health*, which would normally be treated as negative words outside a drug context, whereas in this case, their association with this industry has a small positive change on the effect of the words. The top negative topic is focused on banks and loans with words such as *assets*, *loan*, *losses* and *bank*. Interestingly, the company *citigroup* is also included in this topic, whereas across all contexts, *citigroup* would have a positive effect on stock price. This indicates that the mentioning of *citigroup* in association with other banks, assets and loans will likely have a negative change on the effect of the word and shows how the effect of words can change in different contexts. The topics learnt are very focused on industries and markets such as the oil industry in positive topic 3, manufacturing industry in positive topic 4 and credit cards in negative topic 5 instead of words which are particularly positive or negative. The global economy was in general recovery near the start of 2011 so this positive topic 5 caused a slight positive change in word effects. Similarly to the movie rating dataset, this shows that ksLDA can learn markets, industries and contexts that trend in a certain direction with respect to their constituent words. The word coefficients are more indicative of the direction of a stock rather than the context as can be seen by the low topic coefficients, this again is likely due to the noisy dataset.

In general, the topics learnt from ksLDA are more genre and content oriented than that in sHDP and indicate a perturbation on the effect of a word on the response depending on the word's context. As a result, the topics and topic coefficients have a slightly different interpretation to sHDP and sLDA as they are more dependent on the

Table 6.9: The strongest learnt topics, in terms of their regression coefficients, from the Reuters dataset with the ksLDA model. The topic coefficients and +, − signs indicate a perturbation on the effect of the words in the topic.

+ Topic 1 (0.5)	+ Topic 2 (0.4)	+ Topic 3 (0.4)	+ Topic 4 (0.3)
drug	comment	refinery	industrial
who	familiar	day	equipment
health	matter	monday	maker
drugs	source	oil	makes
medical	declined	per	power
people	immediately	thursday	manufacturing
patients	citing	spokesman	europe
food	street	regulators	electric
treatment	wall	late	european
known	sources	barrel-per-day	economic
+ Topic 5 (0.3)	**− Topic 1 (−1.0)**	**− Topic 2 (−0.7)**	**− Topic 3 (−0.4)**
china	bank	killed	reporting
global	banks	safety	san
international	assets	agency	editing
india	loans	explosion	francisco
growth	loan	federal	major
asia	losses	caused	products
united	america	thursday	three
states	largest	workers	senior
australia	citigroup	being	told
south	mortgage	possible	european
	− Topic 4 (−0.4)	**− Topic 5 (−0.4)**	
	nyse	fees	
	euronext	card	
	bid	bill	
	nasdaq	reform	
	omx	debit	
	deutsche	credit	
	operator	fee	
	boerse	big	
	deal	visa	
	tuesday	charge	

constituent words of the topics. sHDP learns strong topics that are assigned to fewer words and indicate trends and tendencies at a lower level, for example, on the level of actors instead of genres. For ksLDA, changing the prior variance of the word and topic coefficients can change the balance of the effects from words and their contexts. This can be used if it is known that few of the words in a dataset can be directly related to a document's response, such as documents written to be subtle or with a wide vocabulary. In this case, the prior variance of the word coefficients can be reduced and that of the topic coefficients increased. On the other hand, if a set of documents is known to consist mostly of indicative words, such as when categorising documents according to their field, then the field-specific words in the document are likely to be more significant to the document's category than the mixture of topics in the document. In this case, the prior variance of the word coefficients can be increased and that of the topics reduced. This allows the model to be very flexible in combining the two sources of information.

In addition, existing sources of knowledge such as the dictionaries that are used in sentiment analysis problems can be easily incorporated into the ksLDA model by using them to set the word coefficients or to offset the coefficients based on the dictionary entries. The sHDP model can be useful when more specific trends or tendencies are sought and when there is a possibility of overfitting or underfitting due to the number of topics. Finally, preliminary tests with a non-parametric version of ksLDA did not have a significant difference in performance to ksLDA. This is likely because ksLDA is less sensitive to the number of topics in the model than sLDA.

6.10 CONCLUSIONS

I have presented a supervised Bayesian nonparametric model that handles grouped data. Each group of data has an associated response such as sentiment ratings or document popularity. The supervised HDP (sHDP) model learns latent topics that are predictive of document responses without having to choose a fixed number of topics, a deficiency in previous models such as supervised LDA. In those models, overfitting or underfitting can occur if the number of topics is unsuitable for the dataset. The strongest topics learnt in the sHDP are relatively low-level and are associated with fewer topics allowing their effect on the document response to be learnt easily. Regression and classification experiments were performed on real-world datasets and showed that the model performs better than supervised LDA on classification but similarly on regression. Inference in the sHDP remains simple and is an adaptation of that used in the HDP. The flexibility and ease of inference of the sHDP means it has potential uses in many applications. A more Bayesian approach can be taken by sampling over the posterior of the parameters

using Hamiltonian MCMC. Other inference techniques to improve performance can be explored such as variational inference (Asuncion et al., 2009). While the sHDP does not explicitly handle categorical outcomes, extra regression parameters for each topic can be added to do so.

I also presented a supervised Bayesian parametric model for grouped data that directly models the effect of different words on the document response. The keyword-supervised LDA model (ksLDA) learns latent topics and the effect of those topics and individual words on document responses. This prevents the effects from individual words being smoothed out among other words in the same topic as in supervised LDA. As a result, the topics learnt by ksLDA match the content of the corpus well, and the topics learnt perturb the effect of their constituent words on the document responses. In contrast, in sHDP and sLDA, topics are learnt that both describe document themes and consist of words that have a large effect on document responses. The ksLDA topic coefficients also allow a word to have different effects on the document response depending on the word's context. The model can also be easily extended to incorporate additional information such as authors, titles or other metadata. Inference can be performed using Gibbs sampling, an easier alternative to the original variational inference method used for supervised LDA. The model is also more resilient to different numbers of topics compared to sLDA so that a nonparametric extension of ksLDA does not produce significantly different performance. Finally, the model performs better at classification and regression tasks than the state of the art sLDA model.

The supervised models can also be applied to the unsupervised models in Chapters 4 and 5. For example, if additional document metadata is available such as the venue where a paper is published or the institution of the first author then these can be used to learn topics and research groups that are more distinguished along venue or institution lines. It may also be possible to use the latent group structure to help with general document metadata prediction by learning groups that are predictive of document labels. For example, a set of films that are distributed by a particular company may each have the same common mix of themes but expressed in diferent ways. In this case, the user may wish to predict the box office performance of a film which will be affected both by the mix of themes and the distributor of the film. Finally, if unique identifiers are available such as the email addresses of some of the authors, this could futher improve deduplication performance of the model by learning topics that are more predictive of the authors.

While sentiment analysis models such as Pang and L. Lee (2005) have a similar goal of predicting document labels, the models I propose in this chapter are more general than typical sentiment analysis models and do not require any bootstrap dictionary or

labels for the words. My models can additionally deal with a wide range of document response types through a generalised linear model and can easily incorporate additional information into its generative process. The models in this chapter are not restricted to document datasets as they can be used on other kinds of data. For example, topic models have previously been used on extracted image patches or image features by treating the patches or features as words selected from a dictionary of patches. Similarly, the models in this chapter can be used to predict the keywords of an image or the theme of an image by directly modelling the effect of individual patches and their contexts and by not requiring the number of topics to be fixed in advance.

7

COMPARISON OF MODELS

The unsupervised and supervised models described in the previous chapters all involve the joint modelling of topics and document metadata. The metadata often consists of a list of author names but could also consist of labels such as categories and ratings. In this chapter, I will explore some of the differences between the models I proposed in earlier chapters.

7.1 TEXT AND METADATA

Each of the models I presented treats document metadata in different ways. In the author-topic models, the document metadata are variable-length lists of names that refer to latent entities. The proposed author-topic space model in Chapter 4 uses *n*-gram name variation models that have conjugate priors. This enabled relatively simple inference compared with the more complex domain-specific name variation model in Chapter 5. This domain-specific name variation model has the advantage of being able to infer canonical names for each entity. This is not possible when the name variation parameters are integrated out, such as in the *n*-gram models. The domain-specific model additionally allows easier incorporation of knowledge of how common a name is a priori. Information about the commonality of a name is important in determining the ambiguity of a name. For example, *John Smith* is a much more ambiguous name than *Alan Turing*. The *grouped author-topic* model additionally models the co-occurrence of these names and topics and so performs better than models that do not model their co-occurrence. In the proposed sHDP and ksLDA models in Chapter 6, the document labels are modelled with generalised linear models, allowing for a range of label types. Such labels are modelled by a function of the regression coefficients for each topic and in the ksLDA model also by a function of the word coefficients. Parallels can be drawn between the supervised models and the author-topic model for disambiguation where names are modelled as a function of the name parameters for each entity. A key difference is that in the supervised models, the labels are modelled as being non-exchangeable with the words and are conditional on the empirical topic distribution for each document. In the author-topic space model for disambiguation, the labels (names) and words are assumed to be exchangeable. This difference means that some authors

are not associated with any words and some words are not associated with any authors. On the other hand, in the supervised models, the topics all have an effect on the model of the document label.

The topics that are learnt in each of the Bayesian models are also different. In Chapter 4, each author entity had their own individual topic. We saw that this can cause problems both when an author is a prolific writer and when an author writes rarely. In these cases, the probability of each author being allocated to an author entity can be dominated by the number of words that are allocated to that entity so that the name of the author has little effect. In these situations, an author can also end up being allocated to many function words and so ends up being allocated to many words in the corpus. In Chapter 5, the number of topics learnt can vary significantly depending on the structure of the model. For example, the number of topics is usually fewer in the model where a document can be allocated to multiple latent groups than in the model where a document can only be allocated to a single group. This effect is the result of allowing words within the same document to be allocated to different groups. This flexibility means that a given word is preferentially allocated to latent groups that already have that word's topic allocated to them, and therefore new topics are rarely instantiated to explain that word. In both the models in Chapters 4 and 5, the topics that were allocated to the latent groups are relatively good indicators of the topics on which the group of authors write. Finally, the model without any groups has the fewest topics as there are no latent groups to help separate the topics. In Chapter 6, the topics learnt in the sHDP model are oriented around document labels. Since the topics are supervised in contrast to the previous models, the contents of the topics are affected by document labels. In the sHDP model, the supervised topics consist of both words based around themes (such as *comedy*) that tend to affect the document label and unrelated words (such as *excellent*) that also affect the document label but may not be coherent with the rest of the topic. In contrast, the supervised topics in the ksLDA model are based around themes that affect the document label, since the effect of the different words is modelled separately. This produces more coherent topics and allows the effect of different words on the document response to be dependent on context.

7.2 DP STRUCTURE

Even though the Bayesian nonparametric models are based upon the hierarchical Dirichlet process mixture model, the grouped author-topic model I presented in Chapter 5 used the nested Dirichlet process to add a layer of nesting. This extra layer is powerful because it allows random measures to be clustered, where the random measures

are over topics or entities. Each latent group in the *grouped author-topic* model can be viewed as a merge of the documents allocated to that group into one super-document. This can be useful for short documents, such as tweets, which are a type of message with strict character length limits. Topic models typically have difficulties in learning coherent topics from the few words available in such short documents. On the other hand, in the grouped author-topic model, good topics would in effect be learnt from the super-document, which could be a collection of some of the tweets by one person. The grouped author-topic model can represent a typical hierarchical Dirichlet process model by assigning each document to its own latent group. More complex methods of nesting can also be used when there is a priori knowledge of the relationships between the documents.

7.3 INFERENCE AND PARAMETERS

Posterior inference in the models presented is performed using approximate methods since it is infeasible to sum over all possible cluster assignments. In the case of the nonparametric models, collapsed Gibbs sampling is used after marginalising out the document-level random measures. I also investigated the effect of marginalising out the global random measure in Chapter 4 and found that this allowed the chain to mix faster than when the global random measure was sampled. However, there is a greater computational cost as both table and cluster allocations must be sampled for each data point. The other new models I proposed in Chapters 5 and 6 all sampled the global random measure instead of marginalising it out, making inference easier and faster. Variational mean-field inference for existing models such as sLDA and HDP could also be adapted for the supervised models in Chapter 6. Variational inference could result in faster inference; however, the method would introduce new problems such as local minima, which need to be tackled with random restarts.

Most of the parameters in the models I proposed were given a Bayesian treatment. Hyperparameters for the concentration parameters were chosen to have an a priori uniform distribution for the number of entities in the models. For the topic concentration parameters, standard hyperparameters were used so that there is a high probability of relatively few topics compared with the number of words in the corpus as in previous work with HDP-LDA. Concentration parameters were updated by sampling from their posterior distributions during inference. In Chapter 6, however, the generalised linear model parameters were optimised and set to their maximum a posteriori (MAP) values instead of being sampled. This was done to speed up model inference. I expect that this will not significantly influence the results compared with a fully Bayesian treatment.

An empirical Bayes approach was taken in Chapter 5 to set the prior probability that a name in the corpus is a canonical name for a real person.

7.4 PERFORMANCE

The models I proposed were all evaluated on real-world datasets. But due to the lack of publicly available disambiguation datasets, the models in Chapters 4 and 5 were also evaluated on artificial datasets made by conflating the names of a real-world citation database. The evaluation results showed that the proposed models perform better than some of the existing leading unsupervised approaches for name disambiguation. The proposed supervised models in Chapter 6 were also evaluated on real-world publicly available classification and regression datasets, though one of the classification datasets (the newswires dataset) was manually built from real-world data. The prediction scores from the evaluation showed that these proposed supervised models perform better than one of the leading supervised topic models.

7.5 EXTENT OF APPLICATIONS

Even though all the experiments performed with the proposed models involved documents and their free-text, it would be possible to apply these models to other types of data and problems. For example, for images, the analog of words in a document would be the image feature vectors in an image. The proposed models could then be used to analyse image metadata in the same way as document metadata, such as for predicting image labels or resolving ambiguous names or labels of objects in the image to real entities. The models can also be applied to documents where the metadata is not readily available but instead inferred from the text. This is often the case for named entity resolution, word sense disambiguation and coreference resolution problems.

A summary of the differences between the proposed models and some existing models is given in Table 7.1.

Table 7.1: A summary of the differences between the proposed models and some existing models where author-topic is shortened to *A-T*.

Model	Metadata type	Metadata ambiguity	Infinite model capacity	Supervised	Different words in same topic have different effects	Co-occurrence model-ling of metadata
A-T space	Name list	Yes	Yes	No	No	No
Grouped A-T	Name list	Yes	Yes	No	No	Yes
Supervised HDP (sHDP)	Any label	No	Yes	Yes	No	No
Keyword-supervised LDA (ksLDA)	Any label	No	No	Yes	Yes	No
A-T	True authors	No	No	No	No	No
LDA-ER	Name list	Yes	Partial	No	No	Yes
sLDA	Any label	No	No	Yes	No	No

8

CONCLUSIONS AND FUTURE WORK

With the quickly growing number of documents and document collections now available, there is an increasing need to be able to analyse them automatically. Probabilistic topic models are one approach which have had successes in many fields and applications, even for non-textual data. However, documents rarely consist of just free text and often come with additional metadata such as author information and labels. Instead of analysing this metadata separately from the text, it is becoming increasingly important to analyse the metadata and the text together so that inferences about one can help the other. This can uncover useful insights into the document collection. The problem of analysing and integrating multiple sources of information is further related to the field of multi-modal integration (Al-Hames and Rigoll, 2005). In this field, a Bayesian approach has a number of advantages, including being able to integrate different information easily where the prior accuracy or confidence for each source needs to be accounted for when making inferences.

In this thesis, I proposed several novel Bayesian models for name disambiguation and supervised learning. These models all utilise document metadata to both improve the topics that are inferred and predict document labels and authors. Three of the proposed models used Bayesian nonparametric techniques, allowing the model's complexity to adapt to the data and have an infinite capacity for authors or topics. The fourth model used words in addition to topics to model a document's label, allowing words to have different effects on the model of the label according to their context. The models were all evaluated on real-world collections, showing that the combination of models for text and metadata gives good results and has fruitful opportunities.

In Chapter 4, I developed the *author-topic space model for disambiguation*. This is a new Bayesian nonparametric model that integrates a name variant model and a topic model to perform the difficult task of resolving names into authors in the presence of name ambiguity and name variation. I explored a character-based bigram topic model and a bag-of-words trigram based model to model name variation and found that the bigram model performed better. I also compared the CRF with the direct Gibbs sampling inference methods and found that CRF performed slightly better at the cost of greater computational time. The author-topic space model can utilise the free text of a document in contrast to other existing disambiguation models.

There is much opportunity for further work with this proposed model. For example, the author list is modelled as an exchangeable list, which is rarely the case in most documents. To weaken that assumption, an additional Dirichlet process could be used to model the first author of documents differently from the rest of the authors. Ideas from the dynamic topic model could also be used to analyse how the position of an author in their author lists changes over time, for example, new professors gradually move to the end of author lists as they become more senior. The model could also be extended to different kinds of data, for example, replacing words in an image with bag-of-words image features and replacing authors with common keywords or categories. This allows the model to resolve the keywords with the aid of the image features, for example, the model can resolve the sense of the word *bank* based on the presence of a river bank or a high-street bank in the image.

In Chapter 5, I developed the *grouped author-topic model*, an extension of the work in Chapter 4. I also described the hybrid NDP-HDP, an extension of the NDP, and proposed the hierarchical extension to the hybrid NDP-HDP. These were used to develop the grouped author-topic model. This allows the model to have latent groups, each of which models the co-occurrence of latent author entities and topics. The model allows topics and author entities to be members of multiple latent groups, which the original NDP does not allow. The latent group structure can be thought of as research groups where a research group consists of a number of authors working on a set of topics and where these authors and topics may also be involved with other research groups. This prevents authors who write many documents in the corpus from being overweighted in the posterior. The model also uses a non-conjugate and domain-specific name variation model so that names are better modelled than in the n-gram models. Using this name variation model allows inferring which of the names in the corpus is the canonical name for an author entity. The grouped author-topic model was shown to perform better than models that do not take into account the co-occurrence of entities and was shown to perform well at name disambiguation.

For future work, other name variation models can also be explored as well as more principled ways of learning these variation models from training data. Canonicalisation can be improved by merging different names from the corpus into a canonical name or using methods to detect canonical names. This would be necessary, for instance, when the author's full name does not appear all at once in a document but is instead sometimes initialled in the corpus. In this case, the canonical name needs to be inferred from the supplied names. The proposed model can also be extended dynamically using dependent Dirichlet processes, to model the movement of author entities and topics in the latent research groups. For example, authors can move between different research

groups or institutions, or become a less prominent contributor in one research group. Authors interests can also change which may also be a result of moving to a different research group or a result of the research group changing its topic focus. Finally, research groups can be come more or less prominent over time and may also experience splits or merges with other research groups. The model can be applied to other applications such as named entity recognition or more general types of coreference resolution. In these settings, the names and references need to be extracted directly from the document text rather than already being available. It can be useful to explore the modelling of the co-occurrence of names, references and topics in latent groups and use them to improve results on those problems. It can also be useful to model types of data such as image and annotation data where there is a need to model the co-occurrence of image features or annotations. Finally, the research groups can be modelled using the Indian buffet process (IBP) (Tom L. Griffiths and Ghahramani, 2006). The Indian buffet process allows for models where objects such as latent research groups are represented with infinitely many binary features. In the context of the author and research group models, this would prevent the number of authors or topics in a latent group from overly influencing the probability of the group. This is because the research group would be represented by the presence or absence of the authors or topics rather than the number of times the author or topic has been allocated to that research group. This can be a better model for real research groups where high-quality publications that are important to a field or topic are more important than the number of publications.

In Chapter 6, I proposed two novel generative models for supervised learning with topic models. The *supervised HDP* (sHDP) model uses supervised topics to predict document labels without worrying about underfitting or overfitting due to a fixed number of topics. The *keyword-supervised LDA* (ksLDA) model better predicts document labels by directly modelling the effect of words on document labels. The ksLDA model performs better than the sHDP and sLDA model in classification and regression, though both sHDP and ksLDA perform better than sLDA at classification. The direct modelling of words in ksLDA also reduces the importance of the number of topics. The sHDP model performs better than sLDA at classification problems as these problems may be more sensitive to the number of topics. The ksLDA model learns topics that show how the context of different words changes the effect they have on document labels by modelling the effect of different words directly.

There are many opportunities for new applications and future work in these new models. For example, it would be interesting to see how different labels can influence the kinds of topics that are learnt in a document collection. This can be useful when the users of a topic model wish to learn topics around specific themes in the collection

where there are no existing labels corresponding to those themes. It would also be interesting to investigate more situations in which the direct modelling of the words has an advantage over using just the topics or just the words to prediction document labels. For example, for image problems, the direct modelling of the effect of image features on the image label may not give much of a benefit over modelling the effect of the topics to which the images are allocated. It would also be interesting to investigate a nonparametric ksLDA model to see if any of the advantages of the sHDP's flexibility can be carried over to the ksLDA. In addition, it would be interesting to see if adding unlabelled data to train the model can result in better topics and thus improve label predictions.

Supervised learning can be applied to the unsupervised models described in Chapters 4 and 5. This would allow more ways for document metadata to be used to perform name disambiguation. For example, the institution of authors, if available during training can help to learn topics and groups that improve disambiguation performance. Additionally, latent groups can be learnt that are more predictive of certain document metadata or that are more closely aligned with the kind of groups that the user wishes to discover, such as latent groups based around the type of publication (journal, conference, technical report), etc.

The use of Bayesian nonparametric mixture models in general and the advantages that they bring with their infinite capacity also brings risks. The two main parameters that influence the number of clusters in the models are the concentration parameter and the base measure. A naive uninformative prior on the concentration parameter, for instance, can result in a very informative prior for the number of clusters in the models. This may be suitable for topic models where only a small number of topics relative to the number of observations is preferred but becomes unsuitable when there may be a very high number of latent clusters compared to the number of observations. If this is the case, the prior can cause overclustering and it can be necessary to search for an alternative prior that puts a relatively uniform distribution on the number of clusters. The base measure also affects the number of clusters. For example, in topic modelling, the symmetric Dirichlet base measure for the topics influences the number of resulting topics. When the Dirichlet component values are high, topics are preferred that place mass on many words and few topics are needed to represent the corpus. On the other hand, when the component values are low, topics are preferred that place mass on few words and many topics are needed to represent the corpus. Convergence can also be an issue for inference in general Bayesian nonparametric models. Though these models can take many iterations to converge, for my models for name disambiguation, the disambiguation performance stabilises well before sampling has converged. This means

that it is beneficial to run preliminary experiments with a small number of iterations and to report results using the full set of iterations. Variational inference techniques may also improve the speed of inference with these models but is more complex to implement and only gives an approximation to the posterior.

The exchangeability assumptions on the authors and words are unrealistic but are appropriate for the purpose of disambiguation, especially since a name's position in the author list would not be helpful in disambiguation. However, the models I proposed assume a priori that each author contributes to a document equally whereas it is more likely that authors earlier in the author list should be more closely associated with the topics in the document. In general, the likelihood model for the names (or metadata in general) made the biggest impact on performance. But it can still be important to separate words and authors so that in the posterior, the words in the document cannot dominate the authors in the author list. More methods of integrating free text and metadata would probably yield fruitful results as would more work on choosing the correct base measure for a model. The models I proposed can be applied more generally to resolution problems especially in cases when free text and metadata (or other extracted information) need to be analysed together.

REFERENCES

Antoniak, Charles E. (1974). "Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems". In: *The Annals of Statistics* 2.6, pp. 1152–1174. DOI: 10.1214/aos/1176342871 (cit. on pp. 26, 27, 53).

Artiles, Javier, Andrew Borthwick, Julio Gonzalo, Satoshi Sekine and Enrique Amig (2010). "WePS-3 evaluation campaign: overview of the web people search clustering and attribute extraction tasks". In: *Conference on Multilingual and Multimodal Information Access Evaluation (CLEF)*. (Padova). Vol. 24. 2, pp. 243–265. URL: http://clef2010.org/resources/proceedings/clef2010labs_submission_114.pdf (cit. on p. 9).

Artiles, Javier, Julio Gonzalo and Satoshi Sekine (2007). "The SemEval-2007 WePS evaluation: establishing a benchmark for the web people search task". In: *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*. (Prague). Association for Computational Linguistics, pp. 64–69. URL: http://www.aclweb.org/anthology/W/W07/W07-2012 (cit. on p. 9).

Artiles, Javier, Julio Gonzalo and Satoshi Sekine (2009). "WePS 2 evaluation campaign: overview of the web people search clustering task". In: *2nd Web People Search Evaluation Workshop (WePS 2009), 18th WWW Conference*. (Madrid). URL: http://nlp.uned.es/weps/weps2/papers/weps2-clustering-task-description.pdf (cit. on pp. 9, 111).

Asuncion, Arthur, Max Welling, Padraic Smyth and Yee Whye Teh (2009). "On smoothing and inference for topic models". In: *Association for Uncertainty in Artificial Intelligence* Ml, pp. 27–34. URL: http://eprints.pascal-network.org/archive/00006729/ (cit. on p. 146).

Bagga, Amit and Breck Baldwin (1998a). "Algorithms for scoring coreference chains". In: *The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*. (Granada), pp. 563–566 (cit. on p. 55).

Bagga, Amit and Breck Baldwin (1998b). "Entity-based cross-document coreferencing using the vector space model". In: *Proceedings of the 17th international conference on Computational linguistics*. (Montreal). Association for

Computational Linguistics, pp. 79–85. DOI: 10.3115/980845.980859 (cit. on pp. 9, 109).

Baron, Alex and Marjorie Freedman (2008). "Who is who and what is what: experiments in cross-document co-reference". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. (Honolulu). Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 274–283. URL: http://portal.acm.org/citation.cfm?id=1613754 (cit. on p. 10).

Bhattacharya, Indrajit and Lise Getoor (2006). "A latent Dirichlet model for unsupervised entity resolution". In: *The SIAM International Conference on Data Mining (SIAM-SDM)*. (Bethesda, MD, USA) (cit. on pp. 7, 39, 56, 88, 89, 102).

Bhattacharya, Indrajit and Lise Getoor (2007). "Collective entity resolution in relational data". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1.1, p. 5. ISSN: 1556-4681. DOI: 10.1145/1217299.1217304 (cit. on p. 6).

Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning*. Ed. by M Jordan, J Kleinberg and B Scholkopf. Vol. 4. Information Science and Statistics 4. Springer, p. 738. ISBN: 0387310738. DOI: 10.1117/1.2819119 (cit. on p. 5).

Blackwell, David and James B. Macqueen (1973). "Ferguson distributions via Pólya urn schemes". In: *The Annals of Statistics* 1, pp. 353–355 (cit. on p. 50).

Blei, David M. and Michael I. Jordan (2003). "Modeling annotated data". In: *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval SIGIR 03*. (Toronto). Computer Science Division (EECS), University of California; Berkeley. ACM Press, p. 127. ISBN: 1581136463. DOI: 10.1145/860435.860460 (cit. on pp. 54, 72, 88).

Blei, David M., Michael I. Jordan and Andrew Y. Ng (2003). "Hierarchical Bayesian models for applications in information retrieval". In: *Bayesian Statistics 7*. Ed. by Jose M. Bernardo et al. OUP (cit. on p. 22).

Blei, David M and John D Lafferty (2006). "Dynamic topic models". In: *ICML '06: Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM, pp. 113–120. ISBN: 1-59593-383-2. DOI: http://doi.acm.org/10.1145/1143844.1143859 (cit. on p. 22).

Blei, David M. and Jon D. McAuliffe (2007). "Supervised topic models". In: *Advances in Neural Information Processing Systems 20*. (Vancouver). Ed. by John C. Platt, Daphne Koller, Yoram Singer and Sam Roweis. Vol. 20. 21. MIT Press, pp. 1–8 (cit. on pp. 115, 116, 120, 132).

Brooks, Steve, Andrew Gelman, Galin Jones and Xiao-Li Meng, eds. (2011). *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, p. 619. ISBN: 978-1420079418. URL: http://www.mcmchandbook.net/ (cit. on p. 17).

Buntine, Wray L. (1994). “Operations for learning with graphical models”. In: *Journal of Artificial Intelligence Research* 2, pp. 159–225 (cit. on p. 14).

Chang, Jonathan and David Blei (2009). “Relational topic models for document networks”. In: *AIStats* (cit. on p. 22).

Christen, Peter (2006). “A comparison of personal name matching: techniques and practical issues”. In: *Sixth IEEE International Conference on Data Mining Workshops ICDMW06*. (Hong Kong). IEEE, pp. 290–294. ISBN: 0769527027. DOI: 10.1109/ICDMW.2006.2 (cit. on pp. 5, 42).

Clayden, Jonathan D., Amos J. Storkey and Mark E. Bastin (2007). “A probabilistic model-based approach to consistent white matter tract segmentation”. In: *IEEE Transactions on Medical Imaging* 26, pp. 1555–1561 (cit. on p. 4).

Cohen, William W., Pradeep Ravikumar and Stephen E. Fienberg (2003). “A comparison of string metrics for matching names and records”. In: *ACM International Conference on Knowledge Discovery and Data Mining (KDD), Workshop on Data Cleaning, Record Linkage, and Object Consolidation, 2003*. (Washington). Vol. 3. S2. ACM, pp. 73–78 (cit. on p. 5).

Cowans, Philip J. (2004). “Information retrieval using hierarchical Dirichlet processes”. In: *SIGIR ’04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. (Sheffield). ACM, pp. 564–565. ISBN: 1-58113-881-4. DOI: 10.1145/1008992.1009122 (cit. on p. 36).

Culotta, Aron, Pallika Kanani, Robert Hall, Michael Wick and Andrew McCallum (2007). “Author disambiguation using error-driven machine learning with a ranking loss function”. In: *Sixth International Workshop on Information Integration on the Web (IIWeb-07)*. (Vancouver, Canada). URL: http://www.cs.umass.edu/$%5Csim$culotta/pubs/culotta07author.pdf (cit. on pp. 5, 39).

Dai, Andrew M. and Amos J. Storkey (2009). “Author disambiguation: a nonparametric topic and co-authorship model”. In: *NIPS Workshop on Applications for Topic Models Text and Beyond*. (Whistler), pp. 1–4. URL: http://eprints.pascal-network.org/archive/00007742/ (cit. on p. 3).

Dai, Andrew M. and Amos J. Storkey (2011). “The grouped author-topic model for unsupervised entity resolution”. In: *Artificial Neural Networks and Machine Learning - ICANN 2011*. (Espoo). Ed. by Timo Honkela, Włodzisław Duch, Mark Girolami and Samuel Kaski. Vol. 6791. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 241–249. ISBN: 978-3-642-21734-0. DOI: 10.1007/978-3-642-21735-7_30 (cit. on p. 3).

Daumé III, Hal and Daniel Marcu (Sept. 2005). "A Bayesian model for supervised clustering with the Dirichlet process prior". In: *Journal of Machine Learning Research* 6, pp. 1551–1577 (cit. on pp. 7, 39).

Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer and Richard Harshman (1990). "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6, pp. 391–407. ISSN: 1097-4571. DOI: 10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9 (cit. on p. 22).

Diederich, Joachim, Jörg Kindermann, Edda Leopold and Gerhard Paass (May 2003). "Authorship attribution with support vector machines". In: *Applied Intelligence* 19.1-2, pp. 109–123. ISSN: 0924-669X. DOI: 10.1023/A:1023824908771 (cit. on p. 6).

Dorazio, Robert M. (Sept. 2009). "On selecting a prior for the precision parameter of Dirichlet process mixture models". In: *Journal of Statistical Planning and Inference* 139.9, pp. 3384–3390. ISSN: 03783758. DOI: 10.1016/j.jspi.2009.03.009 (cit. on p. 31).

Dredze, Mark, Paul McNamee, Delip Rao, Adam Gerber and Tim Finin (2010). "Entity disambiguation for knowledge base population". In: *COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics*. (Beijing). Ed. by Chu-Ren Huang and Dan Jurafsky. Vol. 3. COLING '10. Tsinghua University Press, pp. 277–285 (cit. on p. 7).

Escobar, Michael D. and Mike West (1995). "Bayesian density estimation and inference using mixtures". In: *Journal of the American Statistical Association* 90, pp. 577–588 (cit. on pp. 30, 31, 53).

Fellegi, Ivan P. and Alan B. Sunter (1969). "A theory for record linkage". In: *Journal of the American Statistical Association* 64.328, pp. 1183–1210. JSTOR: 2286061 (cit. on p. 4).

Ferguson, Thomas S. (1973). "A Bayesian analysis of some nonparametric problems". In: *The Annals of Statistics* 1.2, pp. 209–230. ISSN: 00905364. JSTOR: 2958008 (cit. on p. 27).

Finin, Tim, Zareen Syed, James Mayfield, Paul McNamee and Christine Piatko (2009). "Using Wikitology for cross-document entity coreference resolution". In: *The AAAI Spring Symposium on Learning by Reading and Learning to Read*. (Stanford). AAAI Press, pp. 29–35. URL: http://www.aaai.org/Papers/Symposia/Spring/2009/SS-09-07/SS09-07-006.pdf (cit. on p. 10).

Finkel, Jenny Rose, Trond Grenager and Christopher Manning (2005). “Incorporating non-local information into information extraction systems by Gibbs sampling”. In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics ACL 05*. (Ann Arbor). Vol. 43. Association for Computational Linguistics, pp. 363–370. DOI: 10.3115/1219840.1219885 (cit. on p. 111).

Finkel, Jenny Rose and Christopher D. Manning (2008). “Enforcing transitivity in coreference resolution”. In: *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies Short Papers HLT 08*. (Columbus, June 2008). Association for Computational Linguistics, p. 45. DOI: 10.3115/1557690.1557703 (cit. on p. 8).

Fleischman, Michael Ben and Eduard Hovy (2004). “Multi-document person name resolution”. In: *ACL 2004: Workshop on Reference Resolution and its Applications*. (Barcelona, Spain). Ed. by Sanda Harabagiu and David Farwell. Association for Computational Linguistics, pp. 1–8. URL: http://acl.ldc.upenn.edu/W/W04/W04-0701.pdf (cit. on p. 9).

Gelman, Andrew and Donald B. Rubin (1992). “Inference from iterative simulation using multiple sequences”. In: *Statistical Science* 7.4, pp. 457–472. ISSN: 08834237. DOI: 10.1214/ss/1177011136 (cit. on p. 19).

Geman, S and D Geman (1984). “Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images”. In: *IEEE Trans. Pattern Analysis and Machine Intelligence* 6.6, pp. 721–741 (cit. on p. 18).

Geyer, Charles J. (1992). “Practical Markov chain Monte Carlo”. In: *Statistical Science* 7.4, pp. 473–483. ISSN: 08834237. DOI: 10.1214/ss/1177011137 (cit. on p. 19).

Giles, C. Lee, Kurt D. Bollacker and Steve Lawrence (1998). “Citeseer: an automatic citation indexing system”. In: *Digital Libraries 98 - The Third ACM Conference on Digital Libraries*. (Pittsburgh), pp. 89–98. URL: http://clgiles.ist.psu.edu/papers/DL-1998-citeseer.pdf (cit. on pp. 4, 6, 56, 99).

Gong, Jun, Lidan Wang and Douglas W. Oard (2009). “Matching person names through name transformation”. In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009*. (Hong Kong). Ed. by David Wai-Lok Cheung, Il-Yeol Song, Wesley W. Chu, Xiaohua Hu and Jimmy J. Lin. ACM, pp. 1875–1878. ISBN: 9781605585123 (cit. on p. 5).

Google (2004). *About Google Scholar*. URL: http://scholar.google.com/intl/en/scholar/about.html (cit. on p. 6).

Gooi, Chung Heong and James Allan (2004). "Cross-document coreference on a large scale corpus". In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. (Boston, Massachusetts, USA, May 2004). Ed. by Daniel Marcu Susan Dumais and Salim Roukos. Association for Computational Linguistics, pp. 9–16 (cit. on pp. 9, 109).

Griffin, J. E. and M. F. J. Steel (2006). "Order-based dependent Dirichlet processes". In: *Journal of the American Statistical Association* 101 (473), pp. 179–194 (cit. on p. 89).

Griffiths, Thomas L. and Mark Steyvers (2004). "Finding scientific topics". In: *Proceedings of the National Academy of Sciences of the United States of America* 101.Suppl 1, pp. 5228–35. ISSN: 00278424. DOI: 10.1073/pnas.0307752101 (cit. on p. 25).

Griffiths, Tom L. and Zoubin Ghahramani (2006). "Infinite latent feature models and the Indian buffet process". In: *Advances in Neural Information Processing Systems 18* (cit. on p. 155).

Haghighi, Aria and Dan Klein (June 2007). "Unsupervised coreference resolution in a nonparametric Bayesian model". In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. (Prague, Czech Republic). Association for Computational Linguistics, pp. 848–855. URL: http://www.aclweb.org/anthology/P/P07/P07-1107 (cit. on pp. 8, 98).

Haghighi, Aria and Dan Klein (2009). "Simple coreference resolution with rich syntactic and semantic features". In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 3 EMNLP 09*. (Singapore). Vol. 3. EMNLP '09. Association for Computational Linguistics, p. 1152. ISBN: 9781932432633. DOI: 10.3115/1699648.1699661 (cit. on p. 9).

Hall, Rob, Charles Sutton and Andrew McCallum (2008). "Unsupervised deduplication using cross-field dependencies". In: *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. (Las Vegas). ACM, pp. 310–317. ISBN: 978-1-60558-193-4. DOI: 10.1145/1401890.1401931 (cit. on pp. 7, 40, 76).

Al-Hames, Marc and Gerhard Rigoll (July 2005). "A multi-modal mixed-state dynamic Bayesian network for robust meeting event recognition from disturbed data". In: *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pp. 45–48. DOI: 10.1109/ICME.2005.1521356 (cit. on p. 153).

Hannah, Lauren, David Blei and Warren Powell (2011). “Dirichlet process mixtures of generalized linear models”. In: *Journal of Machine Learning Research* 12, pp. 1923–1953 (cit. on p. 118).

Hastings, W. K. (1970). “Monte Carlo sampling methods using Markov chains and their applications”. In: *Biometrika* 57.1, pp. 97–109. DOI: 10.1093/biomet/57.1.97 (cit. on p. 18).

Hofmann, Thomas (2001). “Unsupervised learning by probabilistic latent semantic analysis”. In: *Machine Learning* 42.1, pp. 177–196. ISSN: 08856125. DOI: 10.1023/A:1007617005950 (cit. on p. 22).

Hogan, Howard (1992). “The 1990 post-enumeration survey: an overview”. In: *The American Statistician* 46.4, pages. ISSN: 00031305. JSTOR: 2685308 (cit. on p. 4).

Huang, Jian, Sarah M. Taylor, Jonathan L. Smith, Konstantinos A. Fotiadis and C. Lee Giles (2009). “Profile based cross-document coreference using kernelized fuzzy relational clustering”. In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. (Singapore). Association for Computational Linguistics, pp. 414–422. ISBN: 9781932432459. DOI: 10.3115/1687878.1687937 (cit. on p. 10).

Indyk, Piotr and Rajeev Motwani (1998). “Approximate nearest neighbors: towards removing the curse of dimensionality”. In: *STOC ’98: Proceedings of the thirtieth annual ACM symposium on Theory of computing*. (Dallas). ACM, pp. 604–613. ISBN: 0-89791-962-9. DOI: 10.1145/276698.276876 (cit. on p. 5).

Jordan, Michael I. (2004). “Graphical models”. In: *Statistical Science (Special Issue on Bayesian Statistics)* 19.1, pp. 140–155. ISSN: 08834237. DOI: 10.1214/088342304000000026 (cit. on p. 14).

Kanani, Pallika, Andrew McCallum and Chris Pal (2007). “Improving author coreference by resource-bounded information gathering from the web”. In: *Proceedings of the International Joint Conference on Artificial Intelligence*. (Hyderabad). IJCAI, pp. 429–434 (cit. on p. 39).

Koppel, Moshe, Jonathan Schler and Shlomo Argamon (Jan. 2009). “Computational methods in authorship attribution”. In: *Journal of the American Society for Information Science and Technology* 60.1, pp. 9–26. ISSN: 1532-2882. DOI: 10.1002/asi.v60:1 (cit. on p. 6).

Lacoste-Julien, Simon, Fei Sha and Michael I. Jordan (2008). “DiscLDA: discriminative learning for dimensionality reduction and classification”. In: *Advances in Neural Information Processing Systems* 21 (cit. on p. 119).

Lee, Dongwon, Jaewoo Kang, Prasenjit Mitra, C. Lee Giles and Byung-Won On (Dec. 2007). "Are your citations clean?" In: *Communications of the ACM* 50.12, pp. 33–38. ISSN: 0001-0782. DOI: 10.1145/1323688.1323690 (cit. on p. 39).

Li, Jiexun, G. Alan Wang and Hsinchun Chen (2008). "PRM-based identity matching using social context." In: *IEEE International Conference on Intelligence and Security Informatics, ISI 2008*. (Taipei). IEEE, pp. 150–155. URL: http://dblp.uni-trier.de/db/conf/isi/isi2008.html#LiWC08 (cit. on p. 6).

Li, Wei, David Blei and Andrew McCallum (2007). "Nonparametric Bayes pachinko allocation". In: *UAI 2007, Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*. (Vancouver). Ed. by Ronald Parr and Linda C. van der Gaag. AUAI Press (cit. on p. 36).

MacEachern, Steven N. (1999). "Dependent nonparametric processes". In: *Proceedings of the Section on Bayesian Statistical Science* (cit. on pp. 30, 89, 113).

MacEachern, Steven N. and L. Mark Berliner (1994). "Subsampling the Gibbs sampler". English. In: *The American Statistician* 48.3, pages. ISSN: 00031305. URL: http://www.jstor.org/stable/2684714 (cit. on p. 20).

Macherey, Klaus, Andrew Dai, David Talbot, Ashok Popat and Franz Och (2011). "Language-independent compound splitting with morphological operations". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. (Portland). Stroudsburg: Association for Computational Linguistics, pp. 1395–1404. URL: http://www.aclweb.org/anthology/P11-1140 (cit. on p. 10).

MacKay, David J. C. and Linda C. Bauman Peto (1995). "A hierarchical Dirichlet language model". In: *Natural Language Engineering* 1.3, pp. 1–19. ISSN: 13513249. DOI: 10.1017/S1351324900000218 (cit. on p. 42).

Mann, Gideon S. and David Yarowsky (2003). "Unsupervised personal name disambiguation". In: *Proceedings of the seventh conference on Natural language learning at HLTNAACL 2003*. (Edmonton). Ed. by Walter Daelemans and Miles Osborne. Vol. 4. 7-8. Association for Computational Linguistics, pp. 33–40. DOI: 10.3115/1119176.1119181 (cit. on p. 9).

Marx, Zvika, Ido Dagan, Joachim M. Buhmann and Eli Shamir (2002). "Coupled clustering: a method for detecting structural correspondence". In: *Journal of Machine Learning Research* 3.1, pp. 747–780. URL: http://jmlr.csail.mit.edu/papers/volume3/marx02a/marx02a.pdf (cit. on p. 4).

Mayfield, James et al. (2009). “Cross-document coreference resolution: a key technology for learning by reading”. In: *AAAI Spring Symposium: Learning by Reading and Learning to Read*. (Stanford). AAAI, pp. 65–70 (cit. on p. 10).

McCallum, Andrew, Kamal Nigam and Lyle H. Ungar (2000). “Efficient clustering of high-dimensional data sets with application to reference matching”. In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (Boston), pp. 169–178. DOI: 10.1145/347090.347123 (cit. on p. 7).

Metropolis, Nicholas, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller and Edward Teller (1953). “Equation of state calculations by fast computing machines”. In: *The Journal of Chemical Physics* 21.6, pp. 1087–1092. DOI: 10.1063/1.1699114 (cit. on pp. 17, 18).

Mimno, David and Andrew McCallum (2008). “Topic models conditioned on arbitrary features with Dirichlet-multinomial regression”. In: *Uncertainty in Artificial Intelligence*. (Helsinki) (cit. on p. 119).

Minka, Thomas P. (2000). *Estimating a Dirichlet distribution*. Tech. rep. Microsoft. URL: http://research.microsoft.com/$%5Csim$minka/papers/dirichlet/minka-dirichlet.pdf (cit. on p. 53).

Minka, Thomas P. (2003). *A comparison of numerical optimizers for logistic regression*. Tech. rep. (cit. on p. 131).

Mitchell, Tom (Oct. 1997). *Machine Learning*. McGraw-Hill Education (ISE Editions). ISBN: 0071154671 (cit. on p. 5).

Mosteller, Frederick and David L. Wallace (1964). *Inference and disputed authorship: The Federalist*. Addison-Wesley series in behavioral science. Addison-Wesley. URL: http://books.google.com.hk/books?id=KKKFAAAAMAAJ (cit. on p. 5).

Neal, Radford M. (2000). “Markov chain sampling methods for Dirichlet process mixture models”. In: *Journal of Computational and Graphical Statistics* 9.2, pp. 249–265. JSTOR: 1390653 (cit. on pp. 28, 33, 127).

Neal, Radford M (2000). “Slice sampling”. In: *Annals of Statistics* 31.3, p. 40. URL: http://arxiv.org/abs/physics/0009028 (cit. on p. 30).

Newcombe, H. B., J. M. Kennedy, S. J. Axford and A. P. James (1959). “Automatic linkage of vital records”. In: *Science* 130, pp. 954–959 (cit. on p. 4).

Newman, David, Chaitanya Chemudugunta and Padhraic Smyth (2006). “Statistical entity-topic models”. In: *KDD ’06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. (Philadelphia). ACM, pp. 680–686. ISBN: 1-59593-339-5. DOI: 10.1145/1150402.1150487 (cit. on p. 7).

Ng, Vincent (2005). “Machine learning for coreference resolution: from local classification to global ranking”. In: *Proceedings of the 43rd Annual Meeting of the Asssociation for Computational Linguistics (ACL05)*. (Ann Arbor). Association for Computational Linguistics, pp. 157–164. DOI: 10.3115/1219840.1219860 (cit. on p. 8).

Ng, Vincent (2008). “Unsupervised models for coreference resolution”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing EMNLP 08*. (Honolulu). Association for Computational Linguistics, p. 640. DOI: 10.3115/1613715.1613795 (cit. on p. 8).

Ng, Vincent (2010). “Supervised noun phrase coreference research: the first fifteen years”. In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. (Uppsala). Association for Computational Linguistics, pp. 1396–1411. URL: http://portal.acm.org/citation.cfm?id=1858681.1858823 (cit. on p. 9).

Ng, Vincent and Claire Cardie (2002). “Improving machine learning approaches to coreference resolution”. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics ACL 02*. (Philadelphia). Association for Computational Linguistics, p. 104. DOI: 10.3115/1073083.1073102 (cit. on p. 8).

NIST Speech Group (2008). *Automatic Content Extraction 2008 Evaluation Plan (ACE08): Assessment of Detection and Recognition of Entities and Relations Within and Across Documents*. Tech. rep. National Institute of Standards and Technology, pp. 1–16 (cit. on p. 9).

Pang, Bo and Lillian Lee (2005). “Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales”. In: *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*. (Ann Arbor). Ed. by Kevin Knight, Hwee Tou Ng and Kemal Oflazer. Vol. 43. 1. Association for Computational Linguistics, pp. 115–124. DOI: 10.3115/1219840.1219855 (cit. on pp. 132, 146).

Pasula, Hanna, Bhaskara Marthi, Brian Milch, Stuart Russell and Ilya Shpitser (2003). “Identity uncertainty and citation matching”. In: *Advances in Neural Information Processing Systems 15 (NIPS 2002)*. (Vancouver). URL: http://www.cs.berkeley.edu/$%5Csim$milch/papers/nipsnewer.pdf (cit. on p. 6).

Pearson, Karl (1901). “On lines and planes of closest fit to systems of points in space”. In: *Philosophical Magazine Series 6* 2.11, pp. 559–572. DOI: 10.1080/14786440109462720 (cit. on p. 22).

Peng, Fuchun and Andrew McCallum (2006). “Information extraction from research papers using conditional random fields”. In: *Information Processing & Management* 42.4, pp. 963–979. ISSN: 03064573. DOI: 10.1016/j.ipm.2005.09.002 (cit. on p. 102).

Pitman, J. and M. Yor (1997). “The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator”. In: *Annals of Probability* 25.2, pp. 855–900. URL: http://stat-www.berkeley.edu/users/pitman/433.pdf (cit. on p. 30).

Poon, Hoifung and Pedro Domingos (Oct. 2008). “Joint unsupervised coreference resolution with Markov Logic”. In: *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. (Honolulu, Hawaii). Association for Computational Linguistics, pp. 650–659. URL: http://www.aclweb.org/anthology/D08-1068 (cit. on p. 8).

Ramage, Daniel, David Hall, Ramesh Nallapati and Christopher D. Manning (2009). “Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora”. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. (Singapore). Association for Computational Linguistics, pp. 248–256 (cit. on p. 119).

Rao, Delip, Paul McNamee and Mark Dredze (2010). “Streaming cross document entity coreference resolution”. In: *COLING ’10 Proceedings of the 23rd International Conference on Computational Linguistics*. (Beijing). Ed. by Chu-Ren Huang and Dan Jurafsky. Tsinghua University Press, pp. 1050–1058. URL: http://www.aclweb.org/anthology/C10-2121 (cit. on p. 10).

Rasmussen, Carl Edward and Chris K. I. Williams (2006). *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts: MIT Press (cit. on p. 117).

Rodriguez, Abel, David B. Dunson and Alan E. Gelfand (2008). “The nested Dirichlet process”. In: *Journal of the American Statistical Association* 103.483, pp. 1131–1154 (cit. on pp. 78, 79, 82, 83).

Rosen-Zvi, Michal, Thomas Griffiths, Mark Steyvers and Padhraic Smyth (2004). “The author-topic model for authors and documents”. In: *UAI ’04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*. (Banff). AUAI Press, pp. 487–494. ISBN: 0-9749039-0-6 (cit. on pp. 7, 25).

Salton, G., A. Wong and C. S. Yang (Nov. 1975). “A vector space model for automatic indexing”. In: *Communications of the ACM* 18.11, pp. 613–620. ISSN: 0001-0782.

DOI: 10.1145/361219.361220. URL: http://doi.acm.org/10.1145/361219.361220 (cit. on p. 21).

Sethuraman, Jayaram (1994). "A constructive definition of Dirichlet priors". In: *Statistica Sinica* 4, pp. 639–650 (cit. on p. 28).

Shahbaba, Babak and Radford M. Neal (2009). "Nonlinear models using Dirichlet process mixtures". In: *Journal of Machine Learning Research* 10.10(Aug), pp. 1829–1850 (cit. on p. 117).

Singh, Sameer, Amarnag Subramanya, Fernando Pereira and Andrew McCallum (2011). "Large-scale cross-document coreference using distributed inference and hierarchical models". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*. HLT '11. Portland, Oregon: Association for Computational Linguistics, pp. 793–803. ISBN: 978-1-932432-87-9. URL: http://dl.acm.org/citation.cfm?id=2002472.2002573 (cit. on p. 10).

Soon, Wee Meng, Hwee Tou Ng and Daniel Chung Yong Lim (2001). "A machine learning approach to coreference resolution of noun phrases". In: *Computational Linguistics* 27.4, pp. 521–544. ISSN: 08912017. DOI: 10.1162/089120101753342653 (cit. on p. 8).

Stamatatos, Efstathios (2006). "Ensemble-based author identification using character n-grams". In: *In Proceedings of the 3rd International Workshop on Text based Information Retrieval*, pp. 41–46 (cit. on p. 6).

Stamatatos, Efstathios (Mar. 2009). "A survey of modern authorship attribution methods". In: *J. Am. Soc. Inf. Sci. Technol.* 60.3, pp. 538–556. ISSN: 1532-2882. DOI: 10.1002/asi.v60:3. URL: http://dx.doi.org/10.1002/asi.v60:3 (cit. on p. 6).

Steyvers, Mark and Tom Griffiths (2007). "Probabilistic topic models". In: *Latent Semantic Analysis: A Road to Meaning*. Ed. by T Landauer, Mc, S Dennis and W Kintsch (cit. on p. 21).

Steyvers, Mark, Padhraic Smyth, Michal Rosen-Zvi and Thomas Griffiths (2004). "Probabilistic author-topic models for information discovery". In: *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. (Seattle). ACM, pp. 306–315. ISBN: 1-58113-888-1. DOI: 10.1145/1014052.1014087 (cit. on p. 40).

Tan, Yee Fan, Min Yen Kan and Dongwon Lee (2006). "Search engine driven author disambiguation". In: *JCDL '06: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries*. (Chapel Hill). ACM, pp. 314–315. ISBN: 1-59593-354-9. DOI: 10.1145/1141753.1141826 (cit. on p. 5).

Teh, Yee Whye and Michael I. Jordan (2009). “Hierarchical Bayesian nonparametric models with applications”. In: *Bayesian Nonparametrics Principles and Practice*. Cambridge University Press, pp. 1–47. URL: http://eprints.pascal-network.org/archive/00003793/ (cit. on p. 26).

Teh, Yee Whye, Michael I. Jordan, Matthew J. Beal and David M. Blei (2006). “Hierarchical Dirichlet processes”. In: *Journal of the American Statistical Association* 101.476, pp. 1566–1581. DOI: 10.1198/016214506000000302 (cit. on pp. 34, 36, 40, 49, 52, 94).

Teh, Yee Whye, Kenichi Kurihara and Max Welling (2008). “Collapsed variational inference for HDP”. In: *Advances in Neural Information Processing Systems*. Vol. 20 (cit. on p. 113).

Tibshirani, Robert (1994). “Regression shrinkage and selection via the lasso”. In: *Journal of the Royal Statistical Society, Series B* 58, pp. 267–288 (cit. on p. 116).

Torvik, Vetle I., Marc Weeber, Don R. Swanson and Neil R. Smalheiser (2005). “A probabilistic similarity metric for Medline records: a model for author name disambiguation”. In: *Journal of the American Society for Information Science and Technology* 56.2, pp. 140–158. DOI: 10.1002/asi.20105 (cit. on pp. 4, 39).

Wallach, Hanna M. (2006). “Topic modeling: beyond bag-of-words”. In: *ICML ’06: Proceedings of the 23rd international conference on Machine learning*. (Pittsburgh). ACM, pp. 977–984. ISBN: 1-59593-383-2. DOI: 10.1145/1143844.1143967 (cit. on pp. 11, 22, 42, 45, 47).

Wallach, Hanna M., David Mimno and Andrew McCallum (2009). “Rethinking LDA: why priors matter”. In: *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009*. (Vancouver). Ed. by Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams and Aron Culotta. Curran Associates, Inc. (cit. on pp. 23, 36).

Wang, Xuerui, Andrew McCallum and Xing Wei (2007). “Topical n-grams: phrase and topic discovery, with an application to information retrieval”. In: *Proceedings of the 7th IEEE International Conference on Data Mining*. (Omaha), pp. 697–702 (cit. on p. 11).

Wellner, Ben, Andrew McCallum, Fuchun Peng and Michael Hay (2004). “An integrated conditional model of information extraction and coreference with application to citation matching”. In: *Uncertainty in Artificial Intelligence 2004*. (Banff). AUAI Press. URL: http://www.cs.umass.edu/$%5Csim$mccallum/papers/integrated04uai.pdf (cit. on p. 6).

Wick, Michael, Aron Culotta, Khashayar Rohanimanesh and Andrew McCallum (2009). “An entity based model for coreference resolution”. In: *SIAM International Conference on Data Mining*. (Sparks). Vol. 9, pp. 365–376. URL: http://www.siam.org/proceedings/datamining/2009/dm09_036_wickm.pdf (cit. on p. 8).

Wick, Michael, Sameer Singh and Andrew McCallum (2012). “A discriminative hierarchical model for fast coreference at large scale”. In: *Association for Computational Linguistics (ACL)* (cit. on p. 10).

Winkler, William E. (2006). *Overview of record linkage and current research directions*. Tech. rep. 2006-2. Statistical Research Division, U.S. Census Bureau, pp. 1–28 (cit. on p. 4).

Winkler, William E. and Yves Thibaudeau (1991). *An Application of the Fellegi-Sunter Model of Record Linkage to The 1990 U.S. Census*. Tech. rep. RR91/09. U.S. Bureau of Census (cit. on p. 4).

Xing, Eric P., Kyung-Ah Sohn, Michael I. Jordan and Yee-Whye Teh (2006). “Bayesian multi-population haplotype inference via a hierarchical Dirichlet process mixture”. In: *ICML ’06: Proceedings of the 23rd international conference on Machine learning*. (Pittsburgh). ACM, pp. 1049–1056. ISBN: 1-59593-383-2 (cit. on p. 22).

Zhu, Jun, Amr Ahmed and Eric P. Xing (2009). “MedLDA: maximum margin supervised topic models for regression and classification”. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. (Montreal). ACM, pp. 1257–1264 (cit. on p. 119).