Australian Government
Department of Defence
Defence Science and
Technology Organisation

Numerical Algorithms for the Analysis of Expert Opinions Elicited in Text Format

W. P. Malcolm [1] ***and Wray Buntine*** [2]

[1] **Joint Operations Division, DSTO**

[2] **National ICT Australia**

Defence Science and Technology Organisation

DSTO–TR–2797

ABSTRACT

Latent Dirichlet Allocation (LDA) is a scheme which may be used to estimate topics and their probabilities within a corpus of text data. The fundamental assumptions in this scheme are that text is a realisation of a stochastic generative model and that this model is well described by the combination of multinomial probability distributions and Dirichlet probability distributions. Various means can be used to solve the Bayesian estimation task arising in LDA. Our formulations of LDA are applied to subject matter expert text data elicited through carefully constructed decision support workshops. In the main these workshops address substantial problems in Australian Defence Capability. The application of LDA here is motivated by a need to provide insights into the collected text, which is often voluminous and complex in form. Additional investigations described in this report concern questions of identifying and quantifying differences between stake-holder group text written to a common subject matter. Sentiment scores and key-phase estimators are used to indicate stake-holder differences. Some examples are provided using unclassified data.

APPROVED FOR PUBLIC RELEASE

Published by

DSTO Defence Science and Technology Organisation
Fairbairn Business Park,
Department of Defence, Canberra, ACT 2600, Australia

Telephone: (02) 6128 6371
Facsimile: (02) 6128 6480

© *Commonwealth of Australia 2013*
AR No. 015-501
April, 2013

APPROVED FOR PUBLIC RELEASE

Numerical Algorithms for the Analysis of Expert Opinions Elicited in Text Format

Executive Summary

This report describes the motivation, scope and outcomes of a recent Defence Science and Technology Organisation (DSTO) research collaboration with Industry, intended to develop a specialised computer-based text analysis capability. In March of 2011, a formal research agreement was struck between the Joint Operations Divsion (JOD) and the National ICT Australia (NICTA). Fundamentally this agreement was aimed at developing a specific text analysis capability, with particular emphasis placed upon examining collections of text-format expert opinions, each of which concerned a given defence capability issue. Here the term *text analysis* might include: identifying a finite number of key topics in a text corpus and their relative weightings, or, some quantitative measure of difference between stake-holder group opinions on specific common issue etc.

The primary motivation for this work is derived from text-data volume & processing issues arising in the Joint Decision Support Centre (JDSC). The JDSC was established in March of 2006 and is one component of a unique collaboration between DSTO and the Capability Development Group (CDG). Part of the JDSC's core program of work concerns providing decision support to current projects listed in the Defence Capability Plan (DCP). The common vehicle for this support is a facilitated defence capability workshop. Such workshops typically run for 2-4 days and may include up to 40 attendees, consisting of technical SMEs, Australian Defence Force (ADF) staff and representatives from various stake-holder groups. These workshops are carefully designed to address specific defence questions and to elicit, record and analyse expert opinions. Note, it is important to understand that the JDSC's scope here best 'approximates' what is known as the *Expert Problem* as its described in the Taxonomy due to French [Fre85]. Briefly, the Expert Problem is defined as follows:

Definition 0.1 (French, 1985). *A group of experts are asked for advice by a Decision Maker (DM) who faces a specific real decision problem. The DM is, or can be taken to be, outside the group. The DM takes responsibility and accountability for the consequences of the decision. The experts are free from such responsibility and accountability. In this context the emphasis is on the DM learning from the experts.*

In our context the relevance of French's definition is primarily expressed in the last sentence of his definition, emphasising the DM learning from experts. Consequently, JDSC decision support workshops are orientated towards *informing* Defence Decision Makers though workshops and their outcomes. JDSC workshop data are generally of two classes: 1) numeric, such as voting scores or quantitative preference rankings, or 2) text data collected through network-based text collection software. The text data collected at JDSC workshops is usually rich in content, but significant in volume. Ideally, this data should be analysed both *in situ*, that is during a given workshop, and off-line post-workshop. The main tasks here are data reduction and visualisation, that is, to compute an accessible summary visualisation of valuable information inherent in a corpus of text likely to inform Defence Decision Makers. It should also be noted here that while the motivation for this project originated from the inherent needs of JDSC workshops, the outcomes of this project are not limited to JDSC related activities.

The main outcomes detailed in this report concern the development and capabilities of a set of text analysis algorithms intended to support and enhance the various tasks described above. Specific capabilities detailed here are:

- **Probabilistic Topic Analysis:** Topic analysis concerns identifying a finite number of topics within a corpus of text and subsequently estimating a level of association of document elements (such as words or phrases) to *each* of these topics.
- **Differential Analysis:** Differential analysis concerns identifying and quantifying the differences between subsets of text, where set membership is by affiliation to a specific stake-holder group. For example, what might be the differences between text data generated by ARMY SMEs and Air Force SMEs on a common defence capability issue? Further, how might such differences be computed and analysed?
- **Key-Phrase Analysis:** Key phrase analysis concerns identifying and ranking the top N phrases in a document, either for the complete document or subsets of text attributed to various stake-holders.

This report also contains technical detail on mathematical foundations of the work and specific details on some algorithmic issues inherent in its complex estimation tasks. Finally, an example of the algorithms at work on an unclassified text data set is provided. This text data was collected at a special JDSC workshop including two groups only, DSTO staff and NICTA staff. Primarily this unclassified data is included to demonstrate graphic visualisations of the three aforementioned core tasks.

Authors

W. P. Malcolm
Joint Operations Division

W. P. Malcolm's tertiary education is in Applied Physics and Applied Mathematics. His PhD from the ANU was awarded in 1999 and concerns topics in hidden-state estimation for stochastic dynamics with counting process observations. He joined Joint Operations Division in January 2009, directly after completing a year as the J6 in the Counter IED Task Force. Prior to joining the CIED Task Force he worked as a Senior Researcher at the National ICT Australia. From July 2003 until July 2004 he was a Postdoctoral Researcher at the Haskayne School of Business at the University of Calgary in Alberta Canada. His research interests concern: parameter estimation, filtering & smoothing, mathematical techniques for the aggregation and analysis of expert opinion, decision analysis, text analysis and modelling with point and marked point processes.

Wray Buntine
National ICT Australia & The Australian National University

Dr. Wray Buntine joined NICTA in Canberra Australia in April 2007. He was previously of University of Helsinki and Helsinki Institute for Information Technology from 2002, where he conceived and coordinated the European Union 6th Framework project ALVIS, for semantic search engines. He was previously at NASA Ames Research Centre, University of California, Berkeley, and Google, and he is known for his theoretical and applied work in document and text analysis, data mining and machine learning, and probabilistic methods. He is currently Principal Researcher working on applying probabilistic and non-parametric methods to tasks such as text analysis. In 2009 he was programme co-chair of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, in Bled, Slovenia, organised the PASCAL2 Summer School on Machine Learning in Singapore in 2011, and was programme co-chair of Asian Conference on Machine Learning in Singapore in 2012. He reviews for conferences such as ECIR, SIGIR, ECML-PKDD, WWW, ICML, UAI, and KDD, and is on the editorial board of Data Mining and Knowledge Discovery.

THIS PAGE IS INTENTIONALLY BLANK

Contents

1 Introduction **1**

1.1 Aims of this Report 2

1.2 Organisation of this Report 3

2 The JDSC **3**

2.1 Background 3

2.2 Function 3

3 Eliciting Expert Opinions via Facilitated Workshop **4**

3.1 Computer-based Text Collection for Sets of Experts 5

4 The Elicitation of an Unclassified Text Data Set **5**

4.1 Background 5

4.2 Specific Elicitation Questions 6

5 Some Basic Elements of Text Analysis **8**

5.1 Representations for Text Data 8

5.1.1 Deterministic Models for Text Data 8

5.1.2 Stochastic Models for Text Data 9

5.2 Some Basic Definitions in NLP 10

5.3 The Notion of Exchangeable Random Variables 13

5.4 Specific Natural Language Processing Routines 13

5.4.1 Acronym Detection Scheme 14

5.4.2 Named Entity Recognition (NER) Scheme 14

5.4.3 Named Entity (NE) Extractor 15

5.4.4 Named Entity (NE) Classifier 15

6 Probabilistic Topic Modelling **15**

6.1 Probabilistic Topic Models 16

6.2 Latent Dirichlet Allocation 18

6.3 Numerical Implementation 22

6.4 MCMC-based LDA and the Uniqueness of Outputs 26

6.5 Visualisation for Topic Estimation 27

7 Differential Analysis of Stake-Holder Text **27**

7.1 Quantifying Sentiment 28

7.2 Visualisation for Sentiment Allocations 30

8 Key-Phrase Identification **32**

8.1 Definitions and Basic Theory 32

8.2 Visualisation For Key Phrases 33

9 Future Work **35**

9.1 Kullback-Leibler Inter-Topic Distances 35

9.2 Differential Analyses for Text Subsets 35

9.3 Algorithm Property Analyses 35

9.4 Statistical Significance Filtering 36

9.5 Topic Modelling Visualisation Schemes 36

9.6 The Analysis of Historical Data 36

10 Conclusion **37**

10.1 Overview 37

10.2 Summary of Contributions 37

11 Acknowledgments **38**

References **38**

Appendices

A Conjugate Priors **47**

A.1 Definitions 47

A.2 Example 47

B Multinomial Probability Distributions **49**

B.1 Background 49

B.2 Derivation 49

B.3 Some Statistics 51

C The Beta Distribution **52**

C.1 Basic Properties 52

C.2 Checking that $\int f(\xi)d\xi = 1$ 53

D The Dirichlet Probability Distribution **55**

D.1 Basic Statistics For A Dirichlet Distribution 56

D.2 Conjugacy with a Multinomial Distribution 57

D.3 Generating Dirichlet Random Variables 58

D.4 Aggregation Property 61

E Gibbs Sampling **63**

E.1 Background 63

E.2 Basic Markov Chain Monte Carlo (MCMC) 63

E.3 Example 65

F Sample Elicited Text Data **67**

Figures

1 Graphic depiction of the JDSC Facility. Typically each participant seated in this facility will have an individual computer terminal for text entry. In this facility visualisation is extensively used to create context and provide information & immersion for the workshop participants. 4

2 A simple example of two documents and a query in vector space form. 9

3 A graphical model representation of Latent Dirichlet Allocation. Here the only observed data are the words, denoted by the shaded circle containing W 19

4 This figure shows a real-data example for 9 topics projected onto 2D. The clusters of words at the topic centers list the top (probabilistically) 5 words associated to a given topic. The details on these plots is given in section 6.5. 28

5 This figure shows a two stake-holder differential analysis at a word level only. 31

6 This figure shows a two stake-holder differential analysis indicating: an estimated collection of key-phrases, the frequency of usage of key phrases and sentiment attached to the usage of these phrases 34

D1 Simulated example of the shape-parameter additive property for gamma distributed random variables. 62

E1 Gibbs Sampler example for the bivariate density given at (E7) 66

Tables

1 Workshop Session 1 (Decision Making) 6

2 Workshop Session 2 (Research and Development) 7

3 Workshop Session 3 (Text Analysis) 7

4 A simple stemming example with German verbs 10

5 The eight basic Parts of Speech (POS) 12

6 Typical(idealised) word-topic allocation probability vectors concerning some areas of defence capability. Note that in this example FIC appears in two different topics, here FSR and Submarine. Note also that the number of most-significant words per topic may vary. The vertical ellipses indicate that the shown words are proper subsets of a greater vocabulary which we denote by V. 17

7 Some regular expression symbols 33

A1 Posterior distribution parameter updates for a Beta prior and binomial likelihood. 48

D1 Some basic statistics for a Dirichlet distribution 57

1 Introduction

Modern algorithmic text analysis includes vast areas of research and application. Of course much of the momentum in these areas has arisen from the enormous changes over the last 20 years or so in the way (text) information is collected, digitised, stored and made available through media such as the Internet. The many current domains of research and application in text analysis are far too numerous to mention in this report (the interested reader might consider [Ber04],[FNR03]). Instead we restrict our attention to a specific defence task in text analysis, that is, supporting Joint Decision Support Centre workshops by providing a means to graphically depict and summarise certain information contained in a corpus[1] of specially elicited text. What we would like to do is examine a corpus of text collected through the JDSC workshop model. In particular we would like an estimated topic map for a given corpus of text, showing subsets of words associated to a given topic and the estimated probability of these associations. Further analysis detailed in this report will consider differential text analysis on a collection of text and the identification and rankings of key phrases. The need for a differential analysis capability arises in part from the *raison d'être* motivating the JDSC, which was to ensure that defence capability questions addressed through the JDSC are contributed to by all stake-holder groups, such as Army, Navy, Air Force etc. Each of these attending stake-holder groups will submit text on a common point of study, for example, the speculated operational value of a certain class of Helicopter *etc.* A natural task here is to examine differences (or similarities) in the text submitted by the different stake-holder groups. We would also like to have a means by which we could identify and rank the main phrases in a corpus of text. The text analysis capability described above is intended to solve two text analysis problems in the JDSC, 1) *the on-line problem* and 2) *the off-line problem.* The on-line problem concerns developing an *in situ* capability which would provide the workshop facilitation and workshop team visibility of the elicited text through summarising graphic depictions of that text. The intention is to provide that capability in real time as the workshops evolve. The off-line problem refers to analysing the elicited text after a workshop is completed. Typically 4-day workshops with 20-40 SMEs can generate a large amount of text. The JDSC analyst needs an effective means of analysing and reporting on such text. Currently there are many software products on the market that address various areas of text mining and text analysis, most with emphasis oriented towards the analysis of news media, or applications in business and marketing domains. However, few of these products directly address stake-holder workshop elicited text data in a defence context. This situation, in part, motivated a text analysis research program with an external partner.

The DSTO/JDSC research agreement with the National ICT Australia to develop a text analysis capability was signed by both parties in March of 2011, with an estimated project duration of approximately 12 months. The deliverable in this agreement included a software capability to analyse workshop text data in three respects: topic estimation, differential analysis and key phrase estimation and ranking. NICTA is a federally funded Australia-wide research organisation founded in 2002 and has six core research groups, these are:

[1] Here the term *corpus of text* refers to a relatively large structured collection/set of text

1. Computer Vision,
2. Control and Signal Processing,
3. Machine Learning,
4. Networks,
5. Optimisation,
6. Software Systems.

Text analysis research in NICTA is carried out predominantly by the Machine Learning area but is not limited to any one specific NICTA research laboratory. The research project detailed in this report engaged machine learning researchers based at the NICTA Laboratory in Canberra led by Prof. Wray Buntine. There are many diverse means by which one can analyse text data, for example, stochastic and deterministic methods can be applied. The approach taken in this project is stochastic and inherently Bayesian and based upon the notion of a stochastic generative[2] model. The specific modelling paradigm used here for topic estimation is Latent Dirichlet Allocation (LDA).

1.1 Aims of this Report

This report essentially describes the outcomes of a recently completed DSTO/NICTA research contract to develop a computer-based text analysis capability. Its main aims were:

- to describe the eliciting of SME text data through the vehicle of JDSC workshops,
- to briefly recall some basic elements/notions in text analysis,
- to describe the technical details of the estimation schemes used, including LDA and its numerical implementation through a variant of Gibbs Sampling
- to explain the visualisations used for: topic estimation, differential analysis and key phrase estimation,
- to provide examples of the text analysis capability developed through an unclassified (real) text data-set collected at a JDSC workshop and
- to provide expository appendices on some fundamental components of this work such as Dirichlet probability distributions and Gibbs sampling.

[2]This class of model will be explained later.

1.2 Organisation of this Report

This report is organised as follows. In Sections 2 and 3, we briefly describe the main functions of the JDSC and how text is elicited at the JDSC through facilitated workshops. These sections provides background and context only. The collection of an unclassified text data set is described in §4.

In §5 we recall some basic notions in text analysis such as *bag of words* representations and text pre-processing tasks such as the removal of stop-words etc. We also provide some brief details on relevant elements of Natural Language Processing.

The main core of technical work in this report begins in §6 detailing the specific theory applied for probabilistic topic modelling. This development of core technical work continues through to Sections 7 and 8, covering differential analysis and key phrase analysis respectively. In §9 we list possible extensions of the work described in this report. Finally, a collection of appendices are provided for completeness, covering certain (less known) probability distributions and some basic on Gibbs sampling. In particular the Dirichlet distribution is discussed in Appendix D. While this interesting distribution is not in general widely known, it plays a fundamental role in the topic estimation results in this report.

2 The JDSC

2.1 Background

The JDSC was established in 2006 and was motivated by the need to support strategic decision making processes concerning Defence Capability. The JDSC supports decision processes by facilitating, enhancing and contributing to the convergence of decision making. The primary aim of the JDSC is to provide objective, impartial, clear and timely decision support to future defence capability decisions, including current projects listed in the Defence Capability Plan. The DCP is defined as follows,

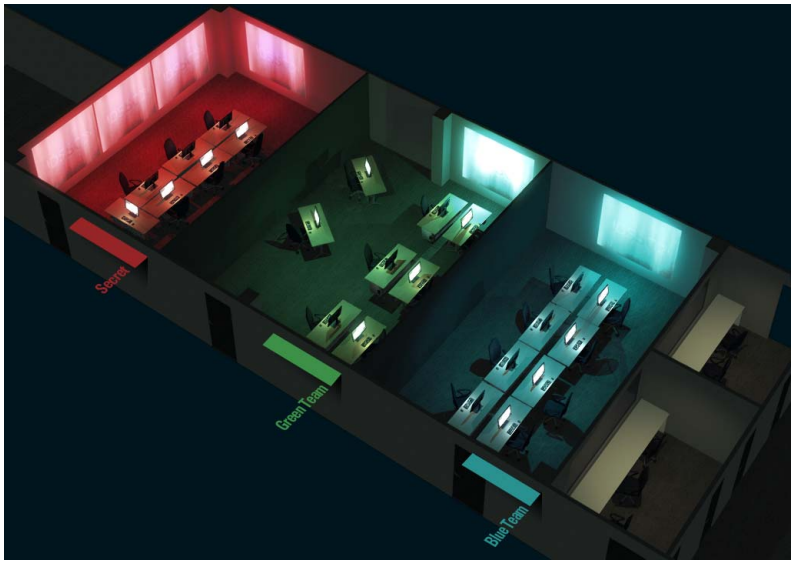
Definition 2.1 (DCP). *The DCP outlines the Governments long term Defence capability plans. It is a detailed, costed 10 year plan comprising the unapproved major capital equipment projects that aim to ensure that Defence has a balanced force that is able to achieve the capability goals identified in the (current) Defence White Paper and subsequent strategic updates.*

2.2 Function

The DCP is a living document listing defence capability projects, for example SEA1000 concerns Australia's Future Submarine, JP2060 Army's Deployed Health. The majority of DCP projects are phased over time and are progressively reviewed by the appropriate committees. Further, such committees may require detailed analysis on key issues concerning a given capability. The JDSC may be tasked to provide such analysis either through the model of a facilitated workshop, or by other means. Capability questions examined through

JDSC workshops may address, for example, the updating/modification/rationalisation of a Defence Capability project, or, the introduction of a new Defence Capability. In most of these cases the outputs of JDSC decision support includes a written report. This support is substantiated, where feasible, by broad engagement of stake-holders, the inclusion of subject matter experts, technical reach-back into DSTO (and through it to the wider scientific community), in-house modelling and simulation and scientific rigor. A graphic

Figure 1: *Graphic depiction of the JDSC Facility. Typically each participant seated in this facility will have an individual computer terminal for text entry. In this facility visualisation is extensively used to create context and provide information & immersion for the workshop participants.*



depiction of the JDSC facility is shown in Figure 1. Typically JDSC workshops run for 2-4 full days and might include 20-40 specialised SMEs.

3 Eliciting Expert Opinions via Facilitated Workshop

The techniques for elicitation and facilitation of SME workshops addressing the *Expert Problem* are diverse and many. This area is and indeed well beyond the scope of this report, however, some comments are in order to provide context. The interested reader might refer to the recent article [PAF+07], or a relatively recent special issue of the Journal of Operations Research Society 2007, Number 58. These papers consider what is generally referred to as Problem Structuring Methods (PSMs). PSMs are fundamental to the facilitated group settings of the JDSC where pre-workshop tasks concern detailed examinations of the defence capability issue to be studied. Ultimately a schedule must be generated for each workshop, including a program of questions, surveys, debate topics voting schemes *etc*, all intended to elicit expert information for the benefit of the remote DM. This task is far more complex than it might first seem. Further, compounding this task is sheer &

often elusive complexity of the problems studied at the JDSC. The unfortunate convention used to name this class of problem is to refer to them as "Wicked Problems" (see [Pid96b][Pid96a]). Such problems are characterised in [PAF+07] as follows:

- They are one-off problems that may have some similarities with previous problems but have never been encountered before.
- Solving Wicked Problems may cause or worsen other interconnected problems.
- There are usually many stake-holders, often holding conflicting values and perspectives in the decision context.
- There is no right or wrong solution; there may be "solutions" that are perceived to be good by some, but seldom all stake-holders.

Suffice to say, the task of eliciting expert opinions in JDSC workshops is generally difficult. However, one can (loosely speaking) categorise typical SME responses. These responses might be a numeric vote of preference, a binary reply accept/reject or a text input response either offering an opinion or a text-form answer to as specific problem. It is precisely the analysis of these text responses that we are concerned with in this report.

3.1 Computer-based Text Collection for Sets of Experts

The JDSC uses collaborative text entry software, with which participants (i.e. experts) can enter their opinions on certain topics. The means of text entry for JDSC workshop (co-located) participants is via a single per/person computer terminal. In most cases each attendee/participant will have their own computer terminal. Each entry is date and time stamped and contains a user number and so can be tagged to a particular participant. Text responses can be a sentence, a paragraph or indeed several paragraphs. An example of this data is given in Appendix F.

4 The Elicitation of an Unclassified Text Data Set

4.1 Background

Given the JDSC is an Australian defence facility, much of its work, including outputs and collected data, is naturally classified and so is not generally available to external research partners. The work in this Technical Report is based upon a research collaboration with NICTA, which is separate to the Department of Defence. This motivated a clear need to develop an unclassified text data set.

On the 27th March 2011, a specially prepared text collection workshop was held at the JDSC Canberra. This workshop involved just two stake-holder groups, JOD/DSTO staff and NICTA research staff. The intention of this event was to generate an unclassified text-data set, elicited through a typical JDSC workshop and to expose the NICTA researchers

to an example of precisely how JDSC workshop text is elicited in an unclassified setting. While the duration of this workshop was only half a working day, it did serve the purpose of generating a useful and accessible text-data.

Prior to this workshop the JDSC Discovery Team[3] contributed to the planning and scheduling of the workshop. Here the foremost task was to decide upon an unclassified subject matter suitable for the workshop, that is, a subject matter rich enough to engender healthy engagement and produce a text corpus containing some diversity, such as different opinions, polarity, sentiment and bias. The workshop ran over the course of roughly one afternoon and engaged approximately 15 participants. Consequently this is statistically typical of JDSC workshops which could run over 4 full days and involve up to 40 or more participants.

4.2 Specific Elicitation Questions

The three tables below list the session-wise questions which were submitted to the participants. In general, the unclassified subject matter from which these questions were derived concerned; decision making, research and automated text analysis. These questions were designed to (hopefully) engender vigorous discussion and illuminate sentiment and potential differences between the two participating groups.

Table 1: *Workshop Session 1 (Decision Making)*

Facilitator Question	Org. Diff.	Sentiment	Polarised	Acronyms
Q1: *What are the difference between complex decision and simple decision making?*	✓	✗	✗	✗
Q2: *Computer-based analytical tools are less effective for decision support than a human analyst?*	✓	✓	✗	✓
Q3: *Users of decision support tools need to know the details (theoretical basis) to achieve effective use?*	✓	✓	✗	✓

Remark 4.1. *The unclassified data set generated from this JDSC activity is relatively short. Consequently its not statistically significant. The purpose of the data is largely to depict software outputs based upon real data, that is text data in a real JDSC workshop. In Appendix F we show the data collected from this activity corresponding to Table 2.*

To provide a brief example here, we show just two responses to Question 4 which were elicited in Session 2. This question was, *How is the value of research best measured ?* Two

[3]The JDSC Discovery Team was raised 2010. Its primary tasks concern pre workshop engagement to identify and shape the specific fundamental defence capability questions being investigated by clients and subsequently addressed in decision support workshops. Typical outcomes of such preparation work might be: a workshop plan/strategy, a workshop schedule and a specific sequence of questions for the workshop facilitator to elicit expert opinions

Table 2: *Workshop Session 2 (Research and Development)*

Facilitator Question	Org. Diff.	Sentiment	Polarised	Acronyms
Q4: *How is the value of research best measured?*	✓	✓	✓	✗
Q5: *Client-based research produces limited outcomes?*	✓	✓	✗	✗
Q6: *Industry-based Research & Development produces more useful outcomes than Academia*	✓	✓	✗	✗

Table 3: *Workshop Session 3 (Text Analysis)*

Facilitator Question	Org. Diff.	Sentiment	Polarised	Acronyms
Q7: *Human analysis of text to create structure is more powerful than computer-based text analysis?*	✓	✓	✓	✗
Q8: *Unstructured data does not help a decision making process, quantitative fact-based data is required*	✓	✓	✗	✗
Q9: *Other industries outside of defence might benefit from automated text analysis*	✗	✗	✗	✗

typical responses showing participant number, date stamps and time stamps are shown below.

- ◇ **1.1** Publications and Journal Rankings
  `Submitted by 19 (2011-03-24 22:14:55)`
- ◇ **1.2** The number of citations both primary and secondary citations
  `Submitted by 14 (2011-03-24 22:15:15)`

5 Some Basic Elements of Text Analysis

In this section we recall some basic elements in modern text analysis and in particular text analysis definitions and routines used in the work reported here. This material includes some basic results in Natural Language Processing (NLP) and a brief statement of the important theorem of Bruno de Finetti.

5.1 Representations for Text Data

It is clear that text-data is an abstract data-type and in general a *difficult* class of data type with significant diversity and complexity. To proceed with any form of analysis on text data one must first decide upon a suitable representation for text such that the text in question can, for example, take values in a space and thereafter be examined through pattern recognition, applying metrics, or statistical analysis *etc.*

There are numerous representations available for text, some deterministic and some stochastic. To fix some basic ideas we start with a deterministic representation and make use of geometric ideas in Euclidean space and linear algebra.

5.1.1 Deterministic Models for Text Data

The most common deterministic model for text is the vector space model due to G. Salmond[4] (see [SM86], [SWY75]). The Vector space model and it's numerous variants, essentially characterise a given document by a point in Euclidean space, that is, document i, written $D^i \in \mathbb{R}^p$, is defined by the vector

$$D^i \triangleq \left(m_1^i, m_2^i, \ldots, m_p^i\right)'. \tag{1}$$

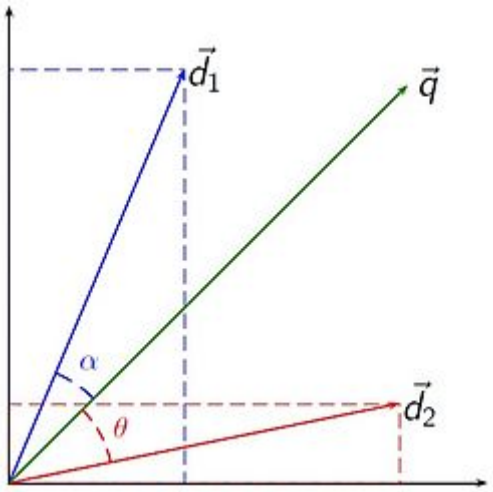
Here the components m_j^i might indicate weights for importance of terms within the document and typically would be computed by combination of a document-importance weight and corpus-importance weight. The convention is to consider a local (document level) weight and a global (corpus level) weight, that is, for some term (typically a word) labelled j, we compute

$$m_j \triangleq \ell_j^i \times g_j. \tag{2}$$

[4] As an aside, there is a curious history with this work and its appearance in the published literature, see the text [Dur04].

There are numerous schemes in the literature to compute ℓ and g, see the texts [FBY92], [Kow97]. The primary application of the vector space model has been in Information Retrieval (IR). Put simply, an IR system matches user queries (formal statements and information needs) to documents stored in a database. Figure 2 shows a simple example of two documents in a Vector space and a "query" in its corresponding vector form. With this model one might define distance (from the document in question) to the query vector through the angle, or through the euclidean norm. As a text analysis example one might convert all document vectors to unit vectors and consider the pattern of points resulting on a unity-radius hypersphere. Presumably a variety of algorithms might then be applied to classify subsets of documents/points in a corpus or to perform cluster analysis (see [Bis06] and [Bra89]). If a corpus of documents is mapped to a vector space through the

Figure 2: *A simple example of two documents and a query in vector space form.*



representation at (1), then one might collect these vectors as columns of a matrix. This is the usual starting point for matrix-based methods of text analysis such as Latent Semantic Indexing (LSI), see [Ber04].

Remark 5.1. *There are known limitations with vector space models which are well documented in the literature. One concerns semantic sensitivity, that is, documents with similar context/meaning but which use different vocabularies will not be associated resulting in a failure to conclude a match.*

5.1.2 Stochastic Models for Text Data

Stochastic models for text are perhaps better known than deterministic models. Indeed the origin of the stochastic processes known as Markov chains, (due to Andrei Andreevich Markov, see [HS01]) was Markov's investigation into a two-state chain (vowels and consonants) derived from Pushkin's poetry. Moreover, its easy to imagine how such an investigation might be extended beyond characters to parts of speech, such as prepositions, nouns and adjectives etc. If one considered English text as a sequence of such parts, then a dependent stochastic process such as a Markov chain might easily be modelled. For example, the probability of a transition from a preposition to a preposition would be low,

however the probability of a transition from an adjective to a noun would be high. An interesting example of Markov chains applied to text is given in the article [IG01]. The work subsequently described in this report is entirely based upon a stochastic model which is a special type of stochastic generative model.

Remark 5.2. *The essential point to note on stochastic models for text analysis is that a corpus of documents, (or indeed a single document), is taken as a realisation of some class of stochastic model. This is a critical assumption and is pivotal to understanding probabilistic topic modelling.*

5.2 Some Basic Definitions in NLP

There are many excellent texts dealing with NLP, for example see [MS99] and [Mit04]. This is a vast subject and well beyond the scope of this report. The inclusions below have been chosen to give some indication about the pre-processing of our text before it is analysed for topics etc.

Definition 5.1 (Corpus). *A corpus is a structured collection of language texts that is intended to be a rational sample of the language in question (see the interesting reference [JT06]). A very well known corpus is the Brown corpus of the English Language which was made available in 1960. The plural of corpus is corpora.*

Definition 5.2 (Stop Words). *A "stop word" is a word that is functional, rather than carrying information or meaning. These words are sometimes called function words. Some examples are prepositions and conjunctions. An example subset of typical stop words in English might be {also, it, go, to, she, they}. There are many well known lists of stop words in English available electronically on the Internet, see for example* `http://www.ranks.nl/resources/stopwords.html`.

Definition 5.3 (Stemming). *Loosely speaking, stemming concerns identifying a canonical or irreducible component of a word common to all inflections of a word. For example, consider the conjugation of German verbs and in particular those verbs known as weak verbs. These verbs have an irreducible component called the stem vowel. For example, the German verb* arbeiten *(to work), is conjugated in the following way in Table 4, In*

Table 4: *A simple stemming example with German verbs*

ich	arbeite,
du	arbeitest,
wir	arbeiten.

this example the stem vowel is arbeit. A similar example in English might be; computer, computing, computed, computation. Many algorithms for stemming are readily available on he Internet, for example see `http://tartarus.org/~martin/PorterStemmer/` *and* `http://www.comp.lancs.ac.uk/computing/research/stemming/`.

Definition 5.4 (Lemmatisation). *Lemmatisation is similar (roughly) to Stemming and refers to an algorithmic process to identify the so-called lemma of a given word. In general*

this task is somewhat more sophisticated than stemming as it may require an analysis of context for a specific word and also a tagging into a group, such as a noun or a verb by using a Parts of Speech (POS) algorithm. For example a verb such as running would have the suffix ing removed whereas, a noun in plural form such as houses, would be reduced to house.

Definition 5.5 (Named Entities & Recognition). *Named Entity recognition involves the task of identifying proper names as they relate to a set of predefined categories of interest. This is in general not a trivial task in NLP and encounters some immediate problems, for example the word June, does it refer to a month or a person called June ? The word Washington might be George Washington and it might be a state of the USA. Sub might refer to Submarine, boat might also refer to submarine.*

Definition 5.6 (Tokens and Tokenisation). *The word Token has a variety of meanings, in our context it means a convenient symbol to represent something of interest, usually words. The task of Tokenisation is to represent text as a collection of tokens. Tokens may be words or grammatical symbols. In most cases tokenisation relies upon locating word boundaries such as the beginning or ending of a word. Tokenisation can sometimes be referred to as word segmentation.*

Definition 5.7 (Common Frequency/Weighting Measures). *In basic text analysis and indeed areas of cryptography, basic frequency indicators are often used for inference and analysis, for example word or character counts and their distributions can be informative. In text analysis the two most common frequency counts used are the so-called term-frequency (TF) and term-inverse-document-frequency counts (TF-IDF). Term Frequency is just a straight count of the number of occurrences of a given term in a given document. For example, suppose the term in question is denoted by t and the document by D^i, then this quantity is written as $TF_{t,D^i} \in \mathbb{N}$. Note this quantity is not a corpus level quantity. In contrast, the TF-IDF includes a measure of frequency at a corpus level, that is, it extends TF to include occurrence counts in all documents in a corpus. To label a single corpus of I documents, we write $\mathcal{D} \triangleq \{D^1, D^2, \ldots, D^I\}$. The Inverse Document Frequency (IDF) may be written,*

$$IDF_{t,\mathcal{D}} \triangleq \log\left\{\frac{|\mathcal{D}|}{1+|\{D \in \mathcal{D} \mid t \in D\}|}\right\} \tag{3}$$

Here $|\cdot|$ denotes cardinality and the addition of unity on the denominator to avoid divide by zero errors in cases where a term does not appear. The most widely used weight, (ie the vector components m_j in equation 2), for a candidate term t_j in given document D^i is

$$m_j^i \triangleq TF_{t_j,D^i} \times IDF_{t_j,\mathcal{D}}. \tag{4}$$

The NLP community uses a variety of frequency measures such as the one at (4), choice usually depends upon context and perhaps specific data issues. One can see from equation (3) that its task is to attenuate the weight of common words used often. For example "the". This word is likely to appear in every document. This means (assuming we disregard the unity offset in the denominator of (3)) that the quotient is unity and hence its log is zero, given the term zero weight, as would be expected for a common word.

Definition 5.8 (Parts of Speech (POS) Tagging). *POS tagging refers to the task of classifying individual words into parts of speech. In English, (and in German), there are eight fundamental parts of speech, these are shown in Table 5 below. One could further*

Table 5: *The eight basic Parts of Speech (POS)*

noun	pronoun	verb	adjective	article	adverb	preposition	conjunction

refine this set by classifying adjectives as descriptive, demonstrative, possessive or interrogative, or nouns that might be plural or singular, see [MS99] page 342. The task of mapping given words to the types in the table above is not trivial. This task is discussed at length in the literature and there are numerous algorithms available. Indeed the POS tagging problem can be cast as a Hidden Markov Model (HMM) problem and solved with estimation schemes such as the Viterbi Algorithm. A good treatment of this approach is given in [MS99].

Definition 5.9 (Bag of Words). *The notion of the so-called* Bag of Words *model for text is central to all the text analysis methods described in this report. In this model (roughly) one considers text as invariant to word order and invariant to grammar. For example, §8.4 from the 2009 Australian Defence White Paper*[5] *reads as follows:*

> *In the case of the submarine force, the Government takes the view that our future strategic circumstances necessitate a substantially expanded submarine fleet of 12 boats in order to sustain a force at sea large enough in a crisis or conflict to be able to defend our approaches (including at considerable distance from Australia, if necessary), protect and support other ADF assets, and undertake certain strategic missions where the stealth and other operating characteristics of highly-capable advanced submarines would be crucial. Moreover, a larger submarine force would significantly increase the military planning challenges faced by any adversaries, and increase the size and capabilities of the force they would have to be prepared to commit to attack us directly, or coerce, intimidate or otherwise employ military power against us.*

Taking the above extract, we first remove (identify in red) all stop words and grammatical marks, with the following result,

> *In the case of the submarine force , the Government takes the view that our future strategic circumstances necessitate a substantially expanded submarine fleet of 12 boats in order to sustain a force at sea large enough in a crisis or conflict to be able to defend our approaches (including at considerable distance from Australia , if necessary) , protect and support other ADF assets , and undertake certain strategic missions where the stealth and other operating characteristics of highly-capable advanced submarines would be crucial . Moreover , a larger submarine force would significantly increase the military planning challenges faced by any adversaries , and increase the size and capabilities of the force they would have to be prepared to commit to attack us directly, or coerce , intimidate or otherwise employ military power against us .*

[5] See the URL, `http://www.defence.gov.au/whitepaper/`

Finally, we delete the stop words and write out the final text subset as a multiset, showing multiplicities of repeated words,

> *case submarine*(×4) *force*(×4) *Government view future strategic*(×2) *circumstances necessary*(×2) *substantially expanded fleet 12 boats order sustain sea large crisis conflict defend approaches including considerable distance Australia protect support ADF assets undertake certain missions stealth operating characteristics highly-capable advanced crucial larger significantly increase*(×2) *military*(×2) *planning challenges faced adversaries size capabilities prepared commit attack directly coerce intimidate otherwise employ power*

Here, the blue text and its given multiplicities, might be thought of as a type of automated speed reading text, where the filtered multiset of interest might be,

$$\{\textit{submarine}(\times 4),\ \textit{force}(\times 4),\ \textit{strategic}(\times 2),\ \textit{necessary}(\times 2),\ \textit{increase}(\times 2),\ \textit{military}(\times 2)\}.$$

Remark 5.3. *The bag of words representation should not be confused with the usual mathematical notion of a set, as it allows repeated elements. However, this representation may be thought of as a multiset (see* `http://mathworld.wolfram.com/Multiset.html`*) which is a set-like object allowing multiplicity of elements.*

Remark 5.4. *As an aside, the Bag of Words representation of text is also used in some forms of SPAM filtering, where one "Bag" contains words in legitimate emails and another "Bag" contains words of a potentially dubious nature.*

5.3 The Notion of Exchangeable Random Variables

In what follows we will see that the bag of words representation has a statistical modelling significance for topic estimation. In particular its invariance to orderings of words which can be expressed through the notion of exchangeability.

Theorem 5.1 (Exchangeability, Bruno de Finetti). *A finite sequence of random variables* $\{X_1, X_2, \ldots, X_m\}$ *is said to be exchangeable if its joint probability distribution is invariant to permutations. That is, for a permutation mapping of the indexes* $\{1, 2, \ldots, n\}$*, which we denote by* $\pi(\cdot)$*, the following equality (in probability distribution) always holds,*

$$\{X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(m)}\} =_d \{X_1, X_2, \ldots, X_m\}. \tag{5}$$

Here the equality $=_d$ means equality in probability distribution. Detailed proofs of this interesting Theorem can be found in [CT78] and [Dur11].

5.4 Specific Natural Language Processing Routines

In this section we describe some NLP routines which are used to convert raw text data in a form amenable to basic tasks of interest, for example probabilistic topic estimation. Most of these tasks are essentially basic pre-processing, however, schemes such as acronym detection are important in our context given the prevalence in Defence of acronyms and identical acronyms with different meanings.

5.4.1 Acronym Detection Scheme

In general most modern text will contain acronyms, however text collected on a Defence subject matter will almost surely contain acronyms. Moreover, its not uncommon in Defence for the same acronym to have two or more distinct meanings. Consequently we implement a scheme to identify acronyms. This scheme is used to identify and display acronyms and also to remove all acronyms from the Bag of Words reduction of the text. The algorithm we use is given in Algorithm 1.

Algorithm 1 Algorithm for detecting Acronyms

1. **for** documents $i = 1$ to I **do**
2. **for** words $\ell = 1$ to L^i **do**
3. **if** word is identified as a stop word, **then**
4. remove word
5. **end if**
6. **if** the word contains a character which is not a letter, or the period ".", **then**
7. remove word.
8. **end if**
9. **if** more than half of the characters of the word are capitalised, **then**
10. consider word as a candidate acronym.
11. **end if**
12. **if** number of acronym names in this sentence is more then half of the words in this sentence, **then**
13. remove all these words from the candidate list. {If a sentence is all capitalised its words will not be classed as acronym}
14. **end if**
15. **if** word's length is > 6 characters, **then**
16. remove word from list of candidate acronyms.
17. **end if**
18. **end for**
19. **end for**

5.4.2 Named Entity Recognition (NER) Scheme

The named entity recogniser (NER) code used in this work is publicly available through the library at `http://code.google.com/p/nicta-ner/`. This library includes an Extractor, which extracts all the first-letter-uppercase words from text, and combines continuous words into phrases. The result set is then passed to a Named Entity Classifier. The Classifier will score all the phrases and indicate if a phrase pertains to one of the following three classes: 1) Location, 2) Person, or 3) an Organization.

5.4.3 Named Entity (NE) Extractor

The NE Extractor is also a scheme developed by NICTA. It tokenises the text by using a Java standard tokeniser class. This scheme is package independent and has a success rate of the order of 75%. The NE Extractor will iterate throughout the tokens and estimate if the tokens correspond to Named Entities. Special cases arise when estimating the first word of a sentence, as this word is usually first-letter-uppercase. Further, a Word Dictionary and some special rules are also implemented to determine if the word corresponds to a name or not. The Word Dictionary is created utilising an open source word list project (see `http://wordlist.sourceforge.net`) and a parts-of-speech Tagger. The Dictionary can be used to determine if the first word in the sentence is a named entity in most cases, however, exceptions are catered for with special rules.

An additional function of the NE Extractor is the extraction Time/Date phrases from text. The NE Extractor can be used to determine if a single word is likely to be a Time/Date word (for example: 21:30, 3rd, 2001, February). If a time/date phrase is extracted from the text, this "word" will be added to a Date-Phrase-Model which iterates to the next word in the text. This search loop will terminate if any word that does not look like a Time/Date word appears. The Date-Phrase-Model will determine if the combination tagged is a time/date phrase or not. For example, a phrase such as "16th" is not a date phrase, as it's ambiguous. However, the term "16th, February" is unambiguously a date-phrase.

5.4.4 Named Entity (NE) Classifier

The NE Classifier we use is a scheme developed by NICTA which implements a numeric scoring of classes based on several features. This classifier receives a phrase string as input and returns a numeric score value. Generally two classes of features can be found by this classifier. The first of these features will test if any key-words, or key-phrases appear embedded within the given phrase. If the classifier returns "true", then the phrase contains the key-word or key-phrase in a word list. The second type of feature extracts the information in the context such as the attached prepositions.

6 Probabilistic Topic Modelling

In general, the two basic aims or probabilistic topic modelling are: 1) to identify and rank a finite set of topics that pervade a given corpus and 2) to annotate/tag documents within a corpus according to the topics they concern.

Topic modelling is an increasingly useful class of techniques for analysing not only large unstructured documents but also data that posit "*bag-of-words*" assumption, such as genomic data [FGK+05] and discrete image data [WG08]. As a promising unsupervised learning approach with wide application areas, it has gained significant momentum recently in machine learning, data mining and natural language processing communities. In this section we review fundamentals (e.g. basic idea and posterior inference) of topic models, especially the Latent Dirichlet Allocation (LDA) model by [BNJ03] that acts as a benchmark model in the topic modelling community.

6.1 Probabilistic Topic Models

Probabilistic topic models [DDF$^+$90, Hof99, Hof01, BNJ03, GK03a, BJ06, SG07, BL09, Hei08] are a discrete analogue to principal component analysis (PCA) and independent component analysis (ICA) that model *topic* at the word level within a document [Bun09]. They have many variants such as Non-negative Matrix Factorisation (NMF) [LS99], Probabilistic Latent Semantic Indexing (PLSI) [Hof99] and Latent Dirichlet Allocation (LDA) [BNJ03], and have applications in fields such as genetics [PSD00, FGK$^+$05], text and the web [WC06, BSB08], image analysis [LP05, WG08, HZ08, CFF07, WBFF09], social networks [MWCE07, MCZZ08] and recommender systems [PG11]. A unifying treatment of these models and their relationship to PCA and ICA is given by [BJ06].

Specifically, probabilistic topic models are a family of generative[6] models for uncovering the latent semantic structure of a corpus by using a hierarchical Bayesian analysis of the text content. The fundamental idea is that each document is a mixture[7] of latent topics, where each topic has a probability distribution over a vocabulary of words. A topic model is a factor model that specifies a simple probabilistic process by which documents can be generated. It reduces the complex process of generating a document to a small number of probabilistic steps by assuming exchangeability. While the model just described is unrealistic as a "true" model of language generation, it is interesting enough to generate useful semantics that we can employ in understanding a collection. Probability density mixture models are widely used in stochastic modelling, in particular Gaussian mixtures, for example see [MP, Mak71] . To loosely fix the basic ideas here with a crude example, we might begin by considering the probability that a given word "w'" is associated to one of a finite collection of topics. This probability might be modelled as follows,

$$\begin{aligned} p(w = w') &= \sum_{j=1}^{K} p(w' \mid T_j) p(T_j), \\ &= \sum_{j=1}^{K} \kappa_j p(T_j). \end{aligned} \tag{6}$$

Here T_j denotes topic j. The κ_j may be thought of as weights in a convex combination of topic-proportion probabilities.

To "generate" a new document, a distribution over topics (i.e., a *topic distribution*) is first drawn from a probability distribution over finite topic vectors. Then, each word in that document is drawn from a *word distribution* associated with a topic drawn from the *topic distribution*. The semantic properties of words and documents can be expressed in terms of probabilistic topics.

[6] In our context the term "generative model" refers to a stochastic model used to generate typical realisations of observed data, that is we assume there exists a stochastic model whose outputs are (loosely speaking) a document. Such models typically have known dependencies upon hidden/latent parameters. The usual estimation task with generative models is to apply Bayesian inference to compute/estimate hidden variables, such as topic information *etc.* A classic example of this class of estimation task is the celebrated Kalman Filter.

[7] Here the term mixture refers to an **admixture**, for example concerning population modelling see the article [SBF$^+$11]. Some synonyms of admixture are, blend, alloy, amalgamation.

In what follows we take the dimensions for a topic estimation task as defined through three index sets $(\mathcal{K}, \mathcal{V}, \mathcal{I})$, with

$$\mathcal{K} \triangleq \{1, 2, \ldots, K\}, \tag{7}$$

$$\mathcal{V} \triangleq \{1, 2, \ldots, V\}, \tag{8}$$

$$\mathcal{I} \triangleq \{1, 2, \ldots, I\}. \tag{9}$$

Here K denotes the number of topics, V the size (in number of words) of the vocabulary and I is the number of documents in the corpus being studied.

Formally, a topic model can be interpreted in terms of a mixture model as follows. Suppose $\boldsymbol{\mu}^i = (\mu_1^i, \mu_2^i, \ldots, \mu_K^i)$ is a document-specific *topic probability distribution*, here $i \in \{1, 2, \ldots, I\}$ is an index to a specific document in the corpus. Further $\mu_k^i \in [0, 1]$ and

$$\mu_k^i = \Pr(\text{Document } i \text{ contains/concerns Topic } k). \tag{10}$$

Suppose a real-valued $V \times K$ (Vocabulary $\times$ Topics) matrix $\boldsymbol{\Phi}$, is a column-wise collection of topic-specific *word probability distributions*. It's important here to note that the matrix $\boldsymbol{\Phi}$ is not document-wise, rather, it applies to the *entire* corpus. To fix this idea, consider three typical topics one might find in text concerning defence capability. These might be, for example, Force Structure Review (FSR), Fundamental Inputs to Capability (FIC[8]) and the topic of Submarines. At the outset one would expect that each of these topics might weight certain words differently. An idealised example of these differences is shown in Table 6. Continuing we write $z_\ell^i \in \mathcal{K}$ as an index/label to a specific topic-word association

Table 6: *Typical(idealised) word-topic allocation probability vectors concerning some areas of defence capability. Note that in this example FIC appears in two different topics, here FSR and Submarine. Note also that the number of most-significant words per topic may vary. The vertical ellipses indicate that the shown words are proper subsets of a greater vocabulary which we denote by V.*

FSR	$\boldsymbol{\Phi}_{\{:,1\}}$
⋮	⋮
cost	0.09
delivery	0.034
FIC	0.034
workforce	0.033
white paper	0.023
risk	0.017
scenarios	0.015
⋮	⋮

FIC	$\boldsymbol{\Phi}_{\{:,2\}}$
⋮	⋮
basing	0.031
facilities	0.027
major systems	0.019
organisation	0.013
personnel	0.011
support	0.011
⋮	⋮

Submarine	$\boldsymbol{\Phi}_{\{:,3\}}$
⋮	⋮
diesel	0.136
FIC	0.042
nuclear	0.025
periscope	0.011
deterrent	0.011
snorkel	0.009
sonar	0.02
torpedo	0.012
⋮	⋮

[8]FIC refers to a canonical list of components which collectively form a defence capability, for example: organisation, personnel, facilities, command and management *etc.*

for word w_ℓ^i, where $\ell \in \{1, 2, \ldots, L^i\}$ (here $L^i \in \mathbb{N}$ is the number of words in document i). The basic statistical sampling process by which we assume a particular document might have been created is as follows,

$$\text{for } \ell = 1, 2, \ldots, L^i \qquad z_\ell^i \sim \boldsymbol{\mu}^i \tag{11}$$

$$w_\ell \mid (z_\ell^i,\, \boldsymbol{\Phi}_{\{:,\, z_\ell^i\}}) \sim F(\boldsymbol{\Phi}_{\{:,\, z_\ell^i\}}), \tag{12}$$

Here and throughout this report, the shorthand notation $a \mid (b, c) \sim d$ indicates that the random variable a, given (b, c) is distributed according to d. The function $F(\cdot)$ is set, in general, to be a discrete distribution. Further, a Dirichlet distribution is assumed as a prior for $\boldsymbol{\mu}$. The *hypothetical* output of the model just described would be a document of the following form,

$$\textit{"Document"} \triangleq \big\{(w_1, z_1), (w_2, z_2), \ldots (w_L, z_L)\big\}. \tag{13}$$

Remark 6.1. *Its important here to understand precisely what the stochastic generative process just described actually means. No "real" intelligible document will ever be generated from such a process, rather, a bag of words collection is generated with topic-word association according to the probability models used. This is quite distinct from some other more well known stochastic models, for example a discrete-time Gauss-Markov system used in a typical Kalman filtering target tracking problem. In that setting a plausible but noise corrupted target track is generated.*

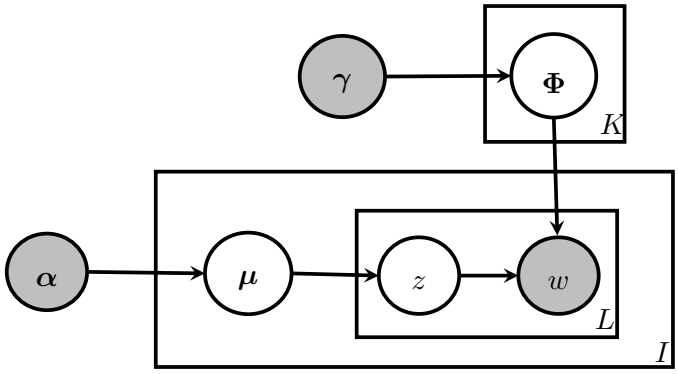
Applying standard Bayesian inference techniques, we can invert the generative process, (described by equations (11) & (12)), to infer the set of optimal latent topics that maximise the likelihood (or the posterior probability) of a collection of documents. Compared with a purely spatial representation (e.g., Vector Space Model [SM86]), the superiority of representing the content of words and documents in means of probabilistic topics is that each topic can be individually interpreted as a probability distribution over words, it picks out a coherent cluster of correlated terms [SG07]. We should also note that each word can appear in multiple clusters, just with different probabilistic weights, which indicates topic models could be able to capture polysemy[9] [SG07]. This generative process is purely based on the "*bag-of-words*" assumption where only word occurrence information (i.e. frequencies) is taken into consideration. This corresponds to the assumption of *exchangeability* in Bayesian Statistics (see [BS94]). However, word-order is ignored even though it might contain important contextual cues to the original content.

6.2 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) [BNJ03], is the form of topic model considered here. It is a fully Bayesian extension of the PLSI model, is a three-level hierarchical Bayesian model for collections of discrete data, e.g. documents. It is also known as multinomial PCA [Bun02]. One can show that two other popular models, PLSI and NMF, are closely related to maximum likelihood versions of LDA [GK03b, BJ06].

[9] The term *polysemy* refers to a symbol/sign/word/phrase having a diversity of meanings. In our context this refers to words and/or phrases which have more than one semantic meaning.

Figure 3: *A graphical model representation of Latent Dirichlet Allocation. Here the only observed data are the words, denoted by the shaded circle containing W*



As a fundamental approach for topic modelling, LDA is usually used as a benchmark model in the empirical comparison with its various extensions and related models.
Figure 3 illustrates the graphical representation of LDA using plate notation (see [Bun94] for an introduction; further details on Graphical models can also be found in [Edw00], [Lau96b], [CDLS99] and [KF09]. In this notation, shaded and unshaded nodes indicate observed and unobserved (i.e. latent or hidden) random variables respectively; edges indicate dependencies among variables; and plates indicate replication. For example the upper most plate is replicated K times for K topics and the largest plate is replicated I times over the number of documents in the corpus. For document analysis, LDA offers a hidden variable probabilistic topic model of documents. The observed data are known word collections

$$\boldsymbol{w}^{\,i} \triangleq \{w_1^i, w_2^i, \ldots, w_{L^i}^i\}, \tag{14}$$

each corresponding to a document in the corpus of the I documents being considered. To denote all collections of words in the corpus, that is the set of sets $\{\boldsymbol{w}^{\,i}\}_{i=1:I}$, we use the shorthand notation $\boldsymbol{W} \triangleq \boldsymbol{w}^{1:I}$, where

$$\boldsymbol{w}^{1:I} = \left\{w_1^1, \ldots, w_{L^1}^1, w_1^2, \ldots, w_{L^2}^2, \ldots, w_1^I, \ldots, w_{L^I}^I\right\} \tag{15}$$

Similarly we write

$$\boldsymbol{z}^{\,i} \triangleq \{z_1^i, z_2^i, \ldots, z_{L^i}^i\}, \tag{16}$$

for document i's word-topic associations (these quantities are integer-valued with $z_\ell^i \in \mathcal{K}$). As above, the complete corpus collection of these associations are denoted by $\boldsymbol{Z} \triangleq \boldsymbol{z}^{1:I}$. Finally we write $\boldsymbol{\mu}$ to denote the collection of $\left\{\boldsymbol{\mu}^1, \boldsymbol{\mu}^2, \ldots, \boldsymbol{\mu}^I\right\}$. The two classes of hidden[10] variables we consider for each document in the corpus are:

1. The topic probability distribution which we denote by the K-component vector $\boldsymbol{\mu}^i$ (here i labels the i-th document in the corpus)

2. The word-topic assignments which are $\{z_1^i, \ldots, z_{L^i}^i\}$.

[10]Here the term "hidden" does not mean hidden as in common parlance, rather it means a quantity that is indirectly observed.

Further basic model parameters are the Dirichlet-prior parameters

$$\boldsymbol{\alpha} \triangleq \{\alpha_1, \alpha_2, \ldots, \alpha_K\}, \tag{17}$$

$$\boldsymbol{\gamma} \triangleq \{\gamma_1, \gamma_2, \ldots, \gamma_{L^i}\}. \tag{18}$$

These parameters are for topic and word distributions respectively. Note that for all components of these parameters, $\alpha_k \in \mathbb{R}_+$ and $\gamma_\ell \in \mathbb{R}_+$, but the sums $\sum_{k=1}^{K} \alpha_k$ and $\sum_{\ell=1}^{L^i} \gamma_\ell$ are unconstrained. Additional model parameters are the word probability distributions (per topic) as collected column-wise in the matrix $\boldsymbol{\Phi}$.

We write $\mathrm{Dir}_K(\cdot)$ to indicate a K-dimensional Dirichlet distribution. The LDA model assumes that documents are consequences of the following generative process:

1. For each topic $k \in \mathcal{K}$,
   (a) choose[11] a word probability distribution according to, $\boldsymbol{\Phi}_{\{:,k\}} \sim \mathrm{Dir}_V(\boldsymbol{\gamma})$.
2. For each document $i \in \mathcal{I}$,
   (a) choose a document-specific topic probability distribution according to, $\boldsymbol{\mu}^i \mid \boldsymbol{\alpha} \sim \mathrm{Dir}_K(\boldsymbol{\alpha})$.
   (b) For each $\ell \in \{1, \ldots, L^i\}$,
      i. choose a topic-word association according to, $z_\ell^i \mid \boldsymbol{\mu}^i \sim \mathrm{Discrete}(\boldsymbol{\mu}^i)$,
      ii. choose a word according to, $w_\ell^i \mid \left(z_\ell^i, \boldsymbol{\Phi}_{\{:,z_\ell^i\}}\right) \sim \mathrm{Discrete}(\boldsymbol{\Phi}_{\{:,z_\ell^i\}})$.

Here, the hyper-parameter[12] $\boldsymbol{\gamma}$ is a Dirichlet prior on *word distributions* (i.e. a Dirichlet smoothing on the multinomial parameter $\boldsymbol{\Phi}$ [BNJ03]). The model parameters can be estimated from the data. The hidden variables can be inferred for each document by inverting the generative process, which are useful for ad-hoc document analysis, for example, information retrieval [WC06] and document summarisation [AR08a, AR08b]. With this process, LDA models documents on a low-dimensional topic space[13], which provides not only an explicit semantic representation of a document, but also a hidden topic decomposition of the document collection [BL09].

Definition 6.1 (LDA Count Frequencies). *In LDA one requires the observed counts/frequencies for the respective multinomial probability distributions. We denote these counts by*

$$\boldsymbol{n}^k \triangleq \left(n_1^k, n_2^k, \ldots, n_V^k\right). \tag{19}$$

Here the components n_v^k denote the number of times word $v \in \{1, 2, \ldots, V\}$ is assigned a specific topic $k \in \{1, \ldots, K\}$.
Similarly we write

$$\boldsymbol{m}^i \triangleq \left(m_1^i, m_2^i, \ldots, m_K^i\right). \tag{20}$$

[11] Here the term "choose" means a stochastic choice, that is, sample from a probability distribution.
[12] The term "hyper-parameter" derives from Bayesian statistics and refers to the parameter of a prior probability distribution.
[13] Note the number of topics associated with a document collection is usually far smaller than the vocabulary size, since documents in a collection tend to be heterogeneous.

Here the components m_k^i denote the number of times topic k is assigned to some word tokens in document i. Note also that in document i, the following sum always holds,

$$\sum_{k=1}^{K} m_k^i = L^i. \tag{21}$$

The integer-valued quantities n_v^k and m_k^i are functions of the integer values z_ℓ^i and also w_ℓ^i, and so may be computed via indicator functions, that is,

$$n_v^k = \sum_{\substack{i=1 \\ \ell=1}}^{L^i,I} \mathbf{1}_{\{z_\ell^i=k\}} \mathbf{1}_{\{w_\ell^i=v\}}, \tag{22}$$

$$m_k^i = \sum_{\ell=1}^{L^i} \mathbf{1}_{\{z_\ell^i=k\}}. \tag{23}$$

Given the two Dirichlet priors parametrised by $\boldsymbol{\alpha}$, and $\boldsymbol{\gamma}$, and the observed document collection $\boldsymbol{W}$, the full joint probability distribution, including the the latent variables $\{\boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{\mu}, \boldsymbol{\Phi}\}$, may be written down directly by combining the information in Figure 3, and the probability distributions given above for an LDA generative process. This joint probability distribution, conditioned on the Dirichlet priors $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, has the form,

$$\begin{aligned} p(\boldsymbol{\mu}, \boldsymbol{\Phi}, \boldsymbol{Z}, \boldsymbol{W} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) &\propto \underbrace{\left(\prod_{\ell=1}^{L^i} \underbrace{p(z_\ell^i \mid \boldsymbol{\mu}^i)}_{\text{Multinomial}} \underbrace{p(w_\ell^i \mid \boldsymbol{\Phi}_{\{:,z_\ell^i\}})}_{\text{Multinomial}} \right)}_{\text{likelihood}} \underbrace{\left(\prod_{k=1}^{K} \underbrace{p(\boldsymbol{\Phi}_{\{:,k\}} \mid \boldsymbol{\gamma})}_{\text{Dirichlet}} \prod_{i=1}^{I} \underbrace{p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha})}_{\text{Dirichlet}} \right)}_{\text{prior}}. \\ &= \left(\prod_{k=1}^{K} \frac{1}{\text{Beta}_V(\boldsymbol{\gamma})} \prod_{v=1}^{V} \left(\boldsymbol{\Phi}_{\{k,v\}}\right)^{\gamma_v + n_v^k - 1} \right) \\ &\quad \times \left(\prod_{i=1}^{I} \frac{1}{\text{Beta}_K(\boldsymbol{\alpha})} \prod_{k=1}^{K} \left(\mu_k^i\right)^{\alpha_k + m_k^i - 1} \right). \end{aligned} \tag{24}$$

In our stochastic generative model we assume the following samplings,

- ◇ the variables in the matrix $\boldsymbol{\Phi}$'s are corpus level variables, which are assumed to be sampled once for the corpus,
- ◇ the document level variables $\boldsymbol{\mu}^i$'s are sampled once for each document,
- ◇ the variables z_ℓ^i's are word level variables that are sampled once per word in each document.

We note that the probability $p(w_\ell^i \mid \boldsymbol{\Phi}_{\{:,z_\ell^i\}})$ simplifies to probability value $\boldsymbol{\Phi}_{\left\{z_\ell^i, w_\ell^i\right\}}$, so the likelihood

$$L(\boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{w} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) \triangleq p(\boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{w} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}), \tag{25}$$

is itself a product of likelihood terms, as is seen by the the properties of Dirichlet distributions (see D.1). Given the observed document collection $\boldsymbol{W}$, the basic task of Bayesian inference is to compute the posterior probability distribution over the model parameters $\boldsymbol{\Phi}$ and the latent variables, $\boldsymbol{\mu}$, and $\boldsymbol{Z}$. The posterior is

$$p(\boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{\Phi} \mid \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) = \frac{p(\boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{W}\boldsymbol{\Phi} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma})}{\int_{\mathrm{Dom.}(\boldsymbol{\mu})} \int_{\mathrm{Dom.}(\boldsymbol{\Phi})} \Big\{ \sum_{\mathrm{Dom.}(\boldsymbol{Z})} p(\boldsymbol{\mu}, \boldsymbol{Z}, \boldsymbol{W} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) \Big\} d\boldsymbol{\mu} d\boldsymbol{\Phi}}. \tag{26}$$

Although the LDA model is a relatively simple model, a direct computation of this posterior is clearly infeasible due to the summation over topics in the integral of the denominator. Further, training LDA on a large collection with millions of documents can be challenging and efficient exact algorithms have not yet been found for such tasks, see [Bun09]. Consequently we appeal to approximate-inference algorithms, in particular: the mean field variational inference [BNJ03], the collapsed variational inference [TNW07], the expectation propagation [ML02], and Gibbs sampling [GS04]. The article [BJ06] has given a detailed discussion on some of these methods and suggested alternatives, such as; the direct Gibbs sampling by [PSD00] and Rao-Blackwellised Gibbs sampling by [CR96]. Furthermore, [WMM09] has studied several classes of structured priors for the LDA model, i.e. asymmetric or symmetric Dirichlet priors on $\boldsymbol{\mu}$ and $\boldsymbol{\Phi}$. They have shown that the LDA model with an asymmetric prior on $\boldsymbol{\mu}$ significantly outperforms that with a symmetric prior. However, there are no benefits from assuming an asymmetric prior for $\boldsymbol{\Phi}$.

Out of all proposed approximate inference algorithms, each of which has advantages and disadvantages, hereafter we focus on the collapsed Gibbs sampling algorithm introduced in [GS04], details can be found in [SG07]. The collapsed Gibbs sampler is found to be as good as others. It is also general enough to be a good base for extensions of LDA.

6.3 Numerical Implementation

Gibbs sampling [GG90] is a special case of the *Metropolis-Hastings* algorithm in the Markov chain Monte Carlo (MCMC) family. The first collapsed Gibbs sampling algorithm for the LDA model is proposed by [GS04] and is known as the Griffiths and Steyvers' algorithm. It marginalises out $\boldsymbol{\mu}$ and $\boldsymbol{\Phi}$ from Equation (26) using the standard normalising constant for a Dirichlet. The strategy of marginalising out some hidden variables is usually referred to as "collapsing" [Nea00], which is the same as Rao-Blackwellised Gibbs sampling [CR96]. The collapsed algorithm samples in a collapsed space, rather than sampling parameters and hidden variables simultaneously [TNW07]. So, the Griffiths and Steyvers' algorithm is also known as a collapsed Gibbs sampler.

The principle of Gibbs sampling is to simulate the high-dimensional probability distribution by conditionally sampling a lower-dimensional subset of variables via a Markov chain, given the values of all the others are fixed. Essentially this means reducing a large joint probability distribution to a collection of univariate probability distributions. The sampling proceeds until the chain becomes stable (i.e. after the so-called "*burn-in*" period, the chain will burn-in to a stable local optimum). Theoretically, the probability distribution drawn from the chain after the "*burn-in*" period will asymptotically approach the true

posterior distribution. In regard to the LDA model, the collapsed Gibbs sampler considers all word tokens in a document collection, and iterates over each token to estimate the probability of assigning the current token to each topic, conditioned on topic assignments of all other tokens.

To derive the conditional distributions of interest, we first need to compute the joint distribution over $\boldsymbol{Z}$ and $\boldsymbol{W}$, conditioned on the hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$. From equation (24) we see that

$$\begin{aligned} p(\boldsymbol{Z}, \boldsymbol{W} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) &= \int_{\text{Dom.}(\boldsymbol{\Phi})} \int_{\text{Dom.}(\boldsymbol{\mu})} p(\boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{\Phi}, \boldsymbol{\mu} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) d\boldsymbol{\Phi} d\boldsymbol{\mu} \\ &\propto \int \int \prod_{k=1}^{K} p(\boldsymbol{\Phi}_{\{:,k\}} \mid \boldsymbol{\gamma}) \prod_{i=1}^{I} p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha}) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) p(w_\ell^i \mid \boldsymbol{\Phi}_{\{:,z_\ell^i\}}) d\boldsymbol{\Phi} d\boldsymbol{\mu} \\ &= \left[\int \left(\prod_{k=1}^{K} p(\boldsymbol{\Phi}_{\{:,k\}} \mid \boldsymbol{\gamma}) \prod_{i=i}^{I} \prod_{\ell=1}^{L} p(w_\ell^i \mid \boldsymbol{\Phi}_{\{:,z_\ell^i\}}) \right) d\boldsymbol{\Phi} \right] \times \\ &\qquad \left[\int \left(\prod_{k=1}^{K} p(\mu \mid \boldsymbol{\alpha}) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) \right) d\boldsymbol{\mu} \right]. \end{aligned} \tag{27}$$

Remark 6.2. *For brevity we have assumed all documents are of the same length in respect of number of words, that is $L^i = L, \forall i \in \{1, 2, \ldots, I\}$. This simplification does not effect the end results for a corpus containing different document lengths.*

To further evaluate the RHS of equation (33) we consider the last integral (immediately above) against $\boldsymbol{\mu}$. Due to the assumptions of independence we note that

$$\begin{aligned} \int \prod_{i=1}^{I} p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha}) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) d\boldsymbol{\mu} &= \int \cdots \int p(\boldsymbol{\mu}^1 \mid \boldsymbol{\alpha}) p(\boldsymbol{\mu}^2 \mid \boldsymbol{\alpha}) \cdots p(\boldsymbol{\mu}^I \mid \boldsymbol{\alpha}) \times \\ &\qquad \prod_{\ell=1}^{L} p(z_\ell^1 \mid \boldsymbol{\mu}^1) \prod_{\ell=1}^{L} p(z_\ell^2 \mid \boldsymbol{\mu}^2) \cdots \prod_{\ell=1}^{L} p(z_\ell^I \mid \boldsymbol{\mu}^I) d\boldsymbol{\mu}^1 d\boldsymbol{\mu}^2 \cdots d\boldsymbol{\mu}^I \\ &= \prod_{i=1}^{I} \int p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha}) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) d\boldsymbol{\mu}^i. \end{aligned} \tag{28}$$

To evaluate the previous product of integrals we need only consider the i^{th} integral. First we recall the explicit form of the Dirichlet probability density, that is,

$$\int \underbrace{p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha})}_{\text{Dirichlet}} \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) d\boldsymbol{\mu}^i = \int \left(\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} (\mu_k^i)^{\alpha_k - 1} \right) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) d\boldsymbol{\mu}^i. \tag{29}$$

The remaining probability density in (29) is a multinomial over the word-topic association indicators z_ℓ^i. Recall that z is integer valued and can take one of $1, 2, \ldots, K$ values, further

several words in any given document may take the same z value and the counts/frequencies of these allocations in document i are denoted by m_k^i. Recalling the sum at equation (21), we note that (ignoring normalization constants)

$$\prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) \propto \prod_{k=1}^{K} \left(\mu_k^i\right)^{m_k^i}. \tag{30}$$

Consequently the integral at (29) may be written as,

$$\begin{aligned} \int p(\boldsymbol{\mu}^i \mid \boldsymbol{\alpha}) \prod_{\ell=1}^{L} p(z_\ell^i \mid \boldsymbol{\mu}^i) d\boldsymbol{\mu}^i &\propto \int \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^{K} (\mu_k^i)^{\alpha_k - 1} \right) \prod_{j=1}^{K} \left(\mu_j^i\right)^{m_j^i} d\boldsymbol{\mu}^i \\ &= \int \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^{K} (\mu_k^i)^{m_k^i + \alpha_k - 1} \right) d\boldsymbol{\mu}^i. \end{aligned} \tag{31}$$

Its clear that the integrand in the last term above is somewhat close to a Dirichlet probability density, except its normalisation term does not match the parameters in its product term. To complete this calculation and eliminate $\boldsymbol{\mu}^i$, we use the fact that the integration over any probability density is unity, that is,

$$\begin{aligned} \int \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^{K} (\mu_k^i)^{m_k^i + \alpha_k - 1} \right) d\boldsymbol{\mu}^i &= \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \right) \int \prod_{k=1}^{K} (\mu_k^i)^{m_k^i + \alpha_k - 1} d\boldsymbol{\mu}^i, \\ &= \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \right) \\ &\times \underbrace{\left[\left(\frac{\prod_{k=1}^K \Gamma(m_k^i + \alpha_k)}{\Gamma(\sum_{k=1}^K (m_k^i + \alpha_k))} \right) \left(\frac{\Gamma(\sum_{k=1}^K (m_k^i + \alpha_k))}{\prod_{k=1}^K \Gamma(m_k^i + \alpha_k)} \right) \right]}_{=1} \\ &\times \int \prod_{k=1}^{K} (\mu_k^i)^{m_k^i + \alpha_k - 1} d\boldsymbol{\mu}^i, \\ &= \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \right) \left(\frac{\prod_{k=1}^K \Gamma(m_k^i + \alpha_k)}{\Gamma(\sum_{k=1}^K (m_k^i + \alpha_k))} \right) \times \\ &\quad \underbrace{\int \left(\frac{\Gamma(\sum_{k=1}^K (m_k^i + \alpha_k))}{\prod_{k=1}^K \Gamma(m_k^i + \alpha_k)} \right) \prod_{k=1}^{K} (\mu_k^i)^{m_k^i + \alpha_k - 1} d\boldsymbol{\mu}^i}_{=1}, \\ &= \left(\frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \right) \left(\frac{\prod_{k=1}^K \Gamma(m_k^i + \alpha_k)}{\Gamma(\sum_{k=1}^K (m_k^i + \alpha_k))} \right). \end{aligned} \tag{32}$$

Similarly one can marginalise out $\boldsymbol{\Phi}$ in equation (33) via calculations similar to those

Algorithm 2 Major cycle of Gibbs sampling for LDA

1. initialise each z_ℓ^i randomly to a topic in $\{1, ..., K\}$
2. **for** documents $i = 1$ to I **do**
3. **for** words $\ell = 1$ to L^i **do**
4. $z_\ell^i \sim p\big(z_\ell^i = k \mid \boldsymbol{Z} \setminus z_\ell^i, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\gamma}\big)$
5. **end for**
6. **end for**

above, the result being,

$$p(\boldsymbol{Z}, \boldsymbol{W} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}) \propto \underbrace{\prod_{i=1}^{I} \left[\left(\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \right) \left(\frac{\prod_{k=1}^{K} \Gamma(m_k^i + \alpha_k)}{\Gamma\big(\sum_{k=1}^{K} (m_k^i + \alpha_k)\big)} \right) \right]}_{\text{over documents}} \times \underbrace{\prod_{k=1}^{K} \left[\left(\frac{\Gamma(\sum_{v=1}^{V} \gamma_v)}{\prod_{v=1}^{V} \Gamma(\gamma_v)} \right) \left(\frac{\prod_{v=1}^{V} \Gamma(n_v^k + \gamma_v)}{\Gamma\big(\sum_{v=1}^{V} (n_v^k + \gamma_v)\big)} \right) \right]}_{\text{over topics}}. \tag{33}$$

Equation (33) provides the required proportionality for the LDA collapsed Gibbs Sampler, with both quantities $\boldsymbol{\mu}$ and $\boldsymbol{\Phi}$ marginalised out. The relation at (33) may be used to compute a univariate sampling density for generating realisations of $\boldsymbol{Z}$, that is, a sampling density of the form

$$p\big(z_\ell^i = k \mid \boldsymbol{Z} \setminus z_\ell^i, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\gamma}\big). \tag{34}$$

Here we write a version of the set difference symbol to indicate all of $\boldsymbol{Z}$ except the element z_ℓ^i, that is

$$\boldsymbol{Z} \setminus z_\ell^i \triangleq \big\{ z_1^1, z_2^1, \ldots z_{L^1}^1, z_1^2, z_2^2, \ldots z_{L^2}^2 \cdots z_1^I, z_2^I, \ldots z_{L^I}^I \big\} \setminus z_\ell^i. \tag{35}$$

Recall the main objective of the collapsed Gibbs sampler is to sample from the univariate probability $p\big(z_\ell^i = k \mid \boldsymbol{Z} \setminus z_\ell^i, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\gamma}\big)$.

Continuing, we note the direct proportionality relationship

$$p\big(z_\ell^i = k \mid \boldsymbol{Z} \setminus z_\ell^i, \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\gamma}\big) \propto p\big(\boldsymbol{Z}, \boldsymbol{W} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}\big). \tag{36}$$

Further, due to statistical independence we need only consider terms affected by the states $z_\ell^i = k$. This is simplified by the key property of the Gamma function, so $\Gamma(n+\alpha+1)/\Gamma(n+\alpha) = n + \alpha$. With a cancellation of factors the full conditional distribution can be derived as

$$p(z_\ell^i = k | \boldsymbol{z}^{1:I} \setminus z_\ell^i, \boldsymbol{w}^{1:I}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) \propto \frac{n_{w_\ell^i}^k + \gamma_{w_\ell^i}}{\sum_{v=1}^{V} \big(n_v^k + \gamma_v\big)} \frac{m_k^i + \alpha_k}{\sum_{\ell=1}^{K} \big(m_\ell^i + \alpha_\ell\big)}. \tag{37}$$

The Gibbs algorithms for sampling the topic assignments $\boldsymbol{Z}$ is given in Algorithm 2. After a sufficient number of Gibbs cycles, which means the sampler has burnt in, the Markov chain is ready to sample. Given the posterior sample statistics, the latent variable $\boldsymbol{\mu}$ and

the model parameter $\mathbf{\Phi}$ can be estimated at any one major cycle using the expectation of the Dirichlet distribution (see §D) as:

$$\widehat{\mathbf{\Phi}}_{\{k,v\}} \triangleq \frac{n_v^k + \gamma_v}{\sum_{v=1}^{V} \left(n_v^k + \gamma_v\right)} \in (0,1], \tag{38}$$

$$\widehat{\boldsymbol{\mu}}_k^i \triangleq \frac{m_k^i + \alpha_k}{\sum_{k=1}^{K} \left(m_k^i + \alpha_k\right)} \in (0,1]. \tag{39}$$

Standard MCMC methodology suggests these values should be averaged across a set of major cycles after "burn in". Then an estimation strategy might be:

1. *"Burn in"* for 400 cycles.
2. Continue major cycles, and at every 20-th major cycle,
   (a) for $k = 1, ..., K$, compute the estimate $\widehat{\mathbf{\Phi}}_{\{:,k\}}$
   and for $i = 1, ..., I$, compute the estimate $\widehat{\boldsymbol{\mu}}^i$,
   (b) compute running averages using these two estimates.

6.4 MCMC-based LDA and the Uniqueness of Outputs

As detailed above, the basic outputs from the LDA implementation we described are the estimated topic probabilities for each document, which we denoted by the vector $\widehat{\boldsymbol{\mu}}^i$. In the previous section we described how these quantities would be estimated using a standard collapsed Gibbs sampling scheme. Given Gibbs sampling is a form of Markov Chain Monte Carlo, this naturally means the outputs of our LDA are themselves inherently random. This fact immediately raises natural questions, for example; how might I best, if at all, interpret the statistics of the LDA outputs ? Does it make sense to compute the estimated variance of these LDA outputs and somehow describe such a variance as a measure of quality ? Ideally we would like such a variance to be small, as this might offer some confidence to the analyst that he/she is near, (in some sense), to the true values. Unfortunately, the answers to these important questions are not so simple.

The theory of this type of convergence analysis is, as yet, not exact or well defined. Indeed, precise convergence bounds in this setting, when developed, require large numbers of samples. As an example of the practice of so-called *convergence diagnostics*, see [CC96]. Note however, this technique is about assessing convergence of estimates; it has some theoretical justification, but is based largely upon heuristics.

The origin of this problem in applying LDA to topic modelling is a basic one of representation and modelling. Suppose we consider K possible topics. Then the LDA model for this scenario has in fact $K! = K \times K-1 \times K-2 \times \cdots \times 1$ equivalent models, all of which are indistinguishable. To put this more simply, recall the matrix $\mathbf{\Phi}$. This matrix is a vocabulary $\times$ topics matrix of probabilities. However there is no good reason to identify

any particular topic with any particular column in this matrix. This means we can consider all of the $K!$ column-wise permutations of $\mathbf{\Phi}$ as valid. Moreover, similar arguments apply to the ordering of the components in the vectors $\boldsymbol{\mu}^i$.

Now, recalling our original concerns, suppose we run the MCMC form of LDA twice on the same text data set and thereby generate two outputs, which will almost surely not be the same. These outputs cannot be easily compared, as their outputs may have arisen from the $K!$ indistinguishable models. This issue is non trivial and remains an as yet unsolved and important problem concerning the use of these LDA methods.

6.5 Visualisation for Topic Estimation

Ultimately all estimated probabilities are of the type $\hat{\mu}(T_j \mid D^i)$. Here T_j denotes topic j and D^j denotes "document" j, which may well be a text "idea/response" rather than an entire document need to be projected, by some means, onto a 2-dimensional space. For example, if the topic number is 4, then for each D^j, our algorithm will compute an estimated (and normalised) probability vector as follows,

$$\begin{aligned}\widehat{\boldsymbol{\mu}}^i &= \big(\hat{\mu}^i_1(T_1 \mid D^i), \hat{\mu}^i_2(T_2 \mid D^i), \hat{\mu}^i_3(T_3 \mid D^i), \hat{\mu}^i_4(T_4 \mid D^i)\big) \\ &\in [0,1] \times [0,1] \times [0,1] \times [0,1] \in \mathbb{R}^4.\end{aligned} \tag{40}$$

Suppose we consider $K \in \mathbb{N}$ for a number of topics. To display the sites/locations of the K topics and estimated probabilities shown in (40), we first arrange the K topic "centres" as equidistant points on a circle, that is $\mathcal{C} \triangleq \big\{(\boldsymbol{X}^{T_1}, \boldsymbol{Y}^{T_1}), \ldots, (\boldsymbol{X}^{T_K}, \boldsymbol{Y}^{T_K})\big\}$ denotes the collection of "centres" appropriately chosen according to a given display. Note that these points are not plotted on the screen, rather, they act as points of location for each topic. Next the collection of all estimated probabilities $\hat{\mu}(T_j \mid D^i)$ are plotted relative to the collection $\mathcal{C}$. The plotting scheme used is a convex combination formulation, for example, the 2-dimensional coordinates for the estimated probability concerning D^i, are computed by,

$$\big(\boldsymbol{X}^{D^i}, \boldsymbol{Y}^{D^i}\big) \triangleq \sum_{j=1}^{K} \hat{\mu}^i_j(T_j \mid D^i) \times \big(\boldsymbol{X}^{T_j}, \boldsymbol{Y}^{T_j}\big). \tag{41}$$

Remark 6.3. *It should be clear that the projection given by (41) is an approximation and that information will be lost when projecting from $K > 2$ down to 2 dimensions onto a visualisation screen. Moreover, this projection is naturally not unique. When projecting from $K > 2$ downs to 2 dimensions, it is possible that distinct points in the probability simplex, may be projected to the same $\big(\boldsymbol{X}^{D^i}, \boldsymbol{Y}^{D^i}\big)$ coordinates. In contrast parallel coordinate plots are unique.*

7 Differential Analysis of Stake-Holder Text

In previous sections of this paper we restricted our attention to the task of topic modelling based upon stochastic generative models and also considerable pre-processing of text, such as the NLP routines described in §5.2 and . In differential analysis we are concerned with

Figure 4: *This figure shows a real-data example for 9 topics projected onto 2D. The clusters of words at the topic centers list the top (probabilistically) 5 words associated to a given topic. The details on these plots is given in section 6.5.*

a slightly different problem. Suppose a collection of two stake-holder groups assemble in the JDSC for an investigation concerning Chinook Helicopters and the use of their hoists. Suppose the two stake-holder groups are ARMY and NAVY, both of which might make use of these Helicopters. In a typical workshop a component discussion may occur on, say, the best operational usage of a Chinook Hoist. Through networked text collection software, participants might enter text offering subject matter expert opinions on such a topic. With this basic example in mind, differential analysis (in the JDSC context) concerns estimating, quantifying or illuminating the differences between two (or more) stake-holder subgroup text responses on the same topic/question. Consequently we might like to estimate basic notions such as bias, polarity or some measure of sentiment indicating the "strength" of responses.

The notion of *differential analysis* of Text immediately raises certain preliminary question, for example, what do we mean by difference ? Is it semantic difference, or some measure of frequency difference ? Moreover, what exactly do we intend to compare ? Is it sentences, noun phrases, paragraphs, or entire documents ? The literature has addressed many of these questions, see the following for some recent and interesting examples [SDK11, CM05, LPW05, Pin04, TVV10, TP09, GST07].

7.1 Quantifying Sentiment

The approach we take in this work is to examine sentiment scores of the text responses elicited through JDSC workshops. Sentiment scores are widely used in opinion mining, see

for example the well known resource SentiWordNet at the URL `http://sentiwordnet.isti.cnr.it`. The (several generations) history and details of SentiWordNet are discussed in the articles [ES06] & [BES10]. See also [TL03]

The basic task here is to score words (numerically) as positive or negative. An additional challenge is to infer (via sentiment) if text is subjective or objective. To score sentiment one typically requires a lexicon. NICTA has developed its own proprietary sentiment score lexicon, which has been applied in this work. A numeric score for a given text idea/response/entry is determined then converted to a colour with the obvious extrema of; strong red for extremely/max negative sentiment and strong green for extremely/max positive sentiment. We first consider this at a word level only (in the Key Phrase section below we also apply sentiment scores to phrases). Recall that we denote a document D^i by the set,

$$D^i \triangleq \{w_1^i, w_2^i, \ldots, w_{L^i}^i\}. \tag{42}$$

Further, write $\rho(\cdot) : V \to [-1, 1]$ for a sentiment function, which maps a given word in the lexicon to a real number in the shown range. In our sentiment analysis we consider two types of computation, the first is a sentiment score per document, the second involves a user query, with sentiment then computed around all instances of the given query. In the first case we compute the sentiment score of the document Sent.(D^i) by a sum, that is,

$$\text{Sent.}(D^i) \triangleq \sum_{j=1}^{L_i} \rho(w_j^i). \tag{43}$$

The calculation shown at (43) is evaluated for all documents in the corpus $\{D1, D^2, \ldots, D^I\}$. Subsequently, each score is rescaled to $[-1, 1]$ through the normalisation Sent.$(D^i)/M$, where

$$M \triangleq \max_{i \in \{1,2,\ldots,I\}} \Big\{ |\text{Sent.}(D^i)| \Big\}. \tag{44}$$

An example of this calculation is shown in Figure 4, here each dot on the simplex labels a document and its colour indicates the computed sentiment.

If the user provides a single word query $q \in V$, then the sentiment calculation is different. In this case document sentiment is only computed for documents containing q one or more times. Suppose a document contains a given query three times, that is

$$D^i = \{w_1^i, q_{\ell_1}, w_3^i, \ldots, q_{\ell_2}, \ldots, q_{\ell_3}, \ldots, w_{L^i}^i\} \tag{45}$$

Here the query word is located at the indices $\{\ell_1, \ell_2, \ell_3\}$. To compute sentiment for this document, with respect to this query, we take a type of windowed average around each occurrence of the query. More generally we write $0 \leq N_q^i \leq L_i$ for the number of matches found in D^i against the query q. The sentiment score for this document is now computed by,

$$\text{Sent.}\big(D^i, q\big) \triangleq \frac{1}{|\mathcal{S}|} \sum_{j \in \mathcal{S}} \Big\{ \rho(q) + \sum_{i=1}^{C} \big\{ \rho(w_{j-1}^i) + \rho(w_{j+1}^i) \big\} \Big\}. \tag{46}$$

Here $\mathcal{S}$ is the set of indices locating the matches of q in the document and $|\mathcal{S}|$ is the cardinality of this set. The window size is $2C + 1$. All outcomes of the calculation shown at (46) are normalised as above to the set $[-1, 1]$. The software developed in this project

also has a capability for multi-word queries, for example "diesel submarine", or "ARMY Helicopter". In this case the formulation of sentiment scoring is similar to that of (46), however the score is computed around instances of the submitted word set. Further, if, for example, a submitted word set is "diesel submarine", these two words need not be adjacent in a given document.

7.2 Visualisation for Sentiment Allocations

In this example for two stake-holder groups only, we consider data derived from the unclassified data set collected by the JDSC in the activity described in §4. In Figure 5 we show a single word-based score for the two groups, NICTA and DSTO. This example is included only to illustrate the sentiment analysis for two groups. No substantial conclusions can be drawn from this data set, it is used for illustration purposes only.

On the left and right of the plot are two vertical histograms for the top N words with respect to a basic frequency count (per stake-holder group). These words are displayed vertically from lowest to highest in frequency. The colours of these histograms indicate the sentiment for how the (respective) groups used the words. Given there is no reason to assume, (in the case of two stake-holder groups), that each will contribute the same volume of text, frequency of usage is computed relative to a group's contribution. For example, suppose two groups are considered, Surface Navy and Submariners, further suppose the Surface Navy contribute twice as much text (on a particular topic) as the Submariners but both use the word *torpedo* in equal proportion. In this case the frequency of usage of this word will be scored the same for both groups.

In the centre of Figure 5 we show an (x, y) graph depiction of all words corresponding to the frequency histograms. Note that if the basic list of words contains 20 entries, then there will be 40 corresponding (x, y) points shown on this plot, which indicate how the same 20 words were used by both groups in respect of (jointly) frequency and sentiment. All (x, y)-located words are tagged by colour to indicate which group they belong to. The y component represents the sentiment value of a word ascending from lowest (most negative or red) to highest (most positive or green). The x coordinate for a given word shows the relative frequency. This axis (for 2 stake-holder scenarios) is really two frequency domains shown side by side. Tagged words in the exact centre of the plate indicate equal usage. To give an example of how frequencies are computed, we write $f_{\text{DSTO}}(w_\ell^i)$ and $f_{\text{NICTA}}(w_\ell^i)$ to denote the frequency of usage of the word w_ℓ^i by the stake-holder groups. In the depiction of Figure 5 the stake-holder group of DSTO is shown on the left side of the figure. Consequently, DSTO-coloured points to the left of the centre of this plot indicated higher frequency of use by DSTO. These particular frequencies (x locations) are computed by

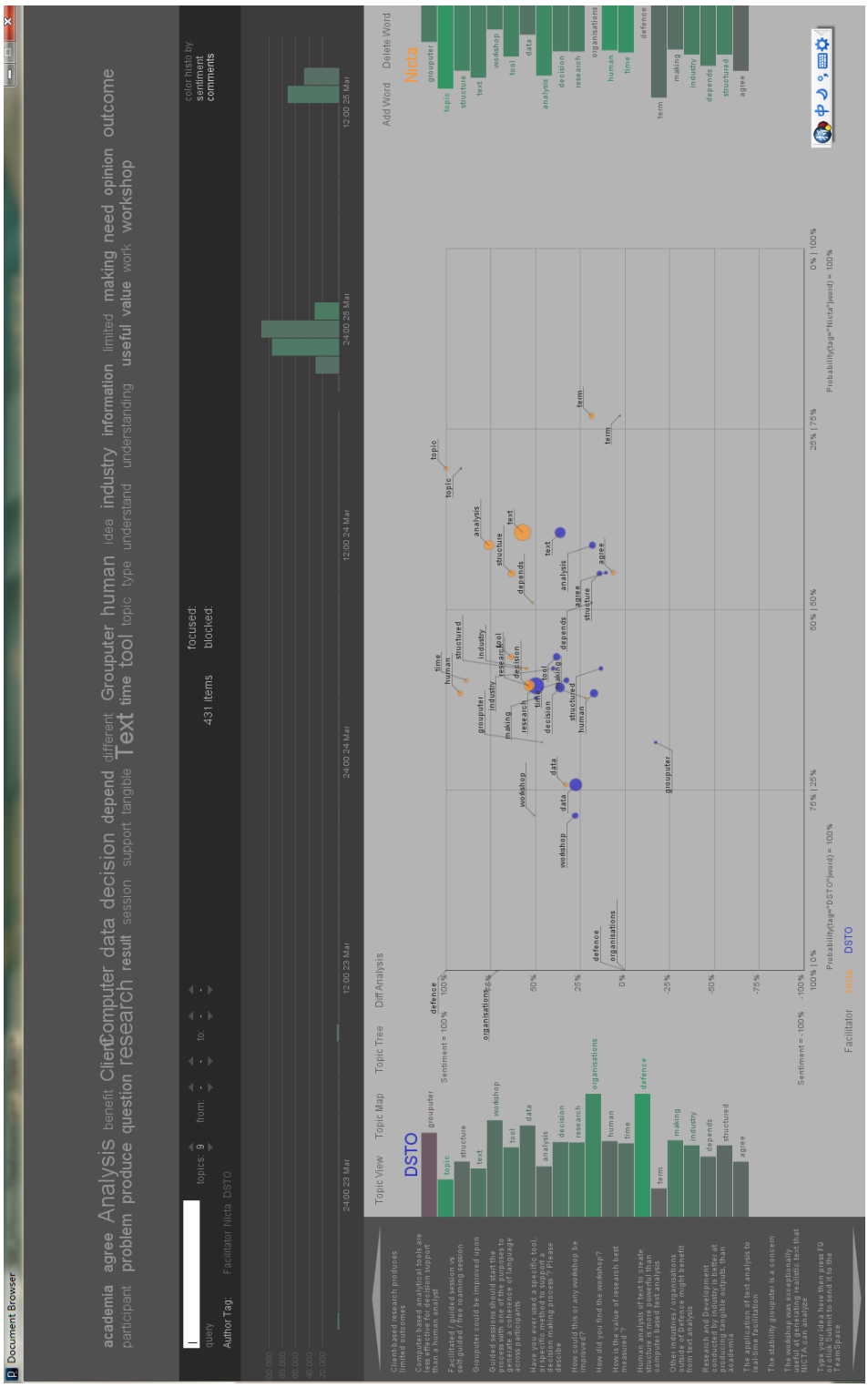
$$x(w_\ell^i) \triangleq \frac{f_{\text{DSTO}}(w_\ell^i)}{f_{\text{DSTO}}(w_\ell^i) + f_{\text{NICTA}}(w_\ell^i).} \tag{47}$$

With x just defined the screen coordinates for a given (DSTO) word are computed as,

$$(x, y) \triangleq \Big(x(w_\ell^i), \text{Sent.}(w_\ell^i)\Big). \tag{48}$$

The corresponding points for NICTA are computed similarly.

Figure 5: *This figure shows a two stake-holder differential analysis at a word level only.*



8 Key-Phrase Identification

The common word "phrase" has a diverse variety of meanings. For example, in music "phrase" has a clear meaning, it is a unit of music that, if extracted, has complete musical sense on its own. In text analysis the notion of a phrase is somewhat similar in respect of what it might convey, however, the classification of phrases in text is far more sophisticated and developed. In text, phrases can be classed in a variety of types, for example: noun phrases, adverbial phrases, adjectival phrases and prepositional phrases, to name just a few. Consequently, we need to identify which class of phrases should be identified and analysed in text data elicited in a defence capability.

In NLP the task of identifying/tagging phrases in text has received considerable attention and a variety of schemes are available to address such tasks. In this work we examine the Multi-Word-Term methods described in [FAM98] and some more basic frequency and sentiment based identification.

8.1 Definitions and Basic Theory

The key-phrase detector we consider here is due to K. Frantzi, S. Ananiadou and H. Mima, see [FAM98]. This paper also cites the following directly related PhD Theses, [Fra98], [Ana88] and [Lau96a]. Some further publications of relevance are: [KU96], [DPL94] and [Dun93]. Unfortunately, a detailed exposition of the interesting results in [FAM98] is well beyond the scope of this report, however, we give here a brief overview of its relevant components. At the outset we note that our term "key-phrase" refers to a Multi Word Term (MWT), typically two or three words in length. The task of identifying such "terms", is usually referred to as Automatic Term Recognition (ATR).

The method developed in [FAM98] is called (by the authors), the C-value/NC-value method. Roughly, this method proposes the combination of two classes of text information to perform ATR, these are; 1) linguistic information (the so-called C-value) and 2) statistical information (the so-called NC-value). Processing the linguistic information requires three components; parts-of-speech tagging, linguistic filters and a specific list of stop words which are not expected to occur in MWTs. The basic notion of parts-of-speech tagging was briefly described in §5.2. Each of these three components will have performance dependent upon the type of text being examined and the specific algorithms being implemented (there are numerous schemes for parts-of-speech tagging). An interesting aspect of the C-value/NC-value method is it uses a type of hybrid linguistic filter. Before describing this filter we recall the notion of regular expressions as they appear in Computer Science.

Definition 8.1 (Regular Expressions). *In computer science, regular expressions are a special type of logical statement specifically constructed for searching and matching purposes, such as matching given strings or given characters etc. More detail on regular expressions can be found in the well known text [SW11], or the URL* `http://introcs.cs.princeton.edu/java/72regular/`. *Various symbols are used in regular expressions. A brief relevant list of these symbols is shown in Table 7.*

The linguistic filter component of [FAM98] is a hybrid or combination of three types of linguistic filters. These filters each search for matches against certain parts-of-speech

Table 7: *Some regular expression symbols*

Symbol	Meaning
$*$	0 or more
?	$\{0, 1\}$
$+$	one or more occurrences
|	logical OR

combinations or specific sequences of parts-of-speech, see, for example [Bou92], [DC95] and [DGL94]. Using the abbreviation Adj. for an adjective and NounPrep for a noun preposition, and the symbols in Table 7, these three linguistic filters are written:

1. $\text{Noun}^{+}\text{Noun}$,
2. $\big(\text{Adj.} \mid \text{Noun}\ \big)^{+}\text{Noun}$,
3. $\Big(\big(\text{Adj.} \mid \text{Noun}\big)^{+} \mid \Big(\big(\ \text{Adj.} \mid \text{Noun}\ \big)^{*}\big(\ \text{NounPrep}\ \big)?\Big)\ \big(\ \text{Adj.} \mid \text{Noun}\ \big)^{*}\Big)\ \text{Noun}$.

To explain this further by way of example, consider the second linguistic filter above. Recall that the exponent of + means one or more occurrences and (Adj. | Noun) is an adjective or a noun. Consequently, a match on this filter would be a sequence of the type, (Adj. | Noun), (Adj. | Noun),. . . ,(Adj. | Noun), Noun.

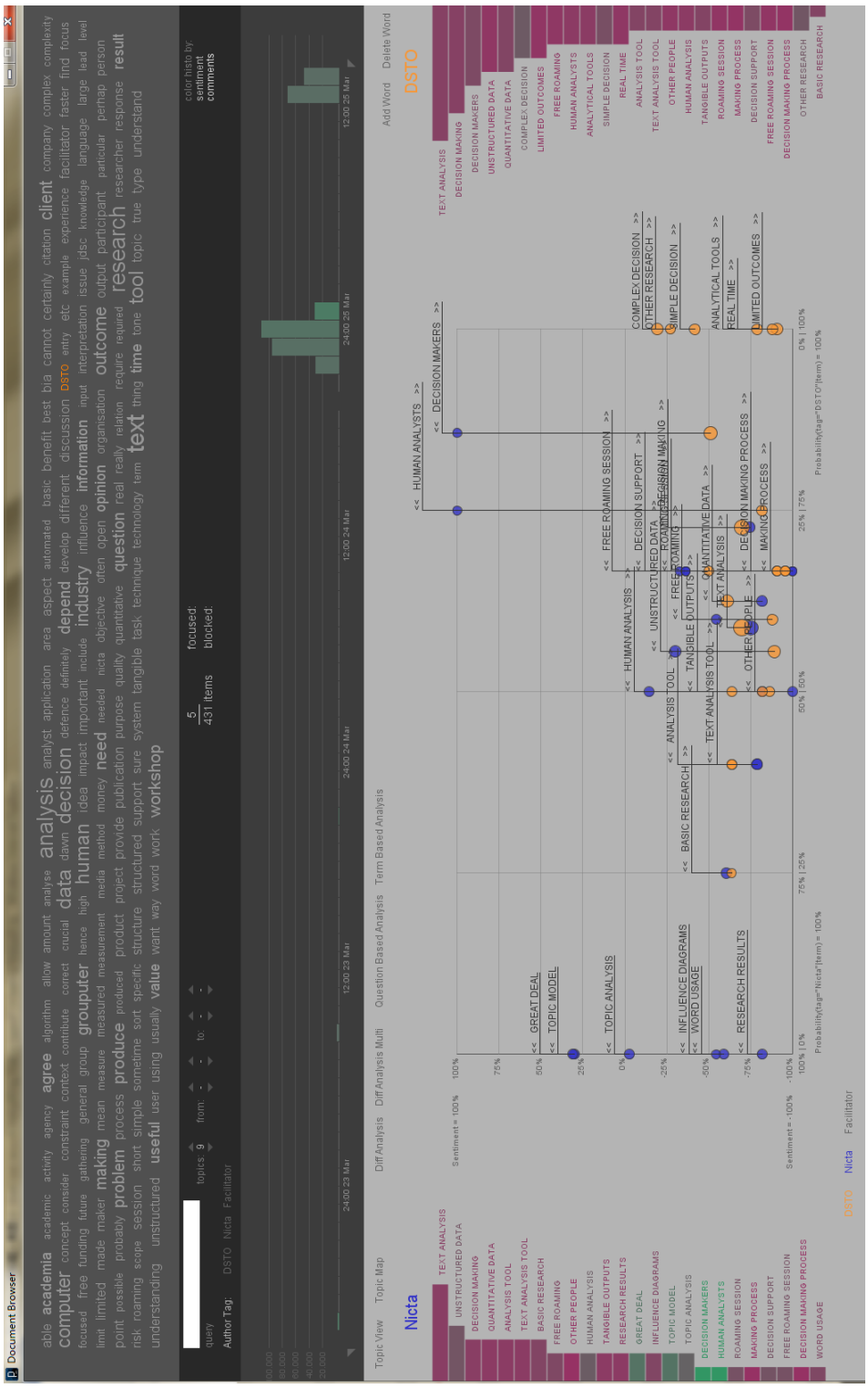
Remark 8.1. *The three listed filters above are meant to indicate a successive compounding of complexity, where 2 is a more complex version of 1 and 3 is a more complex version of 2. Ultimately the filter at 3 is the form we wish to encode for out key-phrase estimation.*

The second, or "statistical" part of the C-value/NC-value algorithm concerns a compensated frequency estimator, which deals with possible complexities/ambiguities of substrings. The interested reader is referred to [FAM98].

8.2 Visualisation For Key Phrases

An example of key phrase estimation for two subgroups is shown in Figure 6. In this example we show ranking of the identified key-phases (or MWTs) and their sentiment is shown by colour. The frequency of use calculations for this plot are identical to those described in §7.2, however frequency is now computed by occurrences of key-phrases (or MWTs).

Figure 6: *This figure shows a two stake-holder differential analysis indicating: an estimated collection of key-phrases, the frequency of usage of key phrases and sentiment attached to the usage of these phrases*



9 Future Work

The project described in this report represents a first-effort (in the JDSC) to apply text analysis to the task of analysing text-based opinions elicited through decision support workshops. While the results reported here are in this sense preliminary, the outcomes of this work have raised several extending questions.

9.1 Kullback-Leibler Inter-Topic Distances

In the current implementation of topic modelling algorithms described in this report all topic centres are placed "equidistant" on a simplex of a given dimension. In some sense this display is a 2D depiction of estimated probability distributions in a space of probability distributions. Technically the distances between these topic centres (on the display) are a type of Euclidean distance, but this is not relevant in a space of probability distributions. Furthermore spacing topics as "equidistant" points on a simplex gives no indication of how close these topic centres are to each other. In a space of probability distributions distance is usually measured by schemes such as the Kulback-Leibler (KL) distance. As an example, recall the idealised probability distributions in Table 6. The LDA scheme will estimate these distributions and there is no good reason to assume they might be equidistant. A potential future investigation on this subject might be to explore the development of a proportional display where distributions are spaced according to a distance such as the symmetric KL distance.

9.2 Differential Analyses for Text Subsets

The work in this Technical Report concerning the interesting topic of differential analyses is in its infancy. Indeed the results shown here are at best preliminary. This is an exiting and challenging area to further explore in a defence science context. The literature has a diverse collection of measures for differential analyses that could be implemented, tested and further developed for defence application. An additional challenge here, as in LDA, is the task of visualisation. Clearly the 2D visualisation task is easy for two stake-holders. However, this is a simple case and in JDSC decision support workshops there will be more than two stake-holder groups.

9.3 Algorithm Property Analyses

The emphasis in this report is upon the development of sets of algorithms to perform various tasks on a corpus of text data. While such outcomes hopefully provide value to Defence analyses, they are by no means complete. What remains is to analyse these algorithms for performance, either empirically or analytically. It would seem that computing analytical results for performance of topic estimation is difficult, but not insurmountable. However, there are numerous empirical tests that can be conceived to develop further intuition for these algorithms and hopefully identify limits and ambiguities.

9.4 Statistical Significance Filtering

The current software developed for this project displays every document (for example every text response) in the corpus of text being examined. In particular, for a JDSC workshop, this means each individuals response is allocated a set of probabilities estimating the response's relevance to a finite number of topics. In some cases this situation might result in too much data and consequently the plots on the topic simplex are cluttered. Ideally we would like a means of carefully filtering/classifying data, based on notions such as the statistical significance as text response, for example, those responses within one standard deviation of a mean etc. For example could one, (via properly defined statistics), reduce the display on the topic simplex to all those text responses within a specified probabilistic range? Alternatively, elicited text data may contain outliers which measured as statistically insignificant, might contain relevant information. In this case one wishes to display those elicited text responses that are, for example, 3 standard deviations from the mean *etc.*

9.5 Topic Modelling Visualisation Schemes

The outputs of probabilistic topic modelling are shown, for example, in Figure 4. Is this the best display ? Higher dimensional data can be depicted in a variety of ways, for example see the excellent text [KC06]. One possible project for the work reported here might be to explore alternative displays of topic estimation. In particular parallel coordinate plots. These plots are especially suitable to our task as the range space of probabilities is always the compact set $[0, 1]$. Arguably a defence analyst examining higher dimensional data computed by LDA would benefit from having a variety of higher dimensional data displays.

9.6 The Analysis of Historical Data

The JDSC has a relatively large repository of workshop text data elicited through networked text collection software. Moreover, each of these workshop data sets have associated written reports. These reports (in most cases) include human analysis of the collected text, the results of which appear in the reports in various forms. Such data will be examined retrospectively with LDA schemes (and the other forms of analysis described in this report) and the results compared against the outcomes of human analysis. Further, there are numerous corpora in Defence which could be similarly analysed, for example, the sequence of Australian Defence White Papers, or the collections of so-called Issues Papers on Force Structure Review.

10 Conclusion

10.1 Overview

In this Technical Report we have described the evolution of research collaboration between DSTO and NICTA, the aims of which were to develop an algorithmic text analysis capability. This work was motivated by a JDSC's need to provide analysis on military SME opinions elicited in text format. The JDSC's *modus operandi* and means of text data collection through networked software has also been described. In a rough way of speaking, one can liken the JDSC workshop text as a type of incomplete book which must be read and analysed, but has no contents page or index. LDA attempts to estimate "contents" in the form of a topic map, and the estimated location (index) of topic-specific material is given in terms of probabilities on a simplex. The second part of our text analysis work concerned sentiment analysis and key phrase analysis. Examples were shown on an unclassified data set collected in a real JDSC workshop.
The work described in this report is clearly in its infancy, however, it is hoped that this contribution might be continued and extended to further develop this capability.

10.2 Summary of Contributions

The main contribution of this work was to bring to bear some modern algorithms of text analysis on a real and ongoing problem in defence, *viz.*, the analysis of SME expert opinions concerning a given topic in defence capability. Probabilistic topic modelling via LDA is now well understood and has been applied in a variety of settings. In our context LDA offered a first exploratory means to estimate a topic map. LDA is one of many topic estimation schemes and LDA itself can be implemented in different ways, for example, Gibbs Sampling approximations or EM algorithms. The true value of LDA in our specific settings is yet to be determined and will form a significant part of our future work. Our main contributions from this work are:

1. the development of a single integrated software package for the text analysis techniques described in this report,

2. the development and implementation of a differential analysis capability using sentiment scores,

3. the implementation of a multi-word term identifier which scores such "phrases" in terms of sentiment,

4. the development of a diverse collection of schemes for the visualisation of text collected from stake holder groups writing a common topic,

5. a computer-based capability for searching specific corpora of decision support defence workshops.

11 Acknowledgments

First and foremost, the authors would like to acknowledge the generous funding support of the Defence Capability Group, without which this project would not have been possible. Similarly, NICTA is funded by the Australian Federal Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

The authors would also like to gratefully acknowledge the continued support and creative contributions of the NICTA research staff throughout this project, namely Dr. Scott Sanner and Mr. William Han. Further, we gratefully acknowledge the support of the JDSC Team, for developing & running the NICTA/DSTO Text Collection Workshop, in particular, Ms. Dawn Hayter (Human Scientist), Mr. Peter Pong (Defence Analyst) and Mr. Eddie Shaw (Defence Analyst).
Finally, W. P. Malcolm would especially like to thank Dr. Tim McKay for his continual support and continual encouragement.

References

ABGK93. P. Andersen, O. Borgon, R. D. Gill, and N. Keiding, *Statistical models based on counting processes*, Springer Texts in Statistics, Springer Verlag, 1993.

AD74. J. H. Ahrens and U. Dieter, *Computer methods for sampling from gamma, beta, poisson and binomial distributions*, Computing **12** (1974), 223–246.

AE04. L. Aggoun and R. J. Elliott, *Measure theory and filtering*, Statistical and Probabilistic Mathematics, Cambridge University Press, 2004.

Ana88. S. Ananiadou, *Towards a methodology for automatic term recognition*, Ph.D. thesis, University of Machester, 1988.

Ant74. Charles E. Antoniak, *Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems*, The Annals of Statistics **2** (1974), no. 6, 1152–1174.

AP76. A. C. Atkinson and M. C. Pearce, *The computer generation of beta, gamma and normal random variables (with discussion)*, Journal of the Royal statistical society (1976), no. 139, 431–461.

AR08a. Rachit Arora and Balaraman Ravindran, *Latent dirichlet allocation and singular value decomposition based multi-document summarization*, ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 2008, pp. 713–718.

AR08b. _____, *Latent dirichlet allocation based multi-document summarization*, AND '08: Proceedings of the second workshop on Analytics for noisy unstructured text data, 2008, pp. 91–97.

AS70. M. Abramowitz and A. Stegun, *Handbook of mathematical functions*, Dover Publication, Inc., New York, 1970.

Asm10. S. Asmussen, *Stochastic simulation: Algorithms and analysis*, Stochastic Modelling and Applied Probability, Springer Verlag, 2010.

Ben04. A. Bensoussan, *Stochastic control of partially observable systems*, Cambridge University Press, 2004.

Ber04. M. J. Berry, *Survey of text mining, clustering, classification, and retrieval*, Springer Verlag, 2004.

BES10. Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani, *SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining*, Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10) (Nicoletta C. Chair, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, eds.), European Language Resources Association (ELRA), May 2010.

BGHM95. J. Besag, P. Green, D. Higdon, and K. Mengersen, *Bayesian computation and stochastic systems*, Statisticial Science **10** (1995), no. 1, 3–36.

Bis06. C. M. Bishop, *Pattern recognition and machine learning*, Springer Texts in Information Science and Statistics, Springer Verlag, 2006.

BJ06. W.L. Buntine and A. Jakulin, *Discrete components analysis*, Subspace, Latent Structure and Feature Selection Techniques, Springer-Verlag, 2006.

BL09. David Blei and John Lafferty, *Topic models*, 2009.

BNJ03. D.M. Blei, A.Y. Ng, and M.I. Jordan, *Latent Dirichlet allocation*, Journal of Machine Learning Research **3** (2003), 993–1022.

Bou92. D. Bourigault, *Surface grammatical analysis for the extraction of terminological noun phrases*, Proceedings of the 14th Inernational Conference on Computational Linguistics, 1992, pp. 977–981.

Bra89. N. Bratchell, *Cluster analysis*, Chememeotrics and Intelligent Laboratory Systems **6** (1989), 105–125.

Bre99. P. Bremaud, *Markov chains*, series in applied mathematics, no. 31, Springer Verlag, 1999.

BS94. Joe M. Bernardo and Adrian F. M. Smith, *Bayesian Theory*, Wiley, 1994.

BSB08. István Bíró, Jácint Szabó, and András A. Benczúr, *Latent dirichlet allocation in web spam filtering*, Proceedings of the 4th international workshop on Adversarial information retrieval on the web, AIRWeb '08, 2008, pp. 29–32.

Bun94. Wray Buntine, *Operations for learning with graphical models*, Journal of Artificial Intelligence Research 2 **96** (1994), 159–225.

Bun02. Wray L. Buntine, *Variational Extensions to EM and Multinomial PCA*, ECML '02: Proceedings of the 13th European Conference on Machine Learning, 2002, pp. 23–34.

Bun09. Wray Buntine, *Estimating likelihoods for topic models*, The first Asian Conference on Machine Learning, 2009, pp. 51–64.

CC96. M.K. Cowles and B.P. Carlin, *Markov chain Monte Carlo convergence diagnostics: A comparative review*, Journal of the American Statistical Association **91** (1996), 883–904.

CDLS99. R. Cowell, A. Dawid, S. Lauritzen, and D. Spiegelhalter, *Probabilistic networks and expert systems*, Springer Verlag, 1999.

CFF07. L. Cao and L. Fei-Fei, *Spatially coherent latent topic model for concurrent object segmentation and classification.*, Proceedings of IEEE Intern. Conf. in Computer Vision (ICCV)., 2007.

CG92. G. Casella and E. I. George, *Explaining the gibbs sampler*, The American Statistician **46** (1992), no. 3, 167–174.

CG95. S. Chib and E. Greenberg, *Understanding the metropolis-hastings algorithm*, The American Statistician **49** (1995), no. 4, 327–335.

CLR01. G. Casella, M. Lavine, and C. P. Robert, *Explaining the perfect sampler*, The American Statistician **55** (2001), no. 4, 299–305.

CM05. Courtney Corley and Rada Mihalcea, *Measuring the semantic similarity of texts*, Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, EMSEE '05, Association for Computational Linguistics, 2005, pp. 13–18.

CMR05. O. Cappe, E. Moulines, and T. Ryden, *Inference in Hidden Markov Models*, Series in Statistics, Springer-Verlag, 2005.

CR96. George Casella and Christain P. Robert, *Rao-Blackwellisation of sampling schemes*, Biometrika **83** (1996), 81–94.

CT78. Y. S. Chow and H. Teicher, *Probability theory, independence, interchangeability, martingales*, 3 ed., Springer Texts in Statistics, Springer Verlag, 1978.

DC95. I. Dagan and K. Church, *Termight: Identifying and translating technical terminology*, Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, 1995, pp. 34–40.

DDF$^+$90. Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman, *Indexing by latent semantic analysis*, Journal of the American Society for Information Science **41** (1990), 391–407.

DeG04. M. H. DeGroot, *Optical statistical decisions*, Wiley Classics Library, Wiley, 2004.

Dev86. L. Devroye, *Non-uniform random variate generation*, Springer Verlag, 1986.

DGL94. B. Daille, E. Gaussier, and J. Langé, *Towards automatic extraction of monolingual and bilingual terminology*, Proceedings of the 15th International Conference on Computational Linguistics, 1994, pp. 515–521.

Dir39. J. P. G. L. Dirichlet, *Sue une nouvelle méthod pour la détermination des intégrales muliples*, Liouville, J. de Mathematiqués (1839), no. 4, 164–168.

DPL94. I. Dagan, F. Pereira, and L. Lee, *Similarity based estimation of word occurence probabilities*, Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, 1994, pp. 272–278.

Dun93. T. Dunning, *Accurate methods for the statistics of surprise*, Computational Linguistics **19** (1993), no. 1, 61–74.

Dur04. D. Durbin, *the most influential paper gerard salton never wrote*, Library Trends **52** (2004), no. 4, 748–764.

Dur11. R. Durrett, *Uncertainty analysis with higher dimensional dependence modelling*, Series in Probability and Statistics, Cambridge University Press, 2011.

Edw00. D. Edwards, *Introduction to graphical modelling*, 2nd ed., Springer Texts in Statistics, Springer Verlag, 2000.

ES06. Andrea Esuli and Fabrizio Sebastiani, *Sentiwordnet: A publicly available lexical resource for opinion mining*, In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC06, 2006, pp. 417–422.

FAM98. K. Frantzi, S. Ananiadou, and H. Mima, *Automatic recognition of multi-word terms: the c-value/nc-value method*, Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries, ECDL '98, Springer-Verlag, 1998, pp. 585–604.

FBY92. W. B. Frakes and R. Baeza-Yates, *Information retrieval, data structures and algorithms*, Prentice Hall, 1992.

Fer73. T. S. Ferguson, *A Bayesian analysis of some nonparametric problems*, The Annals of Statistics **1** (1973), no. 2, 209–230.

FGK+05. Patrick Flaherty, Guri Giaever, Jochen Kumm, Michael I. Jordan, and Adam P. Arkin, *A latent variable model for chemogenomic profiling*, Bioinformatics **21** (2005), no. 15, 3286–3293.

FKG10. Bela A. Frigyik, Amol Kapila, and Maya R. Gupta, *Introduction to the Dirichlet distribution and related processes*, Tech. report, 10 2010.

FNR03. Jürgen Franke, Gholamreza Nakhaeizadeh, and Ingrid Renz (eds.), *Text mining, theoretical aspects and applications*, Physica-Verlag, 2003.

Fra98. K. T. Frantzi, *Automatic recognition of multiword terms*, Ph.D. thesis, Deptatment of Computing and Mathematics Manchester Metropolitan University, 1998.

Fre85. S. French, *Group consensus probability distributions: a critical survey (with discussions)*, Bayesian Statistics (J. M. Bernado, M. H. DeGroot, and A. F. M. Smith, eds.), vol. 2, North Holland, 1985, pp. 183–201.

Gam97. D. Gamerman, *Markov chain monte carlo: Stochastic simulation for bayesian inference*, Chapman Hall, 1997.

Gey92. C. J. Geyer, *Practical markov chain monte carlo*, Statisticial Science **7** (1992), no. 4, 473–511.

GG84. S. Geman and D. Geman, *Stochastic relaxation, gibbs distributions, and the bayesian restoration of images*, IEEE Transactions on pattern analysis and machine intelligence (1984), no. 6, 84–88.

GG90. S. Geman and D. Geman, *Stochastic relaxation, gibbs distributions, and the bayesian restoration of images*, pp. 452–472, 1990.

GK03a. Mark Girolami and Ata Kabán, *On an equivalence between PLSI and LDA*, SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, 2003, pp. 433–434.

GK03b. Mark Girolami and Ata Kabán, *On an equivalence between plsi and lda*, Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, 2003, pp. 433–434.

GR00. I. S. Grandshteyn and I. M. Ryzhik, *Tables of integrals series and products*, sixth ed., Academic Press, 2000.

GR01. R. D. Gupta and D. P. Richards, *The history of the dirichlet and liouville distributions*, International statistical Review **63** (2001), 433–446.

GRS96. W. R. Gilks, Sylvia Richardson, and D. J. Spiegelhalter, *Markov chain monte carlo in practice*, Chapman Hall, 1996.

GS90. A. E. Gelfand and A. F. M. Smith, *Sampling-based approaches to calculating marginal densities*, Journal of the American statistical Association **85** (1990), 398–409.

GS04. T. L. Griffiths and M. Steyvers, *Finding scientific topics.*, Proc Natl Acad Sci USA **101 Suppl 1** (2004), 5228–5235.

GST07. T. L. Griffith, M. Steyvers, and J. B. Tenebaum, *Topics in semantic representation*, Psychological Review **114** (2007), no. 2, 221–244.

HÖ4. O. Häggström, *Fine markov chains and algorithmic applications*, Student Texts, no. 52, London Mathematical Society, 2004.

HC70. R. V. Hogg and A. T. Craig, *Introduction to mathematical satistics*, 4th ed., MacMillan Publishing Inc., 1970.

Hei08. Gregor Heinrich, *Parameter Estimation for Text Analysis*, Tech. report, University of Leipzig, 2008.

Hof99. Thomas Hofmann, *Probabilistic latent semantic indexing*, Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999, pp. 50–57.

Hof01. ———, *Unsupervised learning by probabilistic latent semantic analysis*, Mach. Learn. **42** (2001), no. 1-2, 177–196.

HS01. C. C. Heyde and E. Seneta, *Statisticians of the centuries*, Springer Verlag, 2001.

HZ08. Xuming He and Richard S. Zemel, *Latent topic random fields: Learning using a taxonomy of labels*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008., 2008, pp. 1–8.

IG01. A. Irle and J. Gani, *The detection of words and an orderring for markov chains*, Applied Probability (2001), no. 37A, 66–77.

JT06. R. L. Jones and E. Tschirner, *A frequency dictionary of german*, Routledge, Taylor and Francis Group, 2006.

Kal60. R. E. Kalman, *A new approach to linear filtering and prediction problems*, Transactions of the AMSE, Journal of Basic Engineering **82** (1960), 35–45.

KC06. D. Kurowicka and R. Cooke, *Uncertainty analysis with higher dimensional dependence modelling*, Series in Probability and Statistics, John Wiley & Sons Ltd., 2006.

Ken09. Maurice Kendall, *Advanced theory of statistics*, 6th ed., vol. 1, Wiley, 2009.

KF09. D. Koller and Nir Friedman, *Probabilistic graphical models*, MIT Press, 2009.

Kow97. G. Kowalski, *Information retrieval systesm, theory and implementation*, Kluwer Academic Press, 1997.

KU96. K. Kageura and B. Umino, *Methods of automatic term recognition: a review*, Terminology **3** (1996), no. 2, 259–289.

Lau96a. A. Lauriston, *Automatic term recognition: performance of linguistic and statistical techniques*, Ph.D. thesis, University of Machester, 1996.

Lau96b. S. L. Lauritzen, *Graphical models*, Oxford Science Publications, no. 17, Oxford University Press, 1996.

LK58. E. L. and P. Meier Kaplan, *Nonparametric estimation from incomplete observations*, Journal of the American statistical Association **53** (1958), 457–481.

LP05. Fei-Fei Li and Pietro Perona, *A Bayesian hierarchical model for learning natural scene categories*, CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2, 2005, pp. 524–531.

LPW05. M. Lee, B. Pinicombe, and M. Welsh, *An empirical evaluation of models of text document similarity*, Proceedings of the XXVII Annual Conference of the Cognitive Science Society, 2005, pp. 1245–1259.

LS99. D. Lee and H. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature **401** (1999), 788–791.

Mak71. Armand M. Makowski, *Nonlinear bayesian estimation using gaussian sum approximations*, IEEE Transactions on Automatic Control **17** (1971), 439–448.

Mak86. ———, *Filtering formulae for partially observed linear systems with non-gaussian initial conditions*, Stochastics **16** (1986), no. 1-2, 1–24.

MCZZ08. Qiaozhu Mei, Deng Cai, Duo Zhang, and ChengXiang Zhai, *Topic modeling with network regularization*, Proceeding of the 17th international conference on World Wide Web, 2008, pp. 101–110.

Mit04. R. Mitkov, *The oxford handbook of computational linguistics*, Oxford University Press, 2004.

MKE05. Rasmus E. Madsen, David Kauchak, and Charles Elkan, *Modeling word burstiness using the dirichlet distribution*, Proceedings of the 22nd international conference on Machine learning (New York, NY, USA), ICML '05, ACM, 2005, pp. 545–552.

ML02. Thomas Minka and John Lafferty, *Expectation-propogation for the generative aspect model*, Proceedings of the Proceedings of the Eighteenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-02), 2002, pp. 352–359.

MP. G. McLachlan and D. Peel, *Finite mixture models*, Probability and Statistics, Wiley-Interscience.

MS99. C. D. Manning and H. Schütze, *Foundations of natural language processing*, The MIT Press, 1999.

MT93. S. P. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*, Springer-Verlag, London, 1993.

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

Nar90. A. Narayanan, *Computer generation of dirichlet random vectors*, Journal of Statistical and Computer Simulation **36** (1990), 19–30.

Nea00. Radford M. Neal, *Markov chain sampling methods for Dirichlet process mixture models*, Journal of Computational and Graphical Statistics **9** (2000), no. 2, 249–265.

Nor98. J. R. Norris, *Markov chains*, Statistical and probabilistic mathematics, Cambridge University Press, 1998.

PAF+07. K. N. Papamichail, G Alves, S. French, J. B. Yang, and R. Snowdon, *Facilitation practices in decision workshops*, Journal of the Operations Research Society **58** (2007), 614–632.

PG11. Marco Pennacchiotti and Siva Gurumurthy, *Investigating topic models for social media user recommendation*, Proceedings of the 20th international conference companion on World wide web, 2011, pp. 101–102.

Pid96a. M. Pidd, *System modelling: Theory and practice*, John Wiley and Sons, 1996.

Pid96b. _____, *Tools for thinking, modelling in management science*, John Wiley and Sons, 1996.

Pin04. B. Pinicombe, *Comparison of human and latent semantic analysis (lsa) judgements of pairwise document similarities for a news corpus*, Tech. report, Defence Science and Technology Organisation, 2004.

PSD00. J.K. Pritchard, M. Stephens, and P.J. Donnelly, *Inference of population structure using multilocus genotype data*, Genetics **155** (2000), 945–959.

RC05. Christian P. Robert and George Casella, *Monte Carlo statistical methods (springer texts in statistics)*, 2005.

Rob01. C. P. Robert, *The bayesian choice*, second ed., Springer Series in Statistics, Springer Verlag, 2001.

Ros96. S. M. Ross, *Stochastic processes*, second ed., probability and mathematical statistics, Wiley, 1996.

RS00. H. Raiffa and R Schlaifer, *Applied statistical deicision theory*, second ed., Wiley Classics Library, John Wiley and Sons, Inc., 2000.

RS01. V. K. Rohatgi and A. K. Md. Ehsanes Saleh, *An introduction to probability and statistics*, Probability and Statistics, John Wiley and Sons, Inc., 2001.

SBF+11. V. C. Sousa, M. A. Beaumont, P. Fernandes, M. M. Coelho, and L. Chikhi, *Population divergence with or without admixture: selecting models using an abc approach*, Heredity **108** (2011), 521–530.

SDK11. B. Stone, S. Dennis, and P. Kwantes, *Comparing methods for single paragraph similarity analysis*, Topics in Cognitive Science **3** (2011), 92–122.

Set94. J. Sethuraman, *A constructive definition of Dirichlet priors*, Statistica Sinica **4** (1994), 639–650.

SG92. A. F. M. Smith and A. E. Gelfand, *Bayesian statistics without tears*, The American Statistician **46** (1992), no. 2, 84–88.

SG07. Mark Steyvers and Tom Griffiths, *Probabilistic Topic Models*, 2007.

SM86. Gerard Salton and Michael J. McGill, *Introduction to modern information retrieval*, McGraw-Hill, Inc., New York, NY, USA, 1986.

SW11. Robert Sedgewisck and Kevin Wayne, *Algorithms*, 4th ed., Addison-Wesley, 2011.

SWY75. G. Salton, A. Wong, and C. S. Yang, *A vector space model for automatic indexing*, Communications of the ACM **18** (1975), no. 11, 613–617.

TL03. Peter D. Turney and Michael L. Littman, *Measuring praise and criticism: Inference of semantic orientation from association*, ACM Transactions on Information Systems **21** (2003), 315–346.

TNW07. Yee Whye Teh, David Newman, and Max Welling, *A collapsed variational bayesian inference algorithm for latent dirichlet allocation*, Advances in Neural Information Processing Systems 19, 2007, pp. 1353–1360.

TP09. G. Tsatsaronis and V. Pamagiotopoulou, *A generalised vector-space model for text retrieval based on semantic relatedness*, Proceedings of the EACL, 2009, pp. 70–78.

TVV10. G. Tsatsaronis, I. Varlamis, and M. Vazirgiannis, *Text relatedness based on a word thesaurus*, Artificial Intelligence Research **7** (2010), 1–39.

WBFF09. Chong Wang, David Blei, and Li Fei-Fei, *Simultaneous image classification and annotation*, CVPR, 2009.

WC06. Xing Wei and W. Bruce Croft, *LDA-based document models for ad-hoc retrieval*, SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, 2006, pp. 178–185.

WG08. Xiaogang Wang and Eric Grimson, *Spatial latent Dirichlet allocation*, Advances in Neural Information Processing Systems 20, 2008, pp. 1577–1584.

Wil62. S. S. Wilks, *Mathematical statistics*, John Wiley and Sons, Inc., New York, 1962.

WMM09. H. Wallach, D. Mimno, and A. McCallum, *Rethinking LDA: Why priors matter*, Advances in Neural Information Processing Systems 19, 2009.

Appendix A Conjugate Priors

A.1 Definitions

The notion of conjugate priors is an inherently Bayesian concept and was introduced relatively recently in 1961 by Raiffa and Schlaifer. A more recent edition of this work may be found in [RS00]. A key idea of conjugate priors is (loosely) based on the property *closure* with respect to set membership.

Definition A.1 (Conjugate Prior). *Suppose $\mathcal{F}$ is a family of probability distributions with parameter space Θ. The family $\mathcal{F}$ is said to be conjugate for a likelihood function $\ell(x \mid \theta) \triangleq p(x \mid \theta)$, if for every $\pi(\cdot) \in \mathcal{F}$, then $\pi(\theta \mid x) \in \mathcal{F}$.*

Recall that the normalised version of Bayes' rule is usually written

$$\pi(\theta \mid x) = \frac{\pi(\theta) f(x \mid \theta)}{\int_{\Theta} \pi(\xi) f(x \mid \xi) d\xi}. \tag{A1}$$

The problem-child in equation (A1) is its denominator on the right hand side, which usually requires some form numerical integration when the integrand is complex in form, or multidimensional, or both. However, the integrand in this term is identical in form to the numerator and so Bayes' rule is more commonly written in its proportionality form,

$$\pi(\theta \mid x) \propto \pi(\theta) f(x \mid \theta). \tag{A2}$$

The importance/value of conjugate priors is essentially based on computational issues. If the collection $\mathcal{F}$ is parametrised, then using conjugate priors will mean that updating/switching from a prior to a posterior distribution is just a matter of updating a finite set of parameters. Moreover, the possibility of numerical integrations may be avoided by using conjugate priors.

A.2 Example

Remark A.1 (Example). *The celebrated Kalman filter introduced by Rudolf Kalman (see [Kal60]), offers a classic illustration of the computational value when using conjugate priors. Loosely speaking, the discrete-time Kalman filter estimates a Gaussian distribution (or a function) at each discrete point in time. Fortunately, a Gaussian distribution is fully described by its two sufficient statistics, that is, a mean and a covariance/variance. If a Gaussian prior is used, as was the case in Kalman's original work, then the posterior distribution is also Gaussian, since Gaussians are closed under multiplication. Therefore Kalman's algorithm need only update the sufficient statistics just mentioned. As an aside, the stochastic models studied by Kalman can quickly become computationally complex if non-Gaussian priors are used. Kalman-like estimation schemes for non-Gaussian initial conditions have been studied in [Ben04] and [Mak86].*

Table A1: *Posterior distribution parameter updates for a Beta prior and binomial likelihood.*

Statistic	prior to observing x	posterior to observing x
mean	$\frac{a}{a+b}$	$\frac{a+x}{a+b+n}$
variance	$\frac{ab}{(a+b)^2(a+b+1)}$	$\frac{(a+x)(b+n-x)}{(a+b+n)^2(a+b+n+1)}$

Remark A.2 (Example). *To give an explicit example of Bayesian estimation with conjugate priors, we consider an experiment whose outcomes are distributed according to a binomial probability distribution. Suppose* $X \sim Bin(n, \theta)$ *where* n *is known and* $\theta \in \Theta$ *is an unknown parameter. We suppose that* θ *is in the interval* $(0, 1)$ *and is distributed according to a Beta distribution (see Appendix C), so that*

$$\pi(\theta) = \frac{\theta^{a-1}(1-\theta)^{b-1}}{B(a,b)}. \tag{A3}$$

Consequently the form of $\pi(\theta \mid x)$ *in the above example is again a Beta distribution with parameter updates/recursions listed in Table A1*

Remark A.3. *One well known example of a probability density that does not have a conjugate prior is the uniform probability density.*

The interested reader can find far more detail on conjugate priors in the well known classic works: [Rob01, RS00, DeG04, BS94].

Appendix B Multinomial Probability Distributions

B.1 Background

In this Appendix we recall some basic elements of multinomial distributions. Technical details on these distributions may be found in [Wil62] and [RS01].
In most introductory courses on probability theory one inevitably meets the binomial distribution, which is a probability distribution for a finite sequence of independent experiments, each of which has a two-state outcome, for example $\{0, 1\}$, $\{\mathrm{H}, \mathrm{T}\}$, or $\{\text{Success}, \text{Fail}\}$ *etc.* The very next extension to this situation is a trinomial distribution, modelling experiments with three possible outcomes. Further, multinomial distributions model a more general version of the situations just described where each elementary experiments results in one and one only of $r > 2$ possible outcomes. In particular, the multinomial distribution describes the probabilities of compound events, each consisting of N basic experiments, (elementary events), each of which has r possible outcomes.

For brevity we denote the set of elementary event outcomes by $\mathcal{S} \triangleq \{1, 2, \ldots, r\}$, that is, we consider an integer-valued random variable X with the mapping

$$X : \Omega \to \mathcal{S}. \tag{B1}$$

Consider a compound experiment consisting of N independent repetitions of the the RV X. A specific realisation or outcome of this compound experiment is labelled as

$$\omega = \big\{i_1, i_2, \ldots, i_N\big\}. \tag{B2}$$

Here $i_\ell \in \mathcal{S}$, for $\ell \in \{1, 2, \ldots, N\}$ Note that collection at (B2) is an ordered N-tuple. the probabilities assigned to each ω are

$$\begin{aligned} p(\omega) &= p\big(\omega \mid X_1(\omega) = i_1, X_2(\omega) = i_2, \ldots, X_N(\omega) = i_N\big), \\ &= p_{i_1} \times p_{i_2} \times \cdots \times p_{i_N}. \end{aligned} \tag{B3}$$

What we would like to do is write down the probability distribution for any given event $\omega \in \Omega$. This task is straightforward needing only some simple formulae from combinatorics.

B.2 Derivation

Definition B.1 (Count Frequencies). *Suppose* $\big\{n_1, n_2, \ldots, n_r\big\}$ *natural numbers whose values indicate the number of specific elementary events occurring in a compound event consisting of* N *trials. Clearly, the* n_ℓ *are linearly dependent, so*

$$N = \sum_{\ell=1}^{r} n_\ell. \tag{B4}$$

The numbers n_ℓ *are not to be confused with the values* X *might take from the set* $\mathcal{S}$*, rather, the numbers* n_ℓ *are counts for the occurrence of the* $X(\omega) = \ell$ *in* N *trials.*

Using these count frequencies we can write down the probability for a compound event having n_1 occurrences of the outcome 1, n_2 occurrences of the outcome 2, and similarly up to n_r occurrences of the outcome r. Due to independence this quantity is $p_1^{n_1} \times p_2^{n_2} \times \cdots p_r^{n_r}$. However, numerous outcomes have exactly this probability. For example, suppose we consider a trivial case of $r = 2$ and $N = 3$, then the collection of all possible outcomes is $\{1, 1, 2\}$, $\{1, 2, 1\}$ and $\{2, 1, 1\}$, each having the identical probability $p_1^2 \times p_2^1$. We recall the following result from combinatorics,

Lemma B.1. *The number of unordered samples of size r from a population of size N is,*

$$C(N, r) \triangleq \binom{N}{r} = \frac{N(N-1)(N-2)\cdots(N-r+1)}{r!}. \tag{B5}$$

Further, for $1 \leq k \leq r$ we write $Z_k(\omega)$ for a random variable whose integer value indicates the number of times $k \in \{1, 2, \ldots, r\}$ appeared in N independent repetitions of our basic experiment. Clearly, $\forall\, \omega \in \Omega$

$$\sum_{\ell=1}^{r} Z_\ell(\omega) = N. \tag{B6}$$

Now we can write down the probability for our compound event of N trials which is,

$$\begin{aligned} P\big(Z_1(\omega) = n_1, \ldots, Z_r(\omega) = n_r\big) &= C(N, n_1)C(N, n_2)\cdots C(N, n_r) \times p_1^{n_1} \times \cdots \times p_r^{n_r} \\ &= \binom{N}{n_1}\binom{N-n_1}{n_2}\binom{N-n_1-n_2-\cdots n_{r-1}}{n_r} \\ &\qquad \times p_1^{n_1} \times \cdots \times p_r^{n_r}. \end{aligned} \tag{B7}$$

Consequently multinomial probability density for the joint events $\{Z_1, Z_2, \ldots, Z_r\}$ has the form,

$$\begin{aligned} p(Z_1 = n_1, \ldots, Z_r = n_r) &= \frac{N!}{n_1! n_2! \cdots n_r!} p_1^{n_1} \times \cdots \times p_r^{n_r} \\ &= \left(\frac{N!}{\prod_{\ell=1}^{r} n_\ell!}\right) \prod_{j=1}^{r} p_j^{n_j}. \end{aligned} \tag{B8}$$

The Moment generating function for a Multinomial distribution has the form,

$$\mathrm{MGF}(t_1, t_2, \ldots, t_r) = \Big(p_1 \exp(t_1) + p_2 \exp(t_2) + \cdots + p_r \exp(t_r)\Big)^N. \tag{B9}$$

Here $t_1, t_2, \ldots, t_r \in \mathbb{R}$.

Remark B.1. *For the simple case $r = 2$ the density at (B8) reduces to the binomial probability density. This is easily seen from the MGF by noting that*

$$\begin{aligned} \mathit{MGF}(t_1, 0, 0, \ldots, 0) &= \Big(p_1 \exp(t_1) + p_2 + \cdots + p_r\Big)^N \\ &= \Big(1 - p_1 + p_1 \exp(t_1)\Big)^N, \end{aligned} \tag{B10}$$

which is the MGF for a binomial distribution and Moment generating functions are unique.

B.3 Some Statistics

The following basic statistics are of use, $E[n_i] = Np_i$, $\mathrm{Var}[n_i] = Np_i(1-p_i)$ and $\mathrm{Cov}[n_i, n_j] = -Np_ip_j$, $i \neq j$. More technical details on multinomial distributions can be found in [Wil62, RS01]. The multinomial distribution has also been studied through change of probability-measure techniques. The interested reader is referred to the excellent monograph [AE04].

Appendix C The Beta Distribution

C.1 Basic Properties

The Beta probability distribution belongs to a family of distributions whose forms usually involves Gamma functions. Recall, that the Gamma function may be written

$$\Gamma(x) = \int_0^\infty t^{x-1} \exp(-t) dt. \tag{C1}$$

Here we take $x \in \mathbb{R}_+$. More details on this function and complex valued arguments can be found in [AS70, GR00]. Some useful formulae for this function are;

$$\Gamma(x) = (x-1)!, \qquad x \in \mathbb{N} \tag{C2}$$

$$\Gamma(x) = (x-1)\Gamma(x-1), \qquad x \in \mathbb{R}_+, \tag{C3}$$

$$\Gamma(1/2) = \sqrt{\pi}, \tag{C4}$$

$$\Gamma(2x) = \frac{2^{2x-1}}{\sqrt{\pi}} \Gamma(x)\Gamma(x+\tfrac{1}{2}). \tag{C5}$$

Given the connection of the Beta distribution to the Dirichlet distribution and its correspondence to the Dirichlet, (a Dirichlet distribution for dimension 2 collapses to a Beta distribution), we recall some basic properties of the Beta distribution here.
The term Beta distribution refers to a family of probability distributions defined on the set $(0, 1)$ indexed by two scalar-valued parameters, α and β.

Definition C.1 (Beta). *The probability density for the Beta distribution has the form,*

$$f(x \mid \Theta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} \big(1-x\big)^{\beta-1}. \tag{C6}$$

Here $\Theta = \{\alpha, \beta\}$ *and the bounds on variables are* $0 < x < 1$, $\alpha > 0$ *and* $\beta > 0$ *and* $B(\alpha, \beta)$ *denotes the so-called Beta function,*

$$B(\alpha, \beta) = \int_{0,1} \xi^{\alpha-1} \big(1-\xi\big)^{\beta-1} d\xi. \tag{C7}$$

The Beta function above is directly related to the Gamma function through the following identity,

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} \tag{C8}$$

Conveniently all moments of the Beta distribution can be calculated without evaluating the integrals in either (C6) or (C7). With $n > -\alpha$ we see that

$$\begin{aligned} E\big[X^n\big] &= \frac{1}{B(\alpha, \beta)} \int_0^1 \xi^n \xi^{\alpha-1} \big(1-\xi\big)^{\beta-1} d\xi \\ &= \frac{1}{B(\alpha, \beta)} \int_0^1 \xi^{\big(\alpha+n\big)-1} (1-\xi)^{\beta-1} d\xi \\ &= \frac{B(\alpha+n, \beta)}{B(\alpha, \beta)} \\ &= \frac{\Gamma(\alpha+n)\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+n)\Gamma(\alpha)}. \end{aligned} \tag{C9}$$

Consequently the mean and variance for a Beta distributed random variable are, respectively,

$$E[X] = \frac{\alpha}{\alpha+\beta}, \tag{C10}$$

$$E\big[\big(X - E[X]\big)^2\big] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \tag{C11}$$

Remark C.1. *The Beta distribution provides a significant modelling flexibility for scalar-valued random variables defined on the unit interval, that is, by varying the values of* α *and* β *one can generate a large variety of shape features in Beta distributions. For example, if* $(\alpha > 1, \beta = 1)$ *then the family of curves is strictly increasing, or strictly decreasing if* $(\alpha = 1, \beta > 1)$*. If* $\alpha = \beta$ *symmetric densities are produced. If* $\alpha = \beta = 1$ *the Beta distribution degenerates to a uniform probability distribution.*

Remark C.2. *Beta distribution is a conjugate prior for a binomial distribution.*

C.2 Checking that $\int f(\xi)d\xi = 1$

Finally, it is an interesting exercise to show that the probability density given at (C6) is in fact a valid probability distribution. The parameter ranges given in Definition (C.1) ensure the integral is well defined. However, what remains is to show the that the Beta probability density integrates to unity.
Recall the (real valued) Gamma function defined at (C1). We first consider a product of these functions,

$$\begin{aligned}\Gamma(a)\Gamma(b) &= \int_0^\infty \xi^{(a-1)}\exp(-\xi)d\xi \int_0^\infty \lambda^{(b-1)}\exp(-\lambda)d\lambda \\ &= \int_0^\infty\int_0^\infty \xi^{(a-1)}\exp(-\xi)\lambda^{(b-1)}\exp(-\lambda)d\xi d\lambda.\end{aligned} \tag{C12}$$

Here $a, b \in \mathbb{R}_+$. This double integral may be solved through change of variable techniques, the first being

$$\xi \triangleq r^2\cos^2(\theta), \tag{C13}$$

$$\lambda \triangleq r^2\sin^2(\theta). \tag{C14}$$

The Jacobian for this transformation is

$$J \triangleq \Big|\frac{\partial(\xi,\lambda)}{\partial(r,\theta)}\Big| = 4r^3\sin(\theta)\cos(\theta). \tag{C15}$$

Applying this transformation we get,

$$\Gamma(a)\Gamma(b) = 4\int_0^{\pi/2}\int_0^\infty \big(\cos(\theta)\big)^{(2a-1)}\big(\cos(\theta)\big)^{(2b-1)} r^{(2a+2b-1)}\exp(-r^2)drd\theta. \tag{C16}$$

Now consider the radial component of this double integral and make the substitution $r = \xi^{\frac{1}{2}}$. Then

$$\begin{aligned} I &= 2\int_0^\infty r^{2a+2b-1}\exp(-r^2)dr, \\ &= 2\int_0^\infty \xi^{a+b-\frac{1}{2}}\exp(-\xi)\tfrac{1}{2}\xi^{-\frac{1}{2}}d\xi, \\ &= \int_0^\infty \xi^{(a+b-1)}\exp(-\xi)d\xi, \\ &= \Gamma(a+b). \end{aligned} \tag{C17}$$

Consequently

$$\frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} = 2\int_0^{\pi/2}\cos(\theta)^{(2a-1)}\sin(\theta)^{(2b-1)}d\theta. \tag{C18}$$

To compute this integral we make the substitution $\cos(\theta) = \sqrt{\xi}$. Then

$$\cos(\theta)^{(2a-1)} = \xi^{a-\frac{1}{2}} \tag{C19}$$

$$\sin(\theta)^{(2a-1)} = (1-\xi)^{b-\frac{1}{2}}. \tag{C20}$$

Using these substitutions we see that

$$\begin{aligned} 2\int_0^{\pi/2}\cos(\theta)^{(2a-1)}\sin(\theta)^{(2b-1)}d\theta &= -2\int_0^1 \xi^{(a-\frac{1}{2})}\big(1-\xi\big)^{b-\frac{1}{2}}\frac{d\theta}{d\xi}d\xi, \\ &= -2\int_0^1 \xi^{(a-\frac{1}{2})}\big(1-\xi\big)^{b-\frac{1}{2}} - \tfrac{1}{2}\xi^{-\frac{1}{2}}(1-\xi)^{-\frac{1}{2}}d\xi \\ &= \int_0^1 \xi^{(a-1)}\big(1-\xi\big)^{(b-1)}d\xi \\ &= \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}. \end{aligned} \tag{C21}$$

□

Appendix D The Dirichlet Probability Distribution

At first meeting, the Dirichlet distribution will seem surprising for two reasons

1. **[Form]** Most univariate probability distributions may be written in a general form $f(x;\Theta)$, where x is the independent variable and Θ denotes a set of one or more parameters. However, the Dirichlet distribution is different to such form in that its independent variable is itself a discrete probability distribution and so written (roughly) as $f\big(\{p_1, p_2, \ldots, p_K\};\Theta\big)$. Consequently the Dirichlet distribution is a distribution over finite probability distributions.

2. **[Visualisation]** The second difficulty one encounters with the Dirichlet distribution is that it's not easily visualised, especially for higher dimensions. In the case of a univariate Gaussian distribution (for example), one immediately sees from inspection a measure of central tendency (mean) and a dispersion about a mean (variance). Other basic shape features one might look for in a simple probability distribution are skew, kurtosis and mode *etc.* However, none of these shape features are easily seen with a Dirichlet distribution. At the outset, the domain for a Dirichlet distribution is a standard simplex (defined below), consequently plotting and visualising Dirichlet distributions will not be routine.

The Dirichlet distribution is named after the famous German mathematician Johann Peter Gustav Lejuene Dirichlet[14] [1805-1859]. Dirichlet's seminal paper on this topic (concerning mostly Dirichlet Integrals) appeared in 1839, see [Dir39]. A comprehensive survey of this history of the Dirichlet distribution, the Dirichlet integral and the related Liouville distribution can be found in [GR01]. Surprisingly this distribution is not well known, indeed many modern books on probability theory make no mention of it at all, with one notable example being *Mathematical Statistics* by S. S. Wilks ([Wil62]). However, the Dirichlet distribution has been widely applied in a diversity of settings such as: statistical genetics, belief functions, order statistics (see §8.7 in [Wil62]) and reliability theory, to name a few. Further detail on this distribution and its applications can be found in [Fer73, Ant74, Set94, FKG10, BS94].

The Dirichlet distributions arises quite naturally in LDA text analysis as LDA casts topic modelling and estimation as a Bayesian inference problem. Consequently, basic parameters such as the probability distribution of words in a topic and the probability distribution of topics within a document, are not interpreted as fixed an unknown, rather, as random variables, each with their own right with probability distributions. Consequently the Dirichlet distribution offers a convenient means to write down these problems with an added bonus that the Dirichlet is a conjugate prior with the multinomial distribution. This makes an already complex problem *tractable*. As an aside to this, some authors have studied variants on the Dirichlet distributions in modelling of "bursts of words" in text, see [MKE05].

[14] One indication of the reputation and contributions of Dirichlet is that he was chosen to be C. F. Gauss' successor at the University of Göttingen.

Definition D.1 (The Standard Simplex). *There exists a variety of simplices, for our purposes the so-called standard simplex is sufficient. We denotes a K dimensional standard simplex as $\mathcal{S}_K$, where*

$$\mathcal{S}_K \triangleq \left\{(p_1, p_2, \ldots, p_k) \mid p_i \in [0,1] \,\forall\, i = 1, 2, \ldots, K, \text{ and } \sum_{i=1}^{K} p_i = 1\right\} \subset \mathbb{R}^K. \tag{D1}$$

In some literature the simplex $\mathcal{S}_K$ is referred to as the "probability simplex".

Definition D.2 (Dirichlet Distribution). *Consider a finite probability distribution $\boldsymbol{p}$, where*

$$\boldsymbol{p} \triangleq \{p_1, p_2, \ldots, p_K\} \tag{D2}$$

The probability distribution $\boldsymbol{p}$ is said to have a Dirichlet distribution, if

$$\boldsymbol{p} \sim Dir_K\big(\boldsymbol{p}, \{\alpha_1, \alpha_2, \ldots, \alpha_K\}\big) = \left(\frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)}\right) \prod_{k=1}^{K} p^{\alpha_k - 1} \mathbf{1}_{\{\boldsymbol{p} \in \mathcal{S}_K\}}. \tag{D3}$$

Here the parameter $\boldsymbol{\alpha} \triangleq \{\alpha_1, \alpha_2, \ldots, \alpha_K\}$ is such that

$$\alpha_k \in \mathbb{R}_+ \,\forall\, k = 1, 2, \ldots, K \tag{D4}$$

Remark D.1. *Note that the most basic and expected parameters of mean and variance do not appear explicitly in the function given at equation D3. These parameters are computed/derived from the alpha values. Further, it is convention to use the word "concentration", rather than variance for Dirichlet probability distributions.*

Remark D.2. *If $K = 2$, then then the distribution at (D3) degenerates to the Beta distribution. In this sense the Dirichlet may be thought of as a $K > 2$ generalization of the Beta distribution.*

Remark D.3. *If all components of $\boldsymbol{\alpha}$ are set identically to unity, the Dirichlet becomes uniform in the sense that every candidate $\boldsymbol{p}$ is equally likely to occur.*

Remark D.4. *It is an interesting and non-trivial exercise to show that the Dirichlet distribution integrates to unity on the standard simplex. Details of this calculation can be found in [Wil62].*

D.1 Basic Statistics For A Dirichlet Distribution

Given that the Dirichlet distribution's independent variable is multivariate, its statistics may be given component wise. These statistics may be computed from the characteristic function (see Wilks [Wil62]) or directly using properties of Gamma functions. We show one such calculation here for the means of a Dirichlet distribution. It is sufficient to consider one component of the mean as is shown below.

$$\begin{aligned} E[p_i] &= \int_{\mathcal{S}_K} p_i \mathrm{Dir}_K(\boldsymbol{p}, \boldsymbol{\alpha}) d\boldsymbol{p} \\ &= \int_{\mathcal{S}_K} p_i \mathrm{Dir}_K(\boldsymbol{p}, \boldsymbol{\alpha}) dp_1 dp_2 \ldots dp_K \\ &= \int_{\mathcal{S}_K} p_i \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\cdots\Gamma(\alpha_K)} \prod_{k=1}^{K} p_k^{\alpha_k - 1} d\boldsymbol{p} \end{aligned} \tag{D5}$$

Here we write $\alpha_0 \triangleq \sum_{k=1}^{K} \alpha_k$. Without loss of generality we may take $p_i = p_1$, so that

$$E[p_1] = \int_{\mathcal{S}_K} \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\Gamma(\alpha_2)\cdots\Gamma(\alpha_K)} p_1^{\alpha_1} \prod_{k=2}^{K} p_k^{\alpha_k - 1} d\boldsymbol{p} \tag{D6}$$

Recalling the properties of Gamma functions for $\alpha_i \in \mathbb{R}_+$, we make the following assignments

$$\beta_1 \triangleq \alpha_1 + 1 \tag{D7}$$

$$\beta_i \triangleq \alpha_i, \, \forall \, i = 2, 2, \ldots, K. \tag{D8}$$

Consequently

$$\begin{aligned} E[p_1] &= \int_{\mathcal{S}_K} \frac{\alpha_1}{\alpha_0} \frac{\Gamma(\beta_0)}{\Gamma(\beta_1)\Gamma(\beta_2)\cdots\Gamma(\beta_K)} p_1^{\beta_1 - 1} \prod_{k=2}^{K} p_k^{\beta_k - 1} d\boldsymbol{p} \\ &= \frac{\alpha_1}{\alpha_0} \int_{\mathcal{S}_K} \mathrm{Dir}_K(\boldsymbol{p}, \boldsymbol{\beta}) d\boldsymbol{p} \\ &= \frac{\alpha_1}{\alpha_0}. \end{aligned} \tag{D9}$$

Similar calculations may be used to compute other useful statistics which we list in the table below.

Table D1: *Some basic statistics for a Dirichlet distribution*

$E[p_i]$	$\dfrac{\alpha_i}{\alpha_0}$
$E[\boldsymbol{p}]$	$\left(\dfrac{\alpha_2}{\alpha_0}, \dfrac{\alpha_2}{\alpha_0}, \cdots \dfrac{\alpha_K}{\alpha_0}\right)$
$\mathrm{Var}(p_i)$	$\dfrac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(1 + \alpha_0)}$
$\mathrm{Cov}(p_i, p_j), \; i \neq j$	$\dfrac{\alpha_i \alpha_j}{\alpha_0^2(\alpha_0 + 1)}$
$\mathrm{Mode}(\boldsymbol{p})$	$\left(\dfrac{\alpha_1 - 1}{\alpha_0 - K}, \dfrac{\alpha_2 - 1}{\alpha_0 - K}, \ldots, \dfrac{\alpha_K - 1}{\alpha_0 - K}\right)$

D.2 Conjugacy with a Multinomial Distribution

The following definition is standard in Bayesian statistics,

Definition D.3 (Exponential Families). *Suppose $\mu(\cdot)$ is a σ-finite measure defined on $\mathcal{X}$ and that Θ denotes a parameter space. Suppose that the functions $C(\cdot)$ and $f(\cdot)$ are, respectively from $\mathcal{X}$ and Θ to $\mathbb{R}_+$. Further, the functions $T(\cdot)$ and $R(\cdot)$ map from Θ and $\mathcal{X}$ to $\mathbb{R}^K$. The family of probability distributions with respect to the measure $\mu(\cdot)$*

$$f(x \mid \theta) = C(\theta) f(x) \exp\big(\big\langle R(\theta), T(x) \big\rangle\big) \tag{D10}$$

are called an exponential family of dimension K.

For more details on exponential families see [Rob01].
The Dirichlet distribution may be written as follows,

$$\mathrm{Dir}_K(\boldsymbol{p},\boldsymbol{\alpha}) = C(\boldsymbol{\alpha})f(\boldsymbol{p})\exp\Big(\sum_{k=1}^{K}\alpha_k\ln(p_k)\Big). \tag{D11}$$

Here the operator $T(\cdot)$, as it appears in equation (D10), is explicitly

$$T(\boldsymbol{p}) \triangleq \big(\ln(p_1),\ln(p_2),\ldots,\ln(p_K)\big). \tag{D12}$$

Consequently Dirichlet distributions constitute an exponential family for the operator $T(\cdot)$, as defined by equation (D12) and so have a conjugate prior due to the Pittman-Koopman Lemma (see [Rob01]).

Lemma D.1. *Suppose the probability distribution $\boldsymbol{p} = \{p_1, p_2, \ldots, p_K\}$ is a random prior in a Bayesian estimation task and is assumed to be distributed according to a Dirichlet distribution with parameter $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_K\}$. Further, suppose the set of integers $\{m_1, m_2 \ldots, m_K\}$ is distributed according to a multinomial distribution with the same probabilities $\boldsymbol{p}$ as above and is the assumed likelihood for this Bayesian estimation task. The unnormalised posterior distribution, formed through the product of these two distributions, is an unnormalised Dirichlet distribution.*

Proof of Lemma D.1

$$\begin{aligned}
\pi\big(\boldsymbol{p} \mid \{m_1,m_2,\ldots,m_K\}\big) &\propto \ell\big(\{m_1,m_2,\ldots,m_K\} \mid \boldsymbol{p}\big)\pi(\boldsymbol{p}) \\
&= \frac{N!}{m_1!m_2!\cdots m_K!}\prod_{k=1}^{K} p_k^{m_k}\left(\frac{\Gamma(\sum_{k=1}^{K}\alpha_k)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\right)\prod_{k=1}^{K} p_k^{\alpha_k-1} \\
&= \frac{N!}{m_1!m_2!\cdots m_K!}\left(\frac{\Gamma(\sum_{k=1}^{K}\alpha_k)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\right)\prod_{k=1}^{K} p_k^{(\alpha_k+m_k)-1} \\
&\propto \mathrm{Dir}_K\big(\boldsymbol{p},\{\alpha_1+m_1,\alpha_2+m_2,\ldots,\alpha_K+m_K\}\big) \qquad \square.
\end{aligned} \tag{D13}$$

D.3 Generating Dirichlet Random Variables

There are several schemes by which one might generate samples from a Dirichlet distribution and indeed several physical models which give rise to a Dirichlet distribution, for example, the Polya Urn scheme and the so-called unity-length "stick breaking" scheme, see [FKG10]. Perhaps the most popular and convenient scheme to generate samples from a Dirichlet distribution is via the simulation of independently distributed Gamma random variables, each having a common scale parameter.

Definition D.4 (The Gamma Distribution). *Gamma random variables are continuously distributed and take values in the position half of the real line $\mathbb{R}_+$, according to the probability density function*

$$\Gamma(x \mid k,\theta) = \frac{1}{\theta^k\Gamma(\theta)}x^{k-1}\exp(-x/\theta). \tag{D14}$$

Here k and θ are, respectively, the so-called scale and shape parameters. These parameters take values in the positive half of the real line.

The computer simulation of gamma random variables is discussed in the articles [AD74, AP76].

Lemma D.2. *Suppose $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \ldots, \alpha_K\}$ is a given parameter for the Dirichlet distribution. Further, suppose that for $i = 1, 2, \ldots, K$, $\gamma_i \sim \Gamma(\alpha_i, 1)$.*
Write

$$\widetilde{\boldsymbol{p}} = \{\widetilde{p}_1, \widetilde{p}_2, \ldots, \widetilde{p}_K\} \stackrel{\Delta}{=} \left(\frac{\gamma_1}{\sum_{j=1}^K \gamma_j}, \frac{\gamma_2}{\sum_{j=1}^K \gamma_j}, \cdots \frac{\gamma_K}{\sum_{j=1}^K \gamma_j} \right). \tag{D15}$$

Then

$$\widetilde{\boldsymbol{p}} \sim Dir_K(\boldsymbol{\alpha}). \tag{D16}$$

Proof of Lemma D.2.
We suppose that the random variables $\gamma_1, \gamma_2, \ldots, \gamma_{K+1}$ are independently sampled from Gamma distributions with scale parameters (respectively) $\alpha_1, \alpha_2, \ldots, \alpha_{K+1}$. Here each distribution also has a common shape parameter $k = 1$. The joint probability density for this collection of independent Gamma variables is,

$$\varphi(\gamma_1, \gamma_2, \ldots, \gamma_{K+1}) = \prod_{j=1}^{K+1} \frac{1}{\Gamma(\alpha_j)} \gamma_j^{\alpha_j - 1} \exp(-\gamma_j). \tag{D17}$$

Recall that $0 < \gamma_j < \infty$.
New random variables $u_1, u_2, \ldots, u_K, u_{K+1}$ are now defined by

$$u_i \stackrel{\Delta}{=} \frac{\gamma_i}{\gamma_1 + \gamma_2 + \cdots \gamma_{K+1}}, \qquad i \in \{1, 2, \ldots, K\} \tag{D18}$$

$$u_{K+1} \stackrel{\Delta}{=} \gamma_1 + \gamma_2 + \cdots \gamma_{K+1}. \tag{D19}$$

What we would like to do is compute the joint probability density for the random variables $u_1, \ldots, u_K, u_{K+1}$, noting that u_{K+1} is a Gamma random variable with shape parameter $\sum_{\ell=1}^{K+1} \alpha_\ell$. This calculation may be carried out by using the Jacobian of transformation and the following inverse functions:

$$\gamma_1 = u_1 u_{K+1}, \tag{D20}$$

$$\gamma_1 = u_2 u_{K+1}, \tag{D21}$$

$$\vdots \quad \vdots$$

$$\gamma_{K+1} = u_{K+1}(1 - u_1 - u_2 - \cdots - u_K). \tag{D22}$$

The corresponding Jacobian has the form,

$$J(u_1, u_2, \ldots, u_{K+1}, \gamma_1, \gamma_2, \ldots \gamma_{K+1}) =$$

$$\begin{vmatrix} u_{K+1} & 0 & \cdots & 0 & u_1 \\ 0 & u_{K+1} & \cdots & 0 & u_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & u_{K+1} & u_K \\ -u_{K+1} & -u_{K+1} & \cdots & -u_{K+1} & (1 - u_1 - u_2 - \cdots - u_K) \end{vmatrix} = \big(u_{K+1}\big)^K. \quad \text{(D23)}$$

Writing $f(\cdot)$ for the joint probability density function of $u_1, u_2, \ldots, u_K, u_{K+1}$, we see that

$$\begin{aligned} f\big(u_1, u_2, \ldots u_K, u_{K+1}\big) &= \varphi\big(u_1 u_{K+1}, \ldots u_K u_{K+1}, u_{K+1}(1 - u_1 - u_2 - \cdots - u_K)\big)|J| \\ &= \left(\prod_{j=1}^{K} \frac{1}{\Gamma(\alpha_j)} \big(u_j u_{K+1}\big)^{\alpha_j - 1} \exp(-u_j u_{K+1})\right) \times \\ &\quad \frac{1}{\Gamma(\alpha_{K+1})} \Big(u_{K+1}\big(1 - u_1 - u_2 - \cdots - u_K\big)\Big)^{\alpha_{K+1}-1} \times \\ &\quad \exp\big(-u_{K+1}(1 - u_1 - \cdots - u_K)\big)|J| \\ &= \Big(\prod_{j=1}^{K} \frac{1}{\Gamma(\alpha_j)} u_j^{\alpha_j - 1}\Big) \frac{1}{\Gamma(\alpha_{K+1})} u_{K+1}^{\left(\sum_{\ell=1}^{K} \alpha_\ell\right)} u_{K+1}^{-K} |J| \exp\big(-u_{K+1}\big) \times \\ &\quad \big(u_{K+1}(1 - u_1 - u_2 - \cdots - u_K)\big)^{\alpha_{K+1}-1} \\ &= \Big(\prod_{j=1}^{K} \frac{1}{\Gamma(\alpha_j)} u_j^{\alpha_j - 1}\Big) \frac{1}{\Gamma(\alpha_{K+1})} \times \\ &\quad \big(1 - u_1 - u_2 - \cdots - u_K\big)^{\alpha_{K+1}-1} u_{K+1}^{\left(\sum_{\ell=1}^{K+1} \alpha_\ell - 1\right)} \exp\big(-u_{K+1}\big). \end{aligned} \quad \text{(D24)}$$

To complete our calculations we marginalise out the (independent) Gamma random variable u_{K+1}, that is.

$$\begin{aligned} g(u_1, u_2, \ldots u_K) &\triangleq \int_0^\infty f(u_1, u_2, \ldots u_K, \xi) d\xi \\ &= \Big(\prod_{j=1}^{K} \frac{1}{\Gamma(\alpha_j)} u_j^{\alpha_j - 1}\Big) \frac{1}{\Gamma(\alpha_{K+1})} \big(1 - u_1 - u_2 - \cdots - u_K\big)^{\alpha_{K+1}-1} \times \\ &\quad \Gamma\big(\textstyle\sum_{\ell=1}^{K+1} \alpha_\ell\big) \displaystyle\int_0^\infty \frac{1}{\Gamma\big(\sum_{\ell=1}^{K+1} \alpha_\ell\big)} \xi^{\left(\sum_{\ell=1}^{K+1} \alpha_\ell - 1\right)} \exp(-\xi) d\xi. \end{aligned} \quad \text{(D25)}$$

The last integral directly above is a Gamma probability density integrated over its entire domain and so is unity. Consequently

$$g(u_1, u_2, \ldots u_K) = \frac{\Gamma(\alpha_1 + \alpha_2 + \cdots + \alpha_{K+1})}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_{K+1})} u_1^{\alpha_1 - 1} \cdots u_K^{\alpha_K - 1} \big(1 - u_1 - u_2 - \cdots - u_K\big)^{\alpha_{k+1}-1} \quad \square. \quad \text{(D26)}$$

Remark D.5. *The above technique to simulate Dirichlet random variables via the sampling of a Gamma distribution is commonly used. Moreover, since this is an exact simulation technique, (ie equality in distribution), this result can be used to establish some interesting properties of Dirichlet random variables, such as the aggregation property established in the next section. There are indeed alternative techniques to simulate Dirichlet random variables, for example, five different simulation techniques are presented in [Nar90]. The simulation is Dirichlet random variables is also discussed in [HC70],[Dev86] and [Ken09].*

D.4 Aggregation Property

The so-called aggregation property of the Dirichlet distribution is particularly important in Latent Dirichlet Allocation when applied to topic modelling in text analysis. Perhaps the most popular approach to derive this property, is to make use make use of an additive property for Gamma random variables with common scale-parameters values. This property and the generation of Dirichlet random variables via Gamma random variables, may be combined to establish the aggregation property of Dirichlet random variables.

Suppose X, Y are Gamma random variables with scale parameters θ_X and θ_Y. Further, suppose the shape parameters of these random variables are k_X and k_Y. If $k_X = k_Y$, then the following equality holds (in distribution),

$$\mathrm{Gam}\big((\theta_X + \theta_Y), k\big) =_d \mathrm{Gam}\big(\theta_X, k\big) + \mathrm{Gam}\big(\theta_Y, k\big). \tag{D27}$$

A simulation example of this property is shown in Figure D1. In this example $Z \sim \mathrm{Gam}(5, 2.5)$, $X \sim \mathrm{Gam}(1.7, 2.5)$ and $Y \sim \mathrm{Gam}(3.3, 2.5)$. The resulting estimated statistics were: $E[Z] = 12.5262$, $E\big[(Z - E[Z])^2\big] = 31.2424$, $E[X+Y] = 12.4906$ and $E\big[\big((X+Y) - E[X+Y]\big)^2\big] = 31.2135$. The histograms in Figure D1 were generated from i.i.d. sets of 10,000 samples each. Suppose that $\varphi_i \sim \mathrm{Gam}\big(\alpha_i, 1\big)$, where $\alpha_i > 0$ for $i \in \{1, 2, \ldots, K\}$. Write

$$\boldsymbol{\varphi} \triangleq \big(\varphi_1, \varphi_2, \ldots, \varphi_K\big), \tag{D28}$$

$$\mathbf{1} \triangleq (1, 1, \ldots, 1) \in \mathbb{R}^K. \tag{D29}$$

It was shown above in §D.3 that

$$\boldsymbol{p} \triangleq \left(\frac{\varphi_1}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}, \frac{\varphi_2}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}, \ldots, \frac{\varphi_K}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}\right) \sim \mathrm{Dir}_K(\boldsymbol{p}, \boldsymbol{\alpha}). \tag{D30}$$

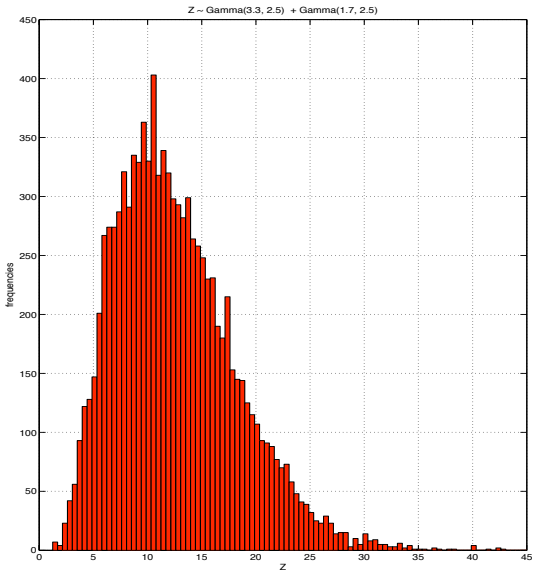
Suppose the index set $\{1, 2, \ldots, K\}$ is decomposed into a union of mutually disjoint sets as follows,

$$\big\{1, 2, \ldots, K\big\} = I_1 \cup I_2 \cdots I_{\mathrm{M}}, \qquad I_i \cap I_j = \emptyset, \ \forall\, i \neq j. \tag{D31}$$

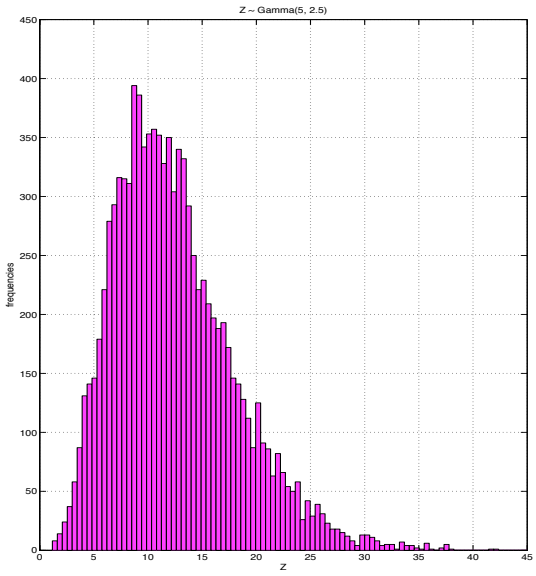
We now consider aggregated "states" formed through various additions of the probabilities p_i, for example,

$$\widetilde{p}_j \triangleq \sum_{i \in I_j} p_i. \tag{D32}$$

Figure D1: *Simulated example of the shape-parameter additive property for gamma distributed random variables.*



(a) $\widetilde{Z} \triangleq X + Y$, where the two independent random variables X and X are distributed as $X \sim \text{Gamma}(1.7, 2.5)$ and $Y \sim \text{Gamma}(3.3, 2.5)$.



(b) $Z \sim \text{Gamma}(5, 2.5)$.

Then

$$\begin{aligned}\{\widetilde{p}_2, \widetilde{p}_2, \ldots \widetilde{p}_M\} &= \left(\sum_{i \in I_1} p_i, \sum_{i \in I_2} p_i, \ldots, \sum_{i \in I_M} p_i\right) \\ &=_{\text{d}} \left(\frac{\sum_{i \in I_1} \varphi_i}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}, \frac{\sum_{i \in I_2} \varphi_i}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}, \ldots \frac{\sum_{i \in I_M} \varphi_i}{\langle \boldsymbol{\varphi}, \mathbf{1} \rangle}\right) \\ &=_d \text{Dir}_M(\{\widetilde{p}_1, \widetilde{p}_1, \ldots, \widetilde{p}_M\},).\end{aligned} \tag{D33}$$

Appendix E Gibbs Sampling

E.1 Background

Gibbs Sampling is a very well known numerical method in statistics based upon Markov Chain Monte Carlo (MCMC) methods. In particular, this method provides (in some cases) a means to sample joint probability distributions of N dimensions by instead sampling N univariate single-dimensional marginal distributions. There are many excellent books covering MCMC schemes which have in recent years become enormously popular in areas such as filtering and quantitative finance. The interested reader is referred to [Gam97] for an introductory level treatment. More technical details on Gibbs sampling may be found in [Asm10], [GRS96] and [RC05]. The statistical literature also contains numerous expository articles on Gibbs Sampling schemes and theory, for example [CLR01, CG92, CG95, SG92, GS90, BGHM95, Gey92].
The seminal ideas of Gibbs sampling were introduced in the article [GG84]. However an equally important article written by Gelfand and Smith demonstrated a much broader class of problems that could be solved through Gibbs Sampling, see [GS90]

The theory of this scheme is based primarily upon asymptotic properties Markov chains, consequently Gibbs sampling is sometimes referred to as Markov Chain Monte Carlo (MCMC). The relevant Markov chain theory in support of Gibbs sampling is discussed in [Nor98] and [HÖ4].

E.2 Basic Markov Chain Monte Carlo (MCMC)

Monte Carlo estimation in general concerns some form of approximation achieved through simulation. In MCMC, the simulation device used for the approximation is one or more Markov chains. These chains are of course not arbitrary and must have certain properties for Gibbs Sampling (and indeed other schemes) to achieve the desired results. Three fundamental Markov chain state-properties needed for Gibbs sampling are 1) irreducibility, 2) recurrence, and 3) aperiodicity.

Definition E.1 (Irreducibility of Markov Chains). *Consider a finite Markov chain with state space $\mathcal{S} \stackrel{\Delta}{=} \{s_1, s_2, \ldots, s_M\}$. In a rough way of speaking, "irreducibility" means any state in $\mathcal{S}$ can be reached from any other state in $\mathcal{S}$. To make this precise the state-classifying notion of "communicating" states is used, that is, s_i communicates with s_j (written $s_i \to s_j$) if the chain has positive probability of ever reaching s_j when it starts in state s_i. Communicating states may also be written by stating there exists a natural number N, such that,*

$$Prob(X_{m+N} = s_j \mid s_m = s_i) > 0. \tag{E1}$$

Note that for a homogeneous Markov chain this probability is independent of m. If $s_i \to s_J$ and $s_j \to s_i$, then the states s_i and s_j are said to intercommunicate, this is written as $s_i \leftrightarrow s_j$.
Communication of states can be shown to be an equivalence relation (see [Ros96]).
A Markov chain with state space $\mathcal{S}$ is said to be irreducible if for all pairs of states $s_i, s_j \in \mathcal{S}$ we have $s_i \leftrightarrow s_j$.

Remark E.1. *If a Markov chain is not irreducible it is then reducible. This means the long term behaviour of such a chain may be analysed by one or more smaller state space Markov chains, hence the term "reducible"*

Definition E.2 (Recurrence). *A Markov chain X is said to be recurrent if the following certain event holds for all states,*

$$prob\big(X_M = s_i \ \text{ for infinitely many } M\big) = 1. \tag{E2}$$

Definition E.3 (Aperiodicity). *A definition of aperiodicity first requires basic notions of divisibility of integers. A common divisor of two integers a and b is a third integer d such that $d|a$ and $d|b$. The greatest common divisor (g.c.d.) of two non-zero integers a and b, is the largest d, such that both $d|a$ and $d|b$. The g.c.d. is sometimes referred to as the highest common factor (h.c.f.). Clearly the g.c.d. can be extended to a collection of more than two integers.*
The period of a state s_i of a Markov chain is defined by

$$\ell_i \triangleq g.c.d.\big\{M \in \mathbb{N} \mid M \geq 1, p_{ii}^M > 0\big\}. \tag{E3}$$

Here

$$p_{ii}^M = prob\{X_{n+M} = s_i \mid X_n = s_i\}. \tag{E4}$$

If $\ell_i = 1$ for all states in a Markov chain, then that chain is said to be aperiodic.

Remark E.2. *Loosely speaking, the period ℓ_i of a Markov chain is the g.c.d. of the set of times (discrete indexes representing time) the chain in question can return to s_i given it started in s_i.*

Indeed much more can be said about the definitions above, the interested reader is referred to [Bre99]

The main consequences for a Markov chain having these three properties is the existence of a stationary distribution (other important properties are the strong law of large numbers will hold and also the chain will be ergodic).
Consequently, the basic idea of Gibbs Sampling is to identify a Markov chain whose stationary distribution is the distribution of interest, or a good approximation to it. Then one samples this Markov chain in an appropriate way to generate samples from the original target distribution, from which all statistics of interest can be estimated. In LDA this approximation substantially reduces the dimension and complexity of the estimation tasks given the large dimensions usually arising in LDA topic modelling problems.

Remark E.3. *It should be noted that the above definitions consider Markov chains on a finite state space, however, Gibbs sampling is not limited to such Markov chains and can be extended to far more general state spaces. This however raises natural questions about the extensions of the above definitions and the existence of a corresponding stationary distribution in such settings. The details of this extension for Gibbs sampling are not trivial and must consider more delicate definitions, such as a transition probability Kernel rather than a transition matrix. For example, suppose we write the transition Kernel from the vector $\boldsymbol{x}$ to the vector $\boldsymbol{y}$ (both in $\mathbb{R}^d$), as $Q(\boldsymbol{x}, \boldsymbol{y})$. Then for $A \subset \mathbb{R}^d$*

$$prob\big(\boldsymbol{Y} \in A \mid \boldsymbol{X} = \boldsymbol{x}\big) = \int_A Q(\boldsymbol{x}, \boldsymbol{y}) d\boldsymbol{y}. \tag{E5}$$

Now the analogue of solving the balance equations for a Markov chain is finding a solution to an integral equation, namely

$$\pi(\boldsymbol{y}) = \int Q(\boldsymbol{x}, \boldsymbol{y})\pi(\boldsymbol{x})d\boldsymbol{x}. \tag{E6}$$

If a solution to (E6) can be found then $\pi(\boldsymbol{y})$ is the corresponding stationary distribution. These details can be found in [MT93],[CMR05] and [RC05].

E.3 Example

Finally, to fix some basic ideas of Gibbs sampling we recall a commonly studied example concerning a bivariate distribution. In some sense this example is vacuous given it concerns a bivariate density and so can be directly addressed, however, this example does offer a clear and convincing illustration of the Gibbs sampling technique and it's trivial to encode in a digital computer.

In this example we consider a bivariate probability density where one variable is integer-valued with finite range and the other variable is continuously-valued on the compact set $[0, 1]$. Explicitly, our joint density has the form,

$$f(k, y) = \binom{N}{k} y^{k+\alpha-1}(1-y)^{N-k+\beta-1} \in \mathbb{N} \times [0, 1]. \tag{E7}$$

Here k takes value in the set $\{0, 1, 2, \ldots, N\}$, and $y \in [0, 1]$.
Obviously it's not immediately clear how one might sample from the non-trivial density given at (E7). Further, it's also not a simple task to compute basic statistics such as the mean and variance for the density at (E7). By inspection however, one might guess that k is some type of random variable with a binomial distribution and that y is some sort of Beta distributed random variable.
Suppose we assume y is known and fixed, then

$$\text{prob}(k \mid y) \propto \binom{N}{k} y^k (1-y)^{-k} \tag{E8}$$

for all $k \in \{0, 1, \ldots N\}$.
Similarly we suppose k is fixed and known, then,

$$\text{prob}(y \mid k, \alpha, \beta) \propto y^{k+\alpha-1}\big(1-y\big)^{N-k\beta-1}. \tag{E9}$$

Here $\alpha > 0$ and $\beta > 0$.
It can be shown that the probability density for k alone has the following form,

$$f(k) = \binom{N}{k} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \frac{\Gamma(k+\alpha)\Gamma(N-k+\beta)}{\Gamma(\alpha+\beta+N)}. \tag{E10}$$

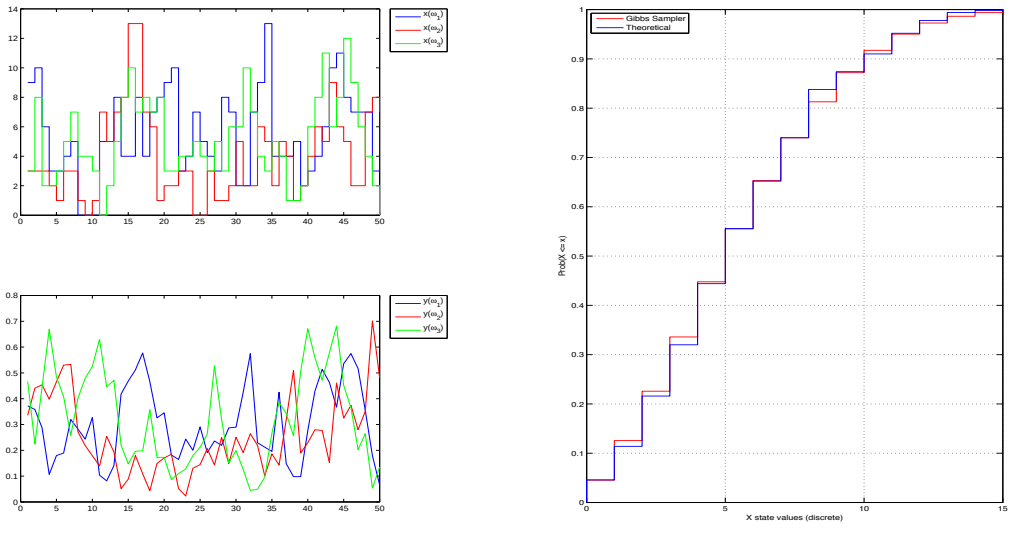
This exact probability density is useful to compare against the Gibbs Sampler.
Following the article [CG92], we consider an explicit scenario with $\alpha = 2$, $\beta = 4$ and $N = 16$.

Algorithm 3 Gibbs Sampling Algorithm

1: Choose a number of iterations $K \in \mathbb{N}$, $K \geq$ "burn in"
2: By some means choose an initial $k \in \{0, 1, 2 \ldots, 15\}$
3: Last-k is assigned the value k
4: **for** $\ell = 1, 2, 3, \ldots, K$ **do**
5: Sample $y_\ell \sim \text{Beta}(\text{Last-}k + 2, 16 + 4)$
6: Sample $k_\ell \sim \text{Binomial}(N, y_\ell)$
7: Last-k is assigned the value k_ℓ
8: Store Gibbs-sequence value: $\Theta_\ell \triangleq (y_\ell, k_\ell)$
9: **end for**

Consequently the Gibbs sampling algorithm for the joint density (E7) is given in algorithmic form at Algorithm (3). Our Gibbs sampler was run 500 times with a realisation length of 2000 samples. At the conclusion of each of the 500 runs the final value for k was stored. Using these k values an empirical estimate was computed for the probability distribution. This estimate was then compared against the exact distribution function which was computed via the density given at (E10). The estimated (cumulative) distribution function was computed in Matlab with the Kaplan-Meier estimator (see [ABGK93, LK58]). The results of this comparison are shown in Figure E1. Typical Gibbs sampler realisations of (k, y) are also shown in Figure E1.

Figure E1: *Gibbs Sampler example for the bivariate density given at (E7)*



(a) Typical realisations of $\Theta = (x, y)$ (b) True and estimated Distribution Functions

Appendix F Sample Elicited Text Data

This Appendix provides a sample of real elicited text data. This data set was collected at the NICTA/DSTO Text collection day and is shown here. Each text entry is tagged with a date stamp and time stamp and also with an index labelling a specific (but anonymous) workshop attendee. This particular data-set was generated in the session corresponding to Table 2 of §**??**. The remaining two sessions from this workshop generated similar data sets. The example below is included here to show a sampling of typical elicited text from a collection of uses. Note that some attendees use punctuation and so do not, some use capitalization and some do not, *etc*.

- **1.1** Publications and Journal Rankings
  `Submitted by 19 (2011-03-24 22:14:55)`
- **1.2** The number of citations both primary and secondary citations
  `Submitted by 14 (2011-03-24 22:15:15)`
- **1.3** understanding / interpretation of the research
  `Submitted by 21 (2011-03-24 22:15:28)`
- **1.4** Against the standard criteria of number publications, media used for publishing, citations
  `Submitted by 2 (2011-03-24 22:15:28)`
- **1.5** Our understanding of problems improves .
  `Submitted by 6 (2011-03-24 22:15:34)`
- **1.6** Several ways of measuring the value of research - publications, general interest, applications with a definable benefit. However, in my opinion, the BEST way of measuring the value of research is not what it solves, but the number of new questions it opens up for further research
  `Submitted by 9 (2011-03-24 22:15:41)`
- **1.7** Depends on the funding source. Value to university is rather nebulous: fame, useful to students, publicity, awards gained, funding subsequently gained.
  `Submitted by 4 (2011-03-24 22:15:41)`
- **1.8** number of citations
  `Submitted by 6 (2011-03-24 22:15:42)`
- **1.9** In relation to applicability and generalisability of research outcomes; practical value of work
  `Submitted by 2 (2011-03-24 22:16:04)`
- **1.10** Research can be measured in a number of ways, both short-term and long-term. Short-term measurements include number of publications, direct funding grants for that research, patents granted and patents licensed, or external auditing. Long-term research is probably best measure in terms of influence, that is the number of citations that work has received after five or ten years (also residual long-term technology and patent licensing).
  `Submitted by 5 (2011-03-24 22:16:14)`

- ◇ 1.11 Measures should be quantifiable and meaningful.
  Submitted by 4 (2011-03-24 22:16:22)
- ◇ 1.12 Research is best measured by the amount of subsequent product that produced by the research results.
  Submitted by 17 (2011-03-24 22:16:30)
- ◇ 1.13 Future outcome (money, businesses, renewal of science) based on this research
  Submitted by 2 (2011-03-24 22:16:31)
- ◇ 1.14 impact on the direction of other research teams
  Submitted by 14 (2011-03-24 22:16:53)
- ◇ 1.15 Decisions made based on the research. Contributions to and assisting existing research. Outcomes in the form of new inventions. Adding to body of existing information. Providing an understanding of a particular problem
  Submitted by 11 (2011-03-24 22:16:59)
- ◇ 1.16 Basic and applied research (and the many shades between) require different measurements since basic research tends to have a longer time before it has impact whereas applied research has more immediate results.
  Submitted by 1 (2011-03-24 22:17:09)
- ◇ 1.17 improvements in quality of life (medicine), better understanding of natural phenomena. Reduces the influence of dogma.
  Submitted by 6 (2011-03-24 22:17:12)
- ◇ 1.18 The value of research cannot be measured. If I knew what the result is, it is not research.
  Submitted by 17 (2011-03-24 22:17:22)
- ◇ 1.19 I would tend to believe more strongly in long-term measurements than short-term measurements, but funding agencies of course must rely on short-term measurements except when funding people, who have an established track record.
  Submitted by 5 (2011-03-24 22:17:27)
- ◇ 1.20 Relation resources put to the research and value got from it
  Submitted by 2 (2011-03-24 22:17:28)
- ◇ 1.21 Carefully.
  Submitted by 1 (2011-03-24 22:17:30)
- ◇ 1.22 Value could be indirect, like Central Limit Theorem, which is a fundamental thing used elsewhere.
  Submitted by 4 (2011-03-24 22:17:31)
- ◇ 1.23 Depends on industry and comparing timelines of similar problems. Is there a tangible outcome or product from the research. Will development continue when the research is completed.
  Submitted by 20 (2011-03-24 22:17:31)
- ◇ 1.24 The value of a research depends on who is measuring it. For universities, it will depend on the amount of citations it receives and publication rankings
  Submitted by 19 (2011-03-24 22:17:57)

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. CAVEAT/PRIVACY MARKING

2. TITLE	3. SECURITY CLASSIFICATION
Numerical Algorithms for the Analysis of Expert Opinions Elicited in Text Format	Document (U) Title (U) Abstract (U)
4. AUTHORS	5. CORPORATE AUTHOR
W. P. Malcolm and Wray Buntine	Defence Science and Technology Organisation Fairbairn Business Park, Department of Defence, Canberra, ACT 2600, Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO–TR–2797	015-501	Technical Report	April, 2013

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. No. OF PAGES	12. No. OF REFS
	07/271	Capability Development Group	68	130

13. URL OF ELECTRONIC VERSION	14. RELEASE AUTHORITY
`http://www.dsto.defence.gov.au/publications/scientific.php`	Chief, Joint Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for Public Release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SOUTH AUSTRALIA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS

No Limitations

18. DSTO RESEARCH LIBRARY THESAURUS

Natural Language Processing, Text Analysis, Sentiment Analysis, Key Phrase Estimation, Probabilistic Topic Estimation, Latent Dirichlet Allocation, Differential Analysis of Stake-Holder Text, Bayesian Estimation, Monte Carlo Methods.

19. ABSTRACT

Latent Dirichlet Allocation (LDA) is a scheme which may be used to estimate topics and their probabilities within a corpus of text data. The fundamental assumptions in this scheme are that text is a realisation of a stochastic generative model and that this model is well described by the combination of multinomial probability distributions and Dirichlet probability distributions. Various means can be used to solve the Bayesian estimation task arising in LDA. Our formulations of LDA are applied to subject matter expert text data elicited through carefully constructed decision support workshops. In the main these workshops address substantial problems in Australian Defence Capability. The application of LDA here is motivated by a need to provide insights into the collected text, which is often voluminous and complex in form. Additional investigations described in this report concern questions of identifying and quantifying differences between stake-holder group text written to a common subject matter. Sentiment scores and key-phase estimators are used to indicate stake-holder differences. Some examples are provided using unclassified data.