JMLR: Workshop and Conference Proceedings 29:1–12, 2014 ACML 2014

Supplementary Material: Bibliographic Analysis with the Citation Network Topic Model

Kar Wai Lim KARWAI.LIM@ANU.EDU.AU
Australian National University, Canberra, Australia
NICTA, Canberra, Australia

Wray Buntine WRAY.BUNTINE@MONASH.EDU
Monash University, Clayton, Australia

Appendix A. On Modelling the Document-topic Hierarchy

Here we discuss the motivation of modelling the document-topic hierarchy (θ and θ') in more details. As mentioned in the paper, such modelling allows the citation information to be given more strength compared to the text information. Recall that each citation and each word correspond to a customer count (see Section 4 and Section 5 in the paper), thus we would have more customer counts for the text information than that of the citation information. An issue with this is that the text information would overwhelm the citation information if they are treated equally. One way to address this problem would be to model the document-topic hierarchy, as the text information is discounted in θ'. Additionally, since word tokens are not equal to citation tokens, we argue that they really should not be coming from the same parent node.

We illustrate with an example, say we have a document with only 2 citations and 6 words, then $c^{\theta} = 6$ corresponds to the text information. If we do not model the document-topic hierarchy, the citation information would contribute 2 counts to c^{θ}, where its effect would be overwhelmed by the text information. If we model the document-topic hierarchy, then the text information get passed up to θ' in the form of table count, t^{θ}, which is lower than c^{θ}. This table count becomes the customer count in θ', which means the text information now contributes less count in θ'. The citation information contributes 2 count to $c^{\theta'}$ directly, its effect is now relatively greater. In our experiments, we find that the table counts in θ tend to be about half of the customer counts in θ.

Appendix B. Delta Method Approximation

We employ the delta method to show that

$$\int f(\theta)\, \exp(-g(\theta))\, \mathrm{d}\theta \approx \exp(-g(\hat{\theta})) \int f(\theta)\, \mathrm{d}\theta$$

for a multivariate distribution characterised by $p(\theta)$ that is proportional to $f(\theta)$. Here, $\hat{\theta}$ is the expected value of the the multivariate distribution:

$$\hat{\theta} = \mathbb{E}[\theta] = \int \theta\, p(\theta)\, \mathrm{d}\theta \quad , \qquad\qquad f(\theta) = c \times p(\theta) \quad .$$

First we apply a one step Taylor approximation for a function $h(\theta) = \exp(-g(\theta))$ at $\hat{\theta}$:

$$h(\theta) \approx h(\hat{\theta}) + \sum_i h_i'(\hat{\theta})\, (\theta_i - \hat{\theta}_i) \quad , \tag{1}$$

where $h_i'(\hat{\theta})$ denotes the i-th partial derivative of $h(\cdot)$ evaluated at $\hat{\theta}$:

$$h_i'(\hat{\theta}) = -g_i'(\hat{\theta})\, h(\hat{\theta}) \quad .$$

Multiply Equation 1 with $f(\theta)$ and integrating gives

$$\begin{aligned}
\int f(\theta)\, h(\theta)\, \mathrm{d}\theta &= \int f(\theta) \left(h(\hat{\theta}) + \sum_i h_i'(\hat{\theta})(\theta_i - \hat{\theta}_i) \right) \mathrm{d}\theta \\
&= h(\hat{\theta}) \int f(\theta)\, \mathrm{d}\theta + \sum_i h_i'(\hat{\theta}) \underbrace{\int f(\theta)\, (\theta_i - \hat{\theta}_i)\, \mathrm{d}\theta}_{0} \\
&= h(\hat{\theta}) \int f(\theta)\, \mathrm{d}\theta \quad .
\end{aligned} \tag{2}$$

Appendix C. Metropolis-Hastings Algorithm for Citation Network

We detail our MH algorithm for the citation network as follows. First, for each document i, we estimate the expected document-topic prior θ'_i with

$$\hat{\theta}'_i = \left(\cdots, \frac{(\alpha^{\theta'_i} T^{\theta'_i} + \beta^{\theta'_i})\nu_{a_i k} + c_k^{\theta'_i} - \alpha^{\theta'_i} T_k^{\theta'_i}}{\beta^{\theta'_i} + C^{\theta'_i}}, \cdots \right) . \tag{3}$$

Note that ν_{a_i} in Equation 3 is recursively computed from μ and its associated counts.

Recall that we jointly model x_{ij} and y_{ij} as a Poisson distribution:

$$x_{ij}, y_{ij} = k | \lambda, \theta' \sim \text{Poisson}\left(\lambda_i^+ \lambda_j^- \lambda_k^T \theta'_{ik} \theta'_{jk}\right) . \tag{4}$$

Then, for each document pair (i, j) where $x_{ij} = 1$, we decrement the network counts associated with x_{ij}, and re-sample y_{ij} with the proposal distribution derived from Equation 4:

$$p(y_{ij}^{\text{new}} = k | \hat{\theta}'_i, \hat{\theta}'_j) \propto \lambda_k^T \hat{\theta}'_{ik} \hat{\theta}'_{jk} \exp\left(-\lambda_i^+ \lambda_j^- \lambda_k^T \hat{\theta}'_{ik} \hat{\theta}'_{jk}\right) \quad . \tag{5}$$

which can be further simplified since the terms inside the exponential are very small, hence the exp term approximates to 1. We empirically inspected the exponential term and we found that almost all of them are between 0.99 and 1. This means the ratio of the exponentials is not significant for sampling new citing topic y_{ij}^{new}. So

$$p(y_{ij}^{\text{new}} = k | \hat{\theta}'_i, \hat{\theta}'_j) \propto \lambda_k^T \hat{\theta}'_{ik} \hat{\theta}'_{jk} \quad . \tag{6}$$

Using the superscripts $\square^{\text{new}}$ and $\square^{\text{old}}$ to denote the proposed sample and the old value respectively, we compute the acceptance probability A for the newly sampled $y_{ij}^{\text{new}} = y'$, changed from $y_{ij} = y^*$, and the successive change to the document-topic priors θ' (to θ'^{new}):

$$A = \frac{\exp\left(-\sum_{ijk} \lambda_i^+ \lambda_j^- \lambda_k^T \; \hat{\theta'}_{ik}^{\text{new}} \; \hat{\theta'}_{jk}^{\text{new}}\right)}{\exp\left(-\sum_{ijk} \lambda_i^+ \lambda_j^- \lambda_k^T \hat{\theta}'_{ik} \hat{\theta}'_{jk}\right)} \frac{p(\mathbf{Z}, \mathbf{W}, \mathbf{T}, \mathbf{C}^{+\text{new}} | \zeta)}{p(\mathbf{Z}, \mathbf{W}, \mathbf{T}, \mathbf{C}^{+\text{old}} | \zeta)} \frac{\lambda_{y^*}^T \; \hat{\theta'}_{iy^*}^{\text{new}} \; \hat{\theta'}_{jy^*}^{\text{new}}}{\lambda_{y'}^T \theta'_{iy'} \theta'_{jy'}} \frac{\sum_k \lambda_k^T \theta'_{ik} \theta'_{jk}}{\sum_k \lambda_k^T \; \hat{\theta'}_{ik}^{\text{new}} \; \hat{\theta'}_{jk}^{\text{new}}} . \tag{7}$$

Note that we have abused the notations i and j in the above equation, where the i and j in the summation indexes all documents instead of pointing to particular document i and document j. We decided against introducing additional variables to make things less confusing.

Finally, if the sample is accepted, we update y_{ij} and the associated customer counts. Otherwise, we discard the sample and revert the changes.

Appendix D. Sampling the Concentration Parameter for PYPs

Here we outline the procedure to sample the concentration parameter $\beta^{\mathcal{N}}$ of a PYP distributed variable $\mathcal{N}$, using an auxiliary variable sampler. Assuming each $\beta^{\mathcal{N}}$ has a Gamma distributed hyperprior with shape τ_0 and rate τ_1, we first sample the auxiliary variables ξ and ψ_j for $j \in \{0, T^{\mathcal{N}} - 1\}$:

$$\xi|\beta^{\mathcal{N}} \sim \text{Beta}(C^{\mathcal{N}}, \beta^{\mathcal{N}}) \quad , \qquad \psi_j|\alpha^{\mathcal{N}}, \beta^{\mathcal{N}} \sim \text{Bernoulli}\left(\frac{\beta^{\mathcal{N}}}{\beta^{\mathcal{N}} + j\alpha^{\mathcal{N}}}\right) \quad . \tag{8}$$

We then sample a new $\beta'^{\mathcal{N}}$ from the following conditional posterior given the auxiliary variables:

$$\beta'^{\mathcal{N}}|\xi, \psi \sim \text{Gamma}\left(\tau_0 + \textstyle\sum_j \psi_j, \tau_1 - \log(1-\xi)\right) \quad . \tag{9}$$

We note that we apply a vague hyperprior for $\beta^{\mathcal{N}}$ by setting $\tau_0 = \tau_1 = 1$ in this paper.

Appendix E. Keywords for Querying CiteSeerX Datasets

1. For ML dataset:

Machine Learning: *neural network, pattern recognition, indexing term, support vector machine, learning algorithm, machine learning, computer vision, face recognition, feature extraction, image processing, high dimensionality, image segmentation, pattern classification, real time, feature space, decision tree, principal component analysis, feature selection, backpropagation, edge detection, object recognition, maximum likelihood, statistical learning theory, supervised learning, reinforcement learning, radial basis function, support vector, em algorithm, self organization, image analysis, hidden markov model, artificial neural network, independent component analysis, genetic algorithm, statistical model, dimensional reduction, indexation, unsupervised learning, gradient descent, large scale, maximum likelihood estimate, statistical pattern recognition, cluster algorithm, markov random field, error rate, optimization problem, satisfiability, high dimensional data, mobile robot, nearest neighbour, image sequence, neural net, speech recognition, classification accuracy, diginal image processing, factor analysis, wavelet transform, local minima, probability distribution, back propagation, parameter estimation, probabilistic model, feature vector, face detection, objective function, signal processing, degree of freedom, scene analysis, efficient algorithm, computer simulation, facial expression, learning problem, machine vision, dynamic system, bayesian network, mutual information, missing value, image database, character recognition, dynamic program, finite mixture model, linear discriminate analysis, image retrieval, incomplete data, kernel method, image representation, computational complexity, texture feature, learning method, prior knowledge, expectation maximization, cost function, multi layer perceptron, iterated reweighted least square, data mining.*

2. For M10 dataset:

Biology: *enzyme, gene expression, amino acid, escherichia coli, transcription factor, nucleotides, dna sequence, saccharomyces cerevisiae, plasma membrane, embryonics.*

Computer Science: *neural network, genetic algorithm, machine learning, information retrieval, data mining, computer vision, artificial intelligent, optimization problem, support vector machine, feature selection.*

Social Science: *developing country, higher education, decision making, health care, high school, social capital, social science, public health, public policy, social support.*

Financial Economics: *stock returns, interest rate, stock market, stock price, exchange rate, asset prices, capital market, financial market, option pricing, cash flow.*

Material Science: *microstructures, mechanical property, transmission electron microscopy, grain boundary, composite material, materials science, titanium, silica, differential scanning calorimetry, tensile properties.*

Physics: *magnetic field, quantum mechanics, field theory, black hole, kinetics, string theory, elementary particles, quantum field theory, space time, star formation.*

Petroleum Chemistry: *fly ash, diesel fuel, methane, methyl ester, diesel engine, natural gas, pulverized coal, crude oil, fluidized bed, activated carbon.*

Industrial Engineering: *power system, construction industry, induction motor, power converter, control system, voltage source inverter, permanent magnet, digital signal processor, sensorless control, field oriented control.*

Archaeology: *radiocarbon dating, iron age, bronze age, late pleistocene, middle stone age, upper paleolithic, ancient dna, early holocene, human evolution, late holocene.*

Agriculture: *irrigation water, soil water, water stress, drip irrigation, grain yield, growing season, crop yield, soil profile, soil salinity, crop production*

3. For AvS dataset:

History: *nineteeth century, cold war, south africa, foreign policy, civil war, world war ii, latin america, western europe, vietnam, middle east.*

Religion: *social support, foster care, child welfare, human nature, early intervention, gender difference, sexual abuse, young adult, self esteem, social services.*

Physics: *magnetic field, quantum mechanics, string theory, field theory, numerical simulation, black hole, thermodynamics, phase transition, electric field, gauge theory.*

Chemistry: *crystal structure, mass spectrometry, copper, aqueous solution, binding site, hydrogen bond, oxidant stress, free radical, liquid chromatography, organic compound.*

Biology: *genetics, enzyme, gene expression, polymorphism, nucleotides, dna sequence, saccharomyces cerevisiae, cell cycle, plasma membrane, embryonics.*

Appendix F. Categories for LINQS Datasets

Here we list the categories labels for the datasets obtained from LINQS (Sen et al., 2008)[1].

4. For AI dataset (6 classes):
   Agents, **AI**, **DB**, **IR**, **ML**, **HCI**.

5. For Cora dataset (7 classes):
   Case Based, **Genetic Algorithms**, **Neural Networks**, **Probabilistic Methods**, **Reinforcement Learning**, **Rule Learning**, **Theory**.

6. For PubMed dataset (3 classes):
   Diabetes Mellitus, Experimental,
   Diabetes Mellitus Type 1,
   Diabetes Mellitus Type 2.

Appendix G. Recovering Word Counts from TF-IDF

The PubMed dataset (Sen et al., 2008) was preprocessed to TF-IDF (term frequency-inverse document frequency) format, *i.e.* the raw word count information is lost. Here, we describe how we recover the word count information, using a simple and reasonable assumption.

We denote t_{dw} as the TF-IDF for word w in document d, f_{dw} as the corresponding term frequency (TF), and i_w as the inverse document frequency (IDF) for word w. Our aim is to recover the word counts c_{dw} given the TF-IDF. TF-IDF is computed[2] as

$$t_{dw} = f_{dw} \times i_w \quad , \qquad f_{dw} = \frac{c_{dw}}{\sum_w c_{dw}} \quad , \qquad i_w = \log \frac{\sum_d 1}{\sum_d I(c_{dw} > 0)} \quad , \tag{10}$$

where $I(\cdot)$ is the indicator function.

We note that $I(c_{dw} > 0) = I(t_{dw} > 0)$ since the TF-IDF for a word w is positive if and only if the corresponding word count is positive. This allows us to compute the IDF i_w easily from Equation 10. We can then determine the TF:

$$f_{dw} = t_{dw}/i_w \tag{11}$$

$$= t_{dw} \times \left(\log \frac{\sum_d 1}{\sum_d I(t_{dw} > 0)} \right)^{-1} \quad . \tag{12}$$

Now we are left with computing c_{dw} given the f_{dw}, however, we can obtain infinitely many solutions since we can always multiply c_{dw} by a constant and get the same f_{dw}. Luckily, since we are working with natural language, it is reasonable to assume that the least occurring words in a document only occur once, or mathematically,

$$c_{dw} = 1 \qquad \text{for} \quad w = \arg\min_w f_{dw} \quad . \tag{13}$$

1. http://linqs.cs.umd.edu/projects/projects/lbc/
2. Note that there are multiple ways to define a TF-IDF in practice. The specific TF-IDF formula used by the PubMed dataset was recovered *via* trial-and-error.

Thus we can work out the normaliser $\sum_w c_{dw}$ and recover the word counts for all words in all documents.

$$\sum_w c_{dw} = \frac{1}{\min_w f_{dw}} \quad , \qquad c_{dw} = f_{dw} \times \sum_w c_{dw} \quad . \tag{14}$$

Appendix H. Exclusion Words to Detect Incorrect Authors

society, university, universität, universitat, author, advisor, available, acknowledgement, mathematik, video, abstract, industrial, review, example, department, information, enterprises, informatik, laboratory, introduction, encyclopedia, algorithm, section

Appendix I. Estimating Topic Distributions for Test Documents

Here, we describe a simple method to estimate the topic distributions θ of the test documents, which is used for perplexity and clustering evaluation. We note that for perplexity evaluation, only the title of the test documents are used in estimating θ; while for the clustering task, both title and abstract text are used.

Denoting w_{dn} to represent the word at position n in a test document d, we *independently* estimate the topic assignment z_{dn} of word w_{dn} by sampling from its predictive posterior distribution given the learned author-topic distributions ν and topic-word distributions ϕ:

$$p(z_{dn}|w_{dn}, \nu, \phi) \propto \sum_k \nu_{a_d k}\, \phi_{k w_{dn}} \quad , \tag{15}$$

noting that the intermediate distributions are integrated out (see Appendix J).

We then build the customer counts $\boldsymbol{c}^{\theta_d}$ from the sampled z (for simplicity, we set the corresponding table counts as 1). With these, we can estimate the document-topic distribution:

$$\theta_d = \left(\cdots, \frac{(\alpha^{\theta_d} T^{\theta_d} + \beta^{\theta_d})\nu_{a_d k} + c_k^{\theta_d} - \alpha^{\theta_d} T_k^{\theta_d}}{\beta^{\theta_d} + C^{\theta_d}}, \cdots \right) \quad , \tag{16}$$

If citation network information is present, we refine the document-topic distribution θ_d using the linking topic y_{dj} for train document j where $x_{dj} = 1$. The linking topic y_{dj} is sampled from the estimated θ_d and is added to the customer counts, which further updates the document-topic distribution θ_d.

Doing the above gives a sample of the document-topic distribution $\theta_d^{(s)}$. We adopt a Monte Carlo approach by generating $R = 500$ samples of $\theta_d^{(s)}$, and calculate the Monte Carlo estimate of θ_d:

$$\theta_d^{MC} = \frac{\sum_s \theta_d^{(s)}}{R} \quad . \tag{17}$$

Appendix J. Integrating Out Probability Distributions

We note that in the paper, we have the following equation when calculating perplexity:

$$p(w_{dn}|\theta_d, \phi) = \sum_k p(w_{dn}|z_{dn} = k, \phi_k)\, p(z_{dn} = k|\theta_d) \tag{18}$$

$$= \sum_k \phi_{kw_{dn}} \theta_{dk} \quad , \tag{19}$$

where ϕ' is implicitly integrated out to arrive at Equation 19.

Here, we present the detailed derivation on how this works:

$$p(w_{dn}|z_{dn} = k, \phi_k) = \int_{\phi'_{dk}} p(w_{dn}, \phi'_{dk}|z_{dn}, \phi_k) \tag{20}$$

$$= \int_{\phi'_{dk}} p(w_{dn}|z_{dn}, \phi'_{dk})\, p(\phi'_{dk}|\phi_k) \tag{21}$$

$$= \int_{\phi'_{dk}} \phi'_{dkw_{dn}}\, p(\phi'_{dk}|\phi_k) \tag{22}$$

$$= \mathbb{E}[\phi'_{dkw_{dn}}|\phi_k] \tag{23}$$

$$= \phi_{kw_{dn}} \quad , \tag{24}$$

where $\mathbb{E}[\cdot]$ denotes the expectation value. We note that the last step (Equation 24) follows from the fact that the expected value of a PYP is the probability vector corresponding to the base distribution of the PYP[3].

Appendix K. Additional Results

K.1. Perplexity Comparison for AvS, AI, Cora and PubMed Datasets

	AvS		AI	
	Train	Test	Train	Test
Bursty HDP-LDA	2460.36 ± 66.38	2612.77 ± 91.70	1509.16 ± 4.09	1577.84 ± 33.81
Non-parametric ATM	2199.65 ± 5.02	2481.72 ± 6.09	N/A	N/A
CNTM w/o network	1621.50 ± 19.48	2079.42 ± 2.62	1509.35 ± 4.06	1580.16 ± 32.57
CNTM w network	**1620.55** ± 2.18	**2028.06** ± 10.87	**1275.27** ± 13.97	**1530.81** ± 49.81

Table 1: Train and Test Perplexity for AvS and AI Datasets.

We present additional results that were not shown in the paper in Table 1 and Table 2. Note that for AI, Cora and PubMed datasets, non-parametric ATM was not performed due to the lack of authorship information in the data. Additionally, we also note that the CNTM here is more akin to a variant of HDP-LDA (with network extension) when no author is observed, which explains why the perplexity results are very similar.

3. Note that this is only true when the base distribution is a probability distribution.

	Cora		PubMed	
	Train	Test	Train	Test
Bursty HDP-LDA	678.13 ± 1.95	706.81 ± 16.96	**299.86** ± 0.15	**300.10** ± 1.28
CNTM w/o network	**620.30** ± 3.37	**686.85** ± 16.63	300.98 ± 0.20	301.19 ± 1.23
CNTM w network	621.12 ± 6.65	687.95 ± 15.69	312.30 ± 1.26	303.21 ± 1.19

Table 2: Train and Test Perplexity for Cora and PubMed Datasets.

K.2. Clustering Results Correspond to Different Degree of Author-merging

	M10		AvS	
	Purity	NMI	Purity	NMI
$\eta = 2$	0.73 ± 0.04	0.74 ± 0.01	0.81 ± 0.01	0.75 ± 0.01
$\eta = 3$	0.75 ± 0.05	0.75 ± 0.02	0.82 ± 0.02	0.75 ± 0.00
$\eta = 4$	**0.78** ± 0.04	0.76 ± 0.02	0.82 ± 0.01	**0.76** ± 0.00
$\eta = 5$	**0.78** ± 0.02	**0.77** ± 0.03	**0.84** ± 0.02	**0.76** ± 0.02

Table 3: Clustering results by varying η from 2 to 5.

K.3. Topic Summaries for M10 and AvS Datasets

Topic	Top Words
DNA Sequencing	genes, gene, sequence, binding sites, dna
Agriculture	soil, water, content, soils, ground
Financial Market	volatility, market, models, risk, price
Bayesian Modelling	bayesian, methods, models, probabilistic, estimation
Quantum Theory	quantum, theory, quantum mechanics, classical, quantum field

Table 4: Topic Summary for M10 Dataset.

Topic	Top Words
Language Modelling	type, polymorphism, types, language, systems
Molecular Structure	copper, protein, model, water, structure
Quantum Theory	theory, quantum, model, quantum mechanics, systems
Social Science	research, development, countries, information, south africa
Woman's and Children's Health	children, health, research, social, women

Table 5: Topic Summary for AvS Dataset.

K.4. Convergence Analysis for Model Training

In Figure 1, we show the training word log likelihood $\sum_{d,n} \log(p(w_{dn}|z_{dn}, \phi'))$ for the CNTM trained with and without the network information. It is interesting to see that the word likelihood improves significantly once the network information is used in the model, which is after the 1000th iteration. Note that the model without network has better log likelihood initially simply due to random chance (even though we have used the same seed, due to having additional network component in full CNTM, the initialisation is different for the two models).

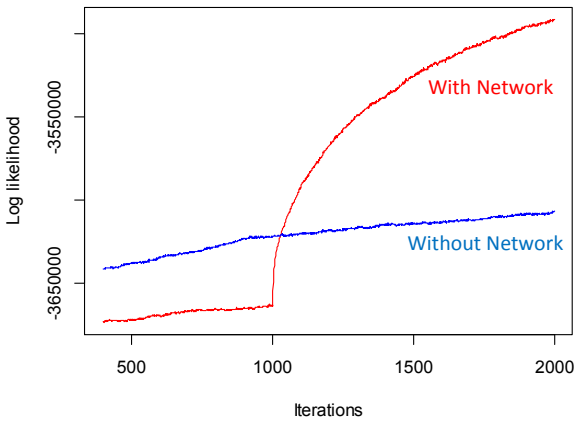

Figure 1: Words log likelihood vs iterations during training of the CNTM: the red line shows the log likelihood of the model with the citation network while the blue line represents the same model without citation network.

Appendix L. Computational Complexity for CNTM

Here we briefly discuss the computational complexity of the proposed MCMC algorithm for CNTM. We first note that we did not particularly optimise our implementation for algorithm speed. All implementations are written in `Java`[4].

For the Gibbs sampling algorithm of the hierarchical PYP topic model, we implemented a general Gibbs sampling framework that works with arbitrary PYP network, this allows us to test various PYP topic models with ease and without spending too much time in coding. However, having a general framework for PYP topic models means it is harder to optimise the implementation, thus it performs slower than existing implementations (such as `hca`[5]). The Gibbs sampling algorithm is linear (in time) with the number of words in the corpus and the number of topics, and constant time with the number of citations.

A naive implementation of the MH algorithm for the citation network would be of polynomial time, due to the calculation of the double summation in the posterior. However, with simple caching and reformulation of the double summation, we can evaluate the posterior in linear time. Our implementation of the MH algorithm is linear (in time) with the

4. http://java.com/
5. http://mloss.org/software/view/527/

number of citations and the number of topics, and it is constant time with respect to the number of words.

We present the average time taken to perform the MCMC algorithm for 2000 iterations in Table 6. All the experiments were performed with a machine having `Intel(R) Core(TM) i7 CPU @ 3.20GHz` (though only 1 processor was used) and 24 Gb RAM.

Datasets	Number of Words	Number of Citations	Number of Topics	Running Time (mins)
ML	8 270 084	1 105 462	20	16 194
M10	595 918	77 222	50	1 772
AvS	1 102 608	54 601	30	2 131
AI	105 322	4 608	6	63
Cora	49 286	5 429	7	32
PubMed	1 332 869	44 335	3	650

Table 6: Time taken by CNTM to run the learning algorithm for 2000 iterations.

Appendix M. Visualisation Results

We graphically visualise the author-topics network extracted by CNTM with `Graphviz`[6]. Defining the influence of an author i as the sum of the λ^- of all his publications:

$$\sum_d \lambda_d^- \, I(a_d = i) \quad , \tag{25}$$

noting that a_d denotes the author of document d as previously defined, we analyse the influential authors on the ML, M10, AvS, and similarly obtained NLP and IR datasets.

Figure 2 shows a snapshot of the visualisation result of the ML dataset. The visualisation result illustrates the connections between authors and topics. For example, we can see that T. Joachims works in the area of classification, support-vector machines and information retrieval; while M. Jordan works with probabilistic inference, classification and another topic that is not shown in Figure 2. For the full visualisation result and results on the other dataset, see https://drive.google.com/folderview?id=0B74l2KFRFZJmVXdmbkc3UlpUbzA. The results are in SVG and best viewed in Chrome or Firefox with magnification. We suggest to download the files instead of viewing with GoogleDoc for best quality.

6. http://www.graphviz.org/

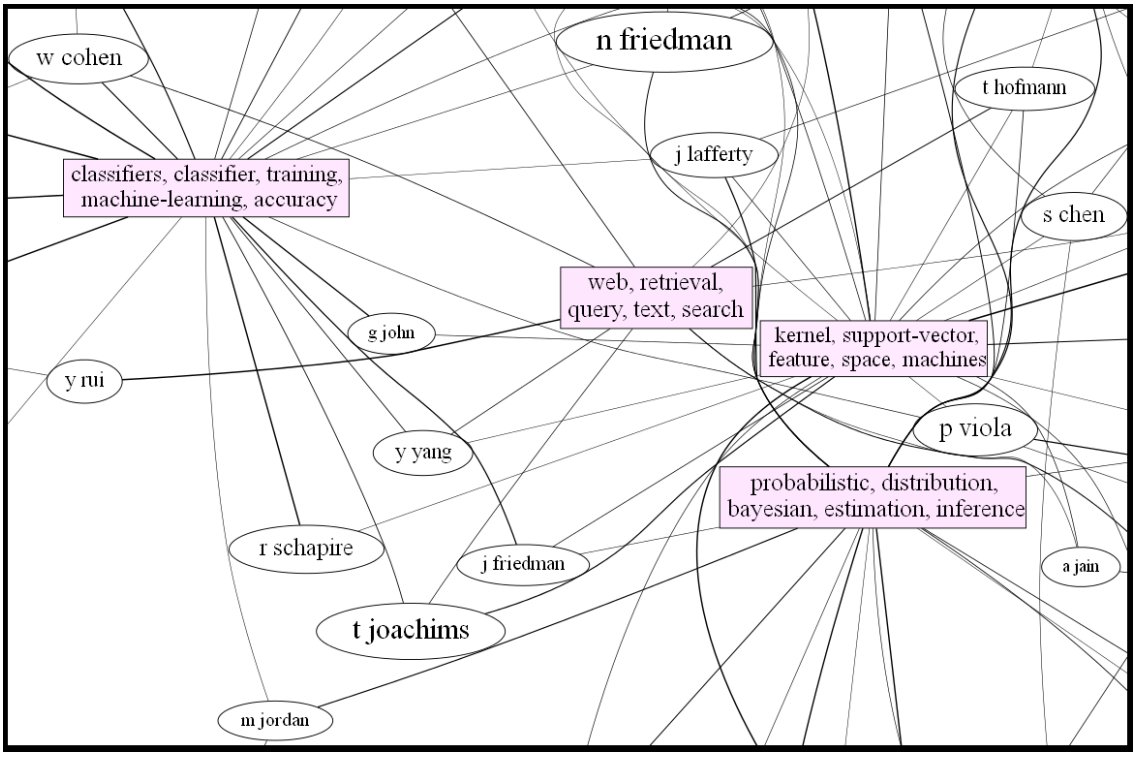

Figure 2: Visualisation snapshot of the ML Dataset. The pink rectangles are the topics learned by CNTM, the intensity (redness) and the size of the topics corresponds to the topic proportion. The ellipses denote the authors, the size of the ellipses correspond to the author's influence. The strength of the connections are given by the lines' thickness.

References

P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008.