Snippet Extraction: A Comparative Study

Abhishek Bafna abafna@umich.edu
Pragya Agrawal agpragya@umich.edu
Kritika Versha kmversha@umich.edu

1 Problem Statement and Motivation

In this report we aim to implement [2] using Labeled LDA to solve the problem of Credit Attribution in multi-labeled corpora.
A plethora of blogging, news and data websites like Reuters, del.icio.us, Flickr and Tumblr have widely useful articles labeled with multiple user-provided tags. However not every part of the document represents every tag equally. There are some parts of the document that focus on a particular tag more than the others. That is, there is no uniform Credit Attribution in a corpora. This poses an avenue to develop a better search system at an intra-document level. The users browsing the documents looking for a particular tag might prefer to look at the most relevant part of the document related to the tag. In other words they would prefer to look at a snippet of the document whose 'specificity' to that particular topic is higher than the rest of the document.
This immediately gives rise to an application resulting from an approached solution to the above described problem of Credit Attribution - Snippet Extraction. This would enable a user searching for a particular label to view relevant portions of the document(to his label) without having to scan the entire document, thus improving his efficiency and the better utilization of existing data.
The main method described in the paper [2], implemented here is an extention of the Latent Dirichlet Allocation (LDA) [3], Labeled-LDA. We have also implemented methods based on Hidden Markov Model and compared the results of L-LDA to it.

2 Related Work

Several modifications of LDA to incorporate supervision have been proposed in the literature. Two such models, Supervised LDA [8] and DiscLDA [9] are inappropriate for multiply labeled corpora because they limit a document to being associated with only a single label. Supervised LDA posits that a label is generated from each documents empirical topic mixture distribution. DiscLDA associates a single categorical label variable with each document and associates a topic mixture with each label. These models fall short as a solution to the credit attribution problem.

3 Notation

Throughout this paper, unless otherwise mentioned, the notation used for Labeled LDA is as follows.
Let each document d be represented by a tuple consisting of a list of word indices $\mathbf{w}^{(d)} = (w_1, \ldots, w_{N_d})$
and a list of binary topic indicators $\Lambda^{(d)} = (l_1, \ldots, l_K)$
where each $w_i \in \{1, \ldots, V\}$
and each $l_k \in \{0, 1\}$.
Here the document length is N_d,
the vocabulary size V
and K is the total number of unique labels in the corpus.
For the purpose of LLDA, K is also the number of latent topics in the corpus.

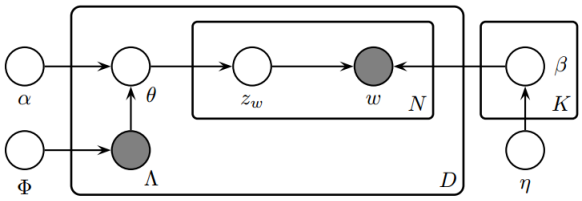

Figure 1: Graphical Model of Labeled LDA

The notation originating from LDA and being extended here is as follows:
K is the number of latent topics in the collection,
$\phi^{(k)}$ is a discrete probability distribution over the k^{th} topic over the distribution,
$\theta^{(d)}$ is a document-specific distribution over the available topics,
z_i is the topic index for word w_i,
and α and β are hyperparameters for the symmetric Dirichlet distributions that the discrete distributions are drawn from.

4 Methodologies explored

The methodologies explored here are the application of Labeled-LDA and HMM to the corpora for snippet-extraction.
This implementation of Labeled LDA is based on Gibb's Sampling. A brief introduction to LDA is provided as follows.

4.1 LDA

LDA is a generative probabilistic model for a collection of documents. It assumes a latent structure of topics where each document has a distribution over the set of topics and each topic has a discrete distribution over the vocabulary of the document collection or corpus. The generative process for the generation of words in the documents is as follows:

1. $\phi^k \sim Dirichlet(\beta)$ for $k = 1, \ldots, K$
2. $\theta^d \sim Dirichlet(\alpha)$ for $d = 1, \ldots, D$
3.
4. $z_i \sim Discrete(\theta^{(d)})$
5. $w_i \sim Discrete(\phi^{(z_i)})$

Since LDA is unsupervised, it is not appropriate for multi-labeled corpora. For a more detailed study of LDA and its implementation using Gibb's sampling please refer [5].

4.2 Labeled LDA

Here, as mentioned above, the number of topics is set to be the number of unique labels K in the corpus. Here, differing from the LDA model, we restrict $\theta^{(d)}$ to be defined only over the labels of the document, $\Lambda^{(d)}$.
In the generation process, we form the document's labels $\Lambda^{(d)}$ using a Bernoulli sample for each topic k. Now, $\lambda^{(d)} = \{k|\Lambda_k^{(d)} = 1\}$. Next we can define a label projection matrix $L^{(d)}$ for each document d of size $M_d X K$ where $M_d = |\lambda^{(d)}|$ such that,

$$L_{ij}^{(d)} = \mathbf{1}_{\lambda_i^{(d)}=j}$$
$$\alpha^{(d)} = L^{(d)} \text{ X } \alpha,$$

For each topic $k \in \{1, \ldots, K\}$:
  Generate $\boldsymbol{\beta}_k = (\beta_{k,1}, \ldots, \beta_{k,V})^T \sim \text{Dir}(\cdot|\boldsymbol{\eta})$
For each document d:
  For each topic $k \in \{1, \ldots, K\}$
    Generate $\Lambda_k^{(d)} \in \{0, 1\} \sim \text{Bernoulli}(\cdot|\Phi_k)$
  Generate $\boldsymbol{\alpha}^{(d)} = L^{(d)} \times \boldsymbol{\alpha}$
  Generate $\boldsymbol{\theta}^{(d)} = (\theta_{l_1}, \ldots, \theta_{l_{M_d}})^T \sim \text{Dir}(\cdot|\boldsymbol{\alpha}^{(d)})$
  For each i in $\{1, \ldots, N_d\}$:
    Generate $z_i \in \{\lambda_1^{(d)}, \ldots, \lambda_{M_d}^{(d)}\} \sim \text{Mult}(\cdot|\boldsymbol{\theta}^{(d)})$
    Generate $w_i \in \{1, \ldots, V\} \sim \text{Mult}(\cdot|\boldsymbol{\beta}_{z_i})$

Figure 2: Generative process for Labeled LDA: β_k is a vector consisting of the parameters of the multinomial distribution corresponding to the k^{th}topic, α are the parameters of the Dirichlet topic prior and η are the parameters of the word prior, while ϕ_k is the label prior for topic k

where $\alpha^{(d)}$ is a lower dimensional Dirichlet topic prior.
Finally, from [1] we have,

$$P(z_i = j|\mathbf{z}_{-i}) \propto \frac{n^{w_i}_{-i,j}+\eta_{w_i}}{n^{(\cdot)}_{-i,j}+\eta^T\mathbf{1}} \text{ X } \frac{n^{d}_{-i,j}+\alpha_j}{n^{(\cdot)}_{-i,j}+\alpha^T\mathbf{1}}$$

[2]

4.3 Hidden Markov Models

One promising approach to pattern recognition based snippet extraction is using Hidden Markov Models(HMM). Given a document and a query for snippet extraction related to query from the document, HMM models each document as an observable symbol chain. Behind every symbol chain lies the hidden state chain, which reveals whether an observed word is relevant to query or not. According to our problem Hidden Markov chain has two states associated, 'relevant'(r) and 'irrelevant'$\tilde{r}$ state. We can expect to extract a relevant snippet from the document if the Markov chain starts with the irrelevant state $\tilde{r_1}$, transits to the relevant state(r) to generate the snippet and then returns to the irrelevant state $\tilde{r_2}$ before it terminates.

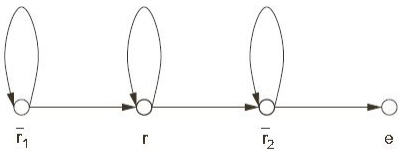

Figure 3: Hidden State Transitions in HMM

We denote all possible state transitions that follow the pattern in figure by τ. In addition, for any T $\in \tau$, we use P(T|d) to denote the probability that the transition is in fact T given the observation of document d. Thus,we are interested in the transition in τ that maximizes such a probability,

$$T^* = \arg max_{T \in \tau} P(d|T) * P(T)$$

Assuming document d can be represented using a first-order Markov Chain, and the words in sequence are independent of each other, we can further re-write the above equation as,

$T^* = \arg max_{T \in \tau} P(T) * \prod_{i=1}^{n} P(w_i|s_i)$

$= \arg max_{T \in \tau} P(s_1) * \prod_{i=1}^{n} -1P(s_{i+1}|s_i) * \prod_{i=1}^{n} P(w_i|s_i)$

where $s_1, s_2, ..., s_n$ is a sequence of states that the Hidden Markov chain has visited when generating d. $P(s_{i+1}|s_i)$ is the state transition probability and $P(w_i|s_i)$ the output probability. The above Maximisation problem can be solved using the Viterbi algorithm [6]. However to apply Viterbi algorithm parameters of HMM, the state transition probabilities and output probabilities, need to be calculated using the Baum-Welch Algorithm [6].

In order to apply snippet extraction, we find the word generation probability of the relevance language model M_r and the irrelevance language model $M_{\tilde{r}}$. $P(w|M_{\tilde{r}})$ can be seen as the word generation probability of w not relevant to query, and is given as

$$P(w|M_{\tilde{r}}) = \frac{f_w}{F}$$

where f_w is number of times word w occurs in the whole corpus and F is the total number of words in corpus.
To find $P(w|M_r)$, word generation probability of w relevant to query, following procedure is adopted,

- For query q form a subset of corpus D_r with documents relevant to query q. In our application we use a multi-labeled corpus with each document from corpus having 4 or more tags. A query for snippet extraction is one of the 'labels' on the document. Hence D_r is formed from all documents having the label q on them, such that $P(d|q)$ is the probability of observing a document d given label q

  $$P(d|q) = 1/(\text{number of times label 'q' occurs in whole corpus})$$

- Calculate the probability $P(w|d)$ of word being generated by a given document using a maximum likelihood estimate,

  $$P(w|d) = \lambda \frac{f_{w,d}}{|d|} + (1 - \lambda) P(w|M_{\tilde{r}})$$

  where $f_{w,d}$ is the number of times that w occurs in d and $|d|$ is the number of tokens in d.

- Calculate $P(w|D_r)$, the probability that the given word is generated by the relevance language model,

  $$P(w|D_r) = \sum_{d \in D_r} P(w|d) P(d|q)$$

After the word generation probabilities of relevant and irrelevant language model is calculated we can apply HMM to extract snippets from the document as follows:
Input : Document d, query q **Output**: relevant snippet S

1. for each word $w \in d$ do

   Find $P(w|M_r)$
   Find $P(w|M_{\tilde{r}})$

2. Initialise State Transition Probability Matrix, T
3. Initialise Emission Probability Matrix, E
4. Estimate HMM parameters, $HMM.baumwelch$
5. Extract relevant snippet, $S = HMM.viterbi$
6. Return S

5 Experiments, Results and Evaluation

5.1 Dataset

We initially planned to use the **"del.icio.us"** dataset [4]. Upon examination it was discovered that documents in the corpus were extracted from different websites and hence the text could not be effectively extracted excluding the comments and advertisements. Then another dataset called **Wiki10+** was used[7]. The final datasubset used to fit the model was created and processed as follows:

Table 1: Sample Topic Distributions Inferred from the Model. Each column represents the highest probable words in the distribution of the topic (Header)

motor	**music**	**scifi**	**immune**	**soup**
electric(0.128)	album(0.046)	fantasy(0.082)	fungi(0.077)	stew(0.052)
magnetic(0.115)	songs(0.041)	novels(0.058)	insect(0.058)	hungarian(0.052)
motor(0.102)	during(0.029)	fiction(0.058)	biological(0.038)	made(0.044)
current(0.051)	released(0.029)	science(0.058)	fungus(0.038)	potatoes(0.032)
lines(0.051)	featured(0.027)	novel(0.043)	stroma(0.038)	peppers(0.028)
along(0.038)	rock(0.024)	names(0.034)	anamorphs(0.019)	popular(0.02)
rotation(0.038)	american(0.024)	game(0.024)	plant(0.019)	pasta(0.02)
tangential(0.038)	palace(0.024)	magic(0.024)	elongated(0.019)	served(0.02)
battery(0.026)	videos(0.022)	wrote(0.019)	thailand(0.019)	simmered(0.016)
perpendicular(0.026)	show(0.022)	readers(0.019)	humid(0.019)	paprikas(0.016)
parallel(0.026)	performed(0.019)	cultures(0.014)	pharmacological(0.019)	vegetables(0.016)

1. Using Regular Expressions and a Stop List, the documents and labels were parsed rejecting the non alphabetic characters.
2. Based on the frequency of labels in the entire corpus, the following specific set of labels, *flabels* were selected to serve as the common label pool: 'history', 'science', 'research', 'programming', 'philosophy', 'people', 'culture', 'software', 'politics', 'art', 'web', 'language', 'design', 'music', 'interesting', 'psychology', 'technology', 'books', 'math', 'theory', 'article', 'religion', 'development', 'literature', 'computer', 'health', 'business', 'economics', 'education', 'mathematics'. This was used to get a good number of documents having varied but popular topics.
3. Documents were chosen such that every document had at most 4 labels in common with the *flabels*. This was based on the work done in [2] and also to avoid a higher degree of similarity between documents
4. Finally, a set of 150 documents, 100 for training and 50 for testing was used for the LLDA model.
5. The total labels in the corpus was 371
6. The word length of the corpus was 18117

5.2 Perplexity

Perplexity is defined over a held-out test set D_{test} as

$$\text{perplexity}(D_{test}) = exp\left\{-\frac{\sum_{d=1}^{M} logp(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d}\right\}$$

In our experiment, we trained the model on 100 documents and tested it on 50 documents, calculating the perplexity for each iteration. Note that perplexity is primarily used for estimation of number of topics. In our case this is not possible since the number of topics, a.k.a the labels are already determined. In general, a lower perplexity score indicates better generalization performance [3].

5.3 Snippet Extraction from LDA

The following were the 30-word snippets extracted from various documents of diverse topics:

1. In a Wikipedia article on movie releases at the box office, the snapshot for label: 'movies' was
   'the' 'half' 'billion' 'dollar' 'domestic' 'milestone' 'well' 'the' 'only' 'film' 'the' 'decade' 'for' 'the' 'top' 'ten' 'films' 'consisted' 'superhero' 'films' 'animated' 'films' 'action' 'films' 'and' 'musical' 'film' 'mamma' 'mia' 'became'

In the document this corresponded to :
On August 31, after 45 days in release, The Dark Knight reached $500 million domestically, becoming only the second film in history after Titanic to cross the half-billion-dollar domestic milestone, as well as the only film of the 2000s decade. For 2008, the top ten films consisted of 3 superhero films, 3 animated films, 3 action films and 1 musical film. Mamma Mia! became the highest grossing film in UK history.

Evaluation: From reading of the document two snippets were expected for that label. They were:

(a) The Dark Knight has grossed more than $1 billion, making it the 4th highest grossing film in history. On August 4, The Dark Knight reached a $400 million domestic gross in a record time of 18 days. The previous record was held by Shrek 2, which reached it in 43 days.

(b) The Dark Knight reached $500 million domestically, becoming only the second film in history after Titanic to cross the half-billion-dollar domestic milestone, as well as the only film of the 2000s decade. For 2008, the top ten films consisted of 3 superhero films, 3 animated films, 3 action films and 1 musical film. Mamma Mia! became the highest grossing film in UK history.

Hence, a useful snippet was extracted in this case.

2. In the article on tree data structure,

(a) The snapshot for tree was:
'the' 'nodes' 'below' 'comprise' 'subtree' 'the' 'subtree' 'corresponding' 'the' 'root' 'node' 'the' 'entire' 'tree' 'the' 'subtree' 'corresponding' 'any' 'other' 'node' 'called' 'proper' 'subtree' 'analogy' 'the' 'term' 'proper' 'subset' 'there' 'are' and from the HMM was
'one' 'parent' 'nodes' 'the' 'bottommost' 'level' 'the' 'tree' 'are' 'called' 'leaf' 'nodes' 'since' 'they' 'are' 'the' 'bottommost'
In the document this corresponded to:
Any node in a tree T, together with all the nodes below it, comprise a subtree of T. The subtree corresponding to the root node is the entire tree; the subtree corresponding to any other node is called is a proper subtree (in analogy to the term proper subset).There are two basic types of trees.
Evaluation: Both formed a part of the definition of a subtree and not specifically the tree, hence better snippets can be extracted.

(b) The snapshot for algorithms was:
are' 'far' 'the' 'most' 'common' 'form' 'tree' 'data' 'structure' 'binary' 'trees' 'are' 'one' 'kind' 'ordered' 'tree' 'because' 'the' 'children' 'are' 'ordered' 'left' 'child' 'node' 'and' 'right' 'child' 'node' 'there' 'are'
In the document this corresponded to:
Ordered trees are by far the most common form of tree data structure. Binary trees are one kind of ordered tree because the children are ordered as left child node and right child node.There are many different ways to represent trees.
Evaluation: This formed a part of the definition of subtree, hence can be acceptable.

3. In an article on vector derivatives,

(a) The snapshot for `gradient` was:
'given' 'location' 'will' 'vector' 'the' 'plane' 'sort' 'like' 'arrow' 'map' 'pointing' 'along' 'the' 'steepest' 'direction' 'the' 'magnitude' 'the' 'gradient' 'the' 'value' 'this' 'steepest' 'slope' 'particular' 'this' 'notation' 'powerful' 'because' 'the' and from HMM
'coordinate' 'operator' 'scalar' 'operator' 'that' 'can' 'applied' 'either' 'vector' 'scalar' 'fields' 'defined' **Evaluation:** This formed a part of the definition of gradient, hence can be acceptable. The snippet from HMM doesn't correspond precisely to the gradient and hence can be better.

(b) The snapshot for divergence was:
'element' 'parentheses' 'can' 'considered' 'single' 'coherent' 'unit' 'fluid' 'dynamics' 'uses' 'this' 'convention' 'extensively' 'terming' 'the' 'convective' 'derivative' 'the'

'moving' 'derivative' 'the' 'fluid' 'the' 'laplace' 'operator' 'scalar' 'operator' 'that' 'can' 'applied'
Evaluation: This is incorrect, because the snippet corresponds to Laplacian and directional derivative but not divergence. This makes it a bad result. The snippet from HMM in this case was the same as before making it a bad result too.

6 Conclusions

1. Labeled LDA was successfully implemented on the Wiki10+ dataset with satisfying results in terms of snippet extraction and topic distributions.
2. Though training the LLDA is very computationally and time intensive, it outperforms the HMM model in snippet extraction.
3. Fitting pre-existing labels to the posterior does not always yield good results as in some cases it overfits the model.
4. There were a couple of shortcomings in the implementation as per the paper being referred. Firstly in the creation of the final training set, according to the paper, the labels of the document other than the labels from the common pool, *flabels* were stripped off from the document. This did not give good results and the justification is that if the same labels exist across many documents then they are not sampled specifically and tend not to represent the correct set of words.

References

[1] Griffiths, Tom. "Gibbs sampling in the generative model of latent dirichlet allocation." (2002).

[2] Ramage, Daniel, et al. "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora." Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1. Association for Computational Linguistics, 2009.

[3] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.

[4] Olaf Grlitz, Sergej Sizov, Steffen Staab: PINTS: Peer-to-Peer Infrastructure for Tagging Systems. 2008. Tampa Bay, USA. 2. Proceedings of the Seventh International Workshop on Peer-to-Peer Systems, IPTPS.

[5] Darling, William M. "A theoretical and practical implementation tutorial on topic modeling and gibbs sampling." Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011.

[6] Rabiner, Lawrence. "A tutorial on hidden Markov models and selected applications in speech recognition." Proceedings of the IEEE 77.2 (1989): 257-286.

[7] Zubiaga, Arkaitz. "Enhancing navigation on wikipedia with social tags." arXiv preprint arXiv:1202.5469 (2012).

[8] Mcauliffe, Jon D., and David M. Blei. "Supervised topic models." Advances in neural information processing systems. 2008.

[9] Lacoste-Julien, Simon, Fei Sha, and Michael I. Jordan. "DiscLDA: Discriminative learning for dimensionality reduction and classification." Advances in neural information processing systems. 2009.