PWA-PEM for Latent Tree Model and Hierarchical Topic Detection

Zhuchen Liu, Hao Chen, Jie Li, Yanhua Yu

PWA-PEM for Latent Tree Model and Hierarchical Topic Detection

Liu Zhuchen[1], Chen Hao[2], Li Jie[1], Yu Yanhua[1]

[1] Beijing University of Posts and Telecommunications, Beijing, China
{liuzhuchen, yuyanhua}@bupt.edu.cn
[2] China Institute of Marine Industrial Systems Engineering, Beijing, China

Abstract. Hierarchical Latent Tree Analysis (HLTA) is a new method of topic detection. However, HLTA data input uses TF-IDF selection term, and relies on EM algorithm for parameter estimation. To solve this problem, a method of accelerating part of speech weight (PWA-PEM-HLTA) is proposed based on Progressive EM-HLTA (PEM-HLTA). Experimental results show that this method improves the execution efficiency of PEM-HLTA, averaging 4.9 times speed, and improves the speed of 6 times in the best case.

Keywords: Hierarchical Latent Tree Analysis, topic detection, Aitken acceleration, PEM

1 Introduction

Latent Tree Analysis (LTA) attempts to describe the correlation between a set of observed variables using a tree model called Latent Tree Model (LTM) [1, 2]. In the model, leaf nodes represent observation variables, internal nodes represent latent variables, and the dependencies between two observation variables are explained by the paths between them. In recent years, LTA model has been widely used in academic research, and put forward some effective new ideas, such as cluster analysis [3,4], topic detection [5], depth probability modelling [6] and so on. Among them, the text data in topic detection applications show the best results. Liu et al. used the word co-occurrence matrix to model the words in the text collection and soft-partitioned the document [5]. The result was that each document might belong to a different partition, and the collection of documents in the partition was interpreted as a topic. In addition, LTM divides the learned latent variables into multiple levels. This led to another approach to hierarchical topic detection, Hierarchical Latent Tree Analysis (HLTA). It proved to be the most advanced methods, themes and better looking than before on the topic hierarchy latent dirichlet allocation based on the most advanced methods [7].

HLTA depends on the EM algorithm to estimate parameters, so there is still some room for improvement in efficiency. Chen et al. uses progressive EM (PEM) to improve the HLTA intermediate model parameter estimation [8]. In each step, PEM only calculates the maximum local likelihood function of the submodels in the model. That is, EM is running on a model that only involves 3 or 4 observation variables. The

improvement of PEM is performed on the E-step of EM algorithm. This paper adopts the method of gradient acceleration optimization and improves it from M-step, and thereby further enhance the computational efficiency of PEM algorithm.

2 Appearance

The LTM is a tree-structured Bayesian network in which leaf nodes represent observational variables and internal nodes represent latent variables[3, 9]. In general, the LTM has n observation variables $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$ and m latent variables $\mathbf{Z} = \{z_1, z_2, \dots, z_m\}$. The parent value of the variable Y is represented as $pa(Y)$, and Y is set as the root and $pa(Y)$ is empty. LTM defines the joint distribution of all observations and latent variables $p(x_1, \dots, x_n, z_1, \dots, z_m) = \prod_{Y \in \boldsymbol{X} \cup \boldsymbol{Z}} P\big(Y \big| pa(Y)\big)$.

Liu et al. proposed a method for analyzing text data and obtaining models based on LTM[5]. At the bottom layer of observed variable representative of a variable in binary form in the presence or absence of the document. At the bottom of the top layers have a plurality of latent variables, each of the lower probability variable representing a word co-occurrence used to explain the relationship between the word co-occurrence. Therefore, the theme of the model obtained at a low level has a specific meaning, and the theme captured at a high level has a more abstract meaning.

2.1 Pretreatment

Prior to analysis items selected n words having the highest TF-IDF values mean average TF-IDF method[5, 8, 10]. For a document set D, the term t-average $\mathrm{TF-IDF}(t, D) = \frac{\sum_{d \in D} tf(t,d) \cdot idf(t,D)}{|D|}$. Where $|D|$ represents the total number of files in the corpus and $tf(t, d)$ is the frequency in item t document d. $idf(t, D) = log(|D| / |\{d \in D : t \in d\}|)$ is the inverse document frequency of the term t in document set D. The traditional TF-IDF thinks that the terms are mutually independent, but in each document expression, combining the current situation, context, and semantics, the terms are related to each other. In order to make up for the word term mutual information calculation when the subject word is extracted, usually only the word frequency is considered. This paper uses the part of speech and the traditional TF-IDF (Pos Weight TF-IDF, PW_TF-IDF) [11] to calculate the term of the document. The PW_TF-IDF value attempts to optimize the term selection to improve subject consistency.

2.2 PEM

The EM algorithm is one of the statistical algorithms often used for parameter estimation problems. In a latent tree model m, let **X** and **H** represent the set of observation variables and latent variables, respectively, $\mathbf{V} = \mathbf{X} \cup \mathbf{H}$. Assume that a latent variable is selected as the root and all edges are far from the root. For any variable v that is not root in **V**, $\mathrm{pa}(v)$ for v is a latent variable that takes a value of "0" or "1". When v is

the root, pa(v) is a virtual variable with only one possible value. List all the variables $v_1, v_2, \ldots, v_{\mathrm{n}}$. The parameter of m is $\theta_{ijk} = P(v_i = k|\mathrm{pa}(v_i) = j)$.

Where i ∈ {1, … , n}, k is the value of v_i , and j is the value of pa(v_{i}). θ is a vector of all parameters. For a given data set D, the log-likelihood function θ is given by $l(\theta|D) = \sum_{d\in D}\sum_{\mathbf{H}} logP(d, \mathbf{H}|\theta)$. The maximum likelihood estimate θ is the value of the maximum log-likelihood function. Start estimating the parameter value of $\theta^{(0)}$, and then generate a sequence of estimates $\{\theta^{(1)}, \theta^{(2)}, \ldots\}$. Assuming the current estimate $\theta^{(\mathrm{t})}$, the next estimate $\theta^{(\mathrm{t+1})}$ is obtained through the E step and the ME step. For the latent tree model, the two steps of the EM algorithm are as follows:

$$\mathrm{n}_{ijk}^{(\mathrm{t})} = \sum_{d\in D} P(v_i = k, pa(v_i) = j|d, m, \theta^t) \tag{1}$$

$$\theta_{ijk}^{t+1} = \frac{n_{ijk}^t}{\sum_k n_{ijk}^t} \tag{2}$$

The PEM calculation submodel is shown in Figure 1. Supposing that Y is selected as the root and all parameters of the model are estimated. Firstly, running the EM model shaded in Figure 1(a), estimate P(Y), P(A|Y), P(B|Y), and P(D|Y), then running the EM model in Figure 1(b) of the shaded part; fix P(Y), P(B|Y) and P(D|Y) to estimate P(Z|Y), P(C|Z) and P(E|Z). The shortage of the EM algorithm is that the computation complexity is large and the convergence speed is slow when the data set is relatively large. Various methods for accelerating EM algorithms have been proposed, such as incremental EM algorithm, lazy EM algorithm, and hybrid EM algorithm. Chen et al. PEM algorithm [8] computational complexity improvement mainly in the E-step and the M-step is not considered, and some acceleration gradient M-step process optimization, an improved method of Step E in combination can further improve the computational efficiency of the EM algorithm.

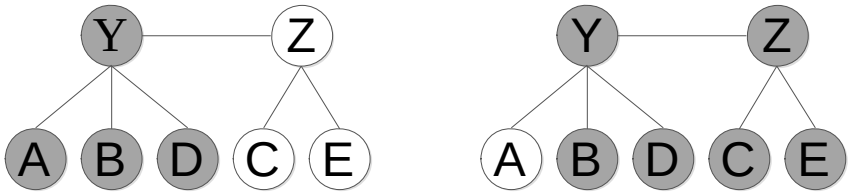


Fig. 1. PEM submodel

3 Research methods

3.1 Word selection based on PW_TF-IDF

There are two main aspects of keyword selection, word weight and theme model selection. This article adds word-based information to words based on word frequency. To select a more suitable word, proceed to the following topic model. Han et al. studied the contribution of different part-of-speech features in texts, and verified the influ-

ence of nouns, verbs, adjectives and adverbs, and their combinations on Chinese and English texts [12]. The experimental results show that these four parts of speech are important part of speech characterizing the content of the text. This paper will statistically count the percentages of nouns, verbs, adjectives and adverbs after word segmentation, and give different part-of-speech weight coefficients to these four parts of speech. Other parts of speech are still calculated according to the traditional TF-IDF. $PW\_TF-IDF = k * TF-IDF$. Where TF-IDF is the value obtained by the traditional calculation method. The coefficient k is the weight coefficient of the four parts of speech. Through the random sampling of 1000 documents in the nips and Reuters data sets, the percentages of nouns, verbs, adjectives and adverbs are obtained as part of speech. The weight coefficients are shown in Table 1.

Table 1. weight coefficient of part of speech

	n	v	adj	adv
Coefficient k	1.6198	1.2919	1.0382	1.0501

3.2 Improved Aitken Accelerated PEM

The Aitken acceleration method is based on the iterative function of the simple iterative method to construct a new iterative function. Theorem [13]: Let the sequence $\{p_n | n \in [0,\infty)\}$ converge linearly to the limit p with $p - p_n \neq 0$. Satisfaction: $\lim_{n\to\infty} \frac{p-p_{n+1}}{p-p_n} = A(|A| < 1)$, then define the sequence $\{r_n | n \in [0,\infty)\}$ Convergence to p, and faster than the sequence $\{p_n\}$, the result is closer to the true value of p, That is $\lim_{n\to\infty} \frac{p-r_n}{p-p_n} = 0$. It defines the sequence formula: $r_n = p_n - \frac{(p_{n+1}-p_n)^2}{2p_{n+2}-3p_{n+1}+p_n}$. The above method is applied to the log-likelihood sequence $\{\theta\}$ in the PEM algorithm, and Aitken acceleration is performed using the above theorem, which is applied to each sub-model of the latent tree model.

```
Algorithm pseudo code: APEM
input: Data set D, maximum number of iterations N,
threshold condition δ.
output: Log-likelihood estimation.
begin
  set k = 1, a0 = 0, a1 = 0
  Repeat
     n_ijk^(t) := Σ_{d∈D} P(Vi = k, pa(Vi) = j | d, m, θ^t)
     θ_ijk^{t+1} := n_ijk^t / Σ_k n_ijk^t
     a2 := θ_ijk^{t+1}
     θ_ijk^{t+1} := a0 − (a1 − a0)^2 / (2a2 − 3a1 + a0)
     a0 := a1, a1 = a2, k = k + 1
  Until k>N or a1 − a0 < δ
end.
```

4 Experimental results

It seeks to optimize PEM-HLTA; therefore, the optimization method mentioned in this text should be compared with PEM-HLTA. On the other hand, it is not necessary to compare with the methods of LDA, HLDA or nHDP because the PEM-HLTA has been proved valid in the literature [5, 8].
Experimental environment: Windows 7 64/cpu i5 3.2Ghz/Ram 12G/java 1.8. All experimental parameters were the same as that of reference [8].

4.1 Data Sources

NIPS[1] data set and Reuters[2] data set adopted in experiment are different from the Liu et al.[5] and Chen et al.[8] NIPS data set, we use the data set NIPS from Kaggle, from the 1987 meeting of the current session in 2016, has 6560 documents. The NIPS data is divided into two experiments. The experiment selects 1955 documents in the same way as document Chen et al. [8]. Experiment 2 uses all documents. Each experiment uses TF-IDF values and PW_TF-IDF values to select vocabulary sizes 1000, 3000, 5000, 7000, and 10000 in five versions, using Nips-1k, Nips-3k, Nips-5k, Nips-7k, and Nips-10k indicates. Two sets of NIPS data were compared using PWA-PEM-HLTA[3] and PEM-HLTA[4] after pretreatment. Experiment 3 uses exactly the same configuration as Experiment 1. The only difference is the use of the Reuters data set to verify that the method has the same effect on different types of data sets.

4.2 Conformity assessment method

The score of topic semantic coherence was calculated using the [14] method. Subject t's theme consistency score is defined as:

$$C\left(t, W^{(t)}\right) = \sum_{m=2}^{M} \sum_{t=1}^{m-1} log \frac{D\left(w_m^{(t)}, w_l^{(t)}\right)+1}{D\left(w_l^{(t)}\right)} \quad (3)$$

Where $\boldsymbol{W}^{(t)} = \{w_1^{(t)}, \ldots, w_m^{(t)}\}$ is the first m words for describing the subject t. $D(w_i)$ is the document frequency of word w_i. $D(w_i, w_j)$ is the common document frequency of words w_i and w. The document frequency is the number of documents containing these words. Given two sets of topics, topics with higher average theme coherence are considered better topics.

[1] https://www.kaggle.com/benhamner/nips-papers/downloads/papers.csv/1
[2] http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html
[3] https://pan.baidu.com/s/1ZWnD-1PT1agJFfsKrOCaYw : byaa
[4] https://github.com/kmpoon/hlta/tree/v2.0

4.3 Experiment 1

Table 2 shows the runtime statistics. The improved method is obviously better than the PEM-HLTA method, and the average improvement efficiency is around 5 times. The efficiency increase rate is shown in Table 3. The average execution efficiency is increased by 4.9 times. In the best case, the execution efficiency is increased by 6 times. From Figure 2(a), it can be intuitively found that the A-PEM-HLTA method and the PWA-PEM-HLTA method have no explicit difference before the Nips-10k. When the number of words reaches 10K, the PWA-PEM-HLTA is optimal and used 58 minutes, while the PEM-HLTA method used 431 minutes. At the same time, it is found that comparing the use of part-of-speech weights and not using part-of-speech weights, the use of part-of-speech weights may lead to the further extraction of words that are closer to the subject, reducing the number of PEM iterations and improving EM implementation efficiency.

Table 2. nips-1955runtime/min

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	5	30	98	165	431
PW-PEM-HLTA	5	32	93	145	329
A-PEM-HLTA	1	5	14	26	67
PWA-PEM-HLTA	1	6	14	26	58

Table 3. nips-1955 comparison of improvement multiples(multiple=pre/post improved -1)

Comparison	Nisp-1k	Nisp-3k	Nisp-5k	Nisp-7k	Nisp-10k	Average
PEM-HLTA/ A-PEM-HLTA	4.0	5.0	**6.0**	5.3	5.4	5.16
PW-PEM-HLTA/ PWA-PEM-HLTA	4.0	4.3	5.6	4.6	4.7	4.65
Average	4.0	4.7	5.8	5.0	5.1	**4.90**

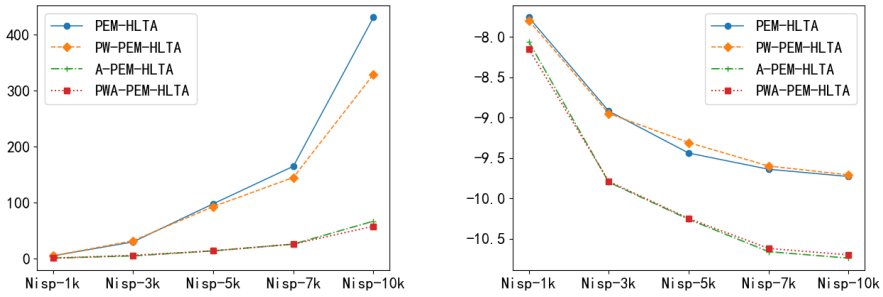
Table 4 shows the average topical consistency score for the topic generated by the improved algorithm. PW-PEM-HLTA (optimization of POS) and PEM-HLTA, A-PEM-HLTA (Aitken acceleration optimization) and PWA-PEM-HLTA (Attenuation Optimization of POS) show that nips-5k is a watershed. When the word exceeds 5k, the participatory weights have some advantages in the choice of terms. When the number of selected terms is small, the top ten words are covered by the TF-IDF value. When the range of selected words is expanded, when the middle and latter parts of all words are selected, the advantages of the word weight selection terms are reflected. When PEM-HLTA was compared with A-PEM-HLTA, PW-PEM-HLTA and PWA-PEM-HLTA using accelerated optimization, the average subject consistency score was significantly decreased, and the average score reduction was around 0.81 ± 0.2. In Figure 2(b), the PW-PEM-HLTA consistency score is best when the number of words reaches 5K. However, when the choice of terms increases, most of the terms are selected, and the result scores converge with the PEM-HLTA. While the overall trend of the average topic consistency scores in the A-PEM-HLTA and PWA-PEM-HLTA

methods is consistent. When the words exceed 5K, the latter has a slight improvement over the former. At the same time, the overall consistency score after using Aitken acceleration optimization can be reduced. This is because Aitken acceleration adopts a simple iterative method and oscillates around the convergence value, which does not guarantee stable growth of the EM likelihood result.

Planned solution: Using the original M algorithm or ECM algorithm when oscillating near the convergence value so that the likelihood result can grow steadily again.

Table 4. nips-1955 average thematic consistency score

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	-7.76	-8.92	-9.44	-9.64	-9.73
PW-PEM-HLTA	-7.80	-8.95	-9.31	-9.60	-9.71
A-PEM-HLTA	-8.06	-9.80	-10.26	-10.66	-10.74
PWA-PEM-HLTA	-8.15	-9.79	-10.25	-10.62	-10.70



(a) Running time (b) Consistency score

Fig. 2. Comparison of nips-1955 running time and consistency score

4.3. Experiment 2

Compared the results of Experiment 2 in Table 5-7 with those of Experiment 1 in Table 2-4, the calculation efficiency improvement average fold value is 4.97, which is approximately the same as the result of Experiment 1. The word-based weighting tends to be the same as the tendency of increasing the calculation efficiency and the average subject consistency score. The experimental results show that the improved method has the same effect on small data sets and relatively large data.

4.3. Experiment 3

The results are shown in Table7-8. Under the same environmental conditions, the performance of experimental results on the Reuters news data set has the same trend

as that of the nips data set, but the effect is not as efficient as the improvement of the nips data set. After statistical analysis of the data set, Reuters news data has a total number of word segmentation of 289,759, an average of 144.8795 single document word counts, a total number of word segmentation of nips data set 8515607, and an average number of single document word 4257.8035. Reuters is more sparse than nips data sets when using the bag-of-words model to represent documents. Therefore, there is no big nips in the improvement of computational efficiency.

Table 5. nips-6560 running time/min

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	27	297	787	1865	3683
PW-PEM-HLTA	26	330	760	1801	3430
A-PEM-HLTA	6	50	115	298	635
PWA-PEM-HLTA	6	51	108	276	573

Table 6. nips-6560 average theme consistency score

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	-7.73	-9.07	-9.85	-10.56	-10.34
PW-PEM-HLTA	-7.84	-9.05	-9.91	-10.25	-10.41
A-PEM-HLTA	-8.37	-9.85	-10.71	-11.19	-11.55
PWA-PEM-HLTA	-8.57	-10.02	-10.65	-11.11	-11.41

Table 6. Reuters-2000 running time/min

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	4	16	32	51	89
PW-PEM-HLTA	4	16	31	55	90
A-PEM-HLTA	1	4	11	20	39
PWA-PEM-HLTA	1	4	11	20	41

Table 7. Reuters-2000 average theme consistency score

Method	Nips-1k	Nips-3k	Nips-5k	Nips-7k	Nips-10k
PEM-HLTA	-10.57	-9.97	-8.85	-7.93	-6.85
PW-PEM-HLTA	-10.37	-9.68	-8.73	-7.86	-6.64
A-PEM-HLTA	-10.86	-10.29	-10.22	-9.41	-8.03
PWA-PEM-HLTA	-10.46	-10.69	-9.79	-8.83	-8.00

5 Conclusions

Based on a state-of-the-art hierarchical topic detection method called HLTM, we improved the PEM-HLTA method to reduce computation time. We can use a single machine to handle relatively larger data sets instead of just adding more computing resources. The empirical results show that PWA-PEM-HLTA has a significant improvement in the efficiency of the implementation, allowing 10k words on a personal computer, the data set of 6k documents can be calculated within 12 hours, and data of 5k words in 6k documents can be calculated in 2 hours.

In the future, we plan to further study the application of HLTA's multi-categorization of words and to improve the topic semantic consistency scores. The other is distributed research on HLTA.

References

1. Zhang N.L., Poon L.K.M.: Latent Tree Analysis. In: AAAI Publications, Thirtieth AAAI Conference on Artificial Intelligence, 4891-4897(2017).
2. Knott M., Bartholomew D.J.: Latent Variable Models and Factor Analysis. Arnold, London. In: Journal of Educational Statistics, 7(4), 650-663(1999).
3. Chen Tao, Zhang N.L., Liu Tengfei, Poon K.M., Wang Yi: Model-based multidimensional clustering of categorical data. In: Artificial Intelligence, vol.176, No.1, pp.2246-2269(2012).
4. Li Yang, Aggen S, Shi Songtao, et al.: Subtypes of major depression: latent class analysis in depressed Han Chinese women. In: Psychological Medicine, vol.44, pp.3275-88(2014).
5. Liu Tengfei, Zhang N.L., Chen Peixian.: Hierarchical latent tree analysis for topic detection. In: ECML&PKDD, pp.256-272(2014).
6. Chen Zhourong, Zhang N.L., Yeung D.Y., Chen Peixian.: Sparse boltzmann machines with structure learning as applied to text analysis. In: Thirty-First AAAI CAI, pp.1805-1811(2017).
7. Paisley J., Wang Chong, Blei D.M., Michael I.J.: Nested Hierarchical Dirichlet Processes. In: Pattern Analysis & Machine Intelligence, vol.37, No.2, pp.256-270(2015).
8. Chen Peixian., Zhang N.L., Poon K.M., Chen Zhourong.: Progressive EM for latent tree models and hierarchical topic detection. In: Thirtieth AAAI CAI, pp.1498-1504(2016).
9. Zhang N.L.: Hierarchical latent class models for cluster analysis. In: Journal of Machine Learning Research, vol.5, pp.697–723(2004).
10. Chen Peixian, Zhang N.L., Liu Tengfei, et al.: Latent Tree Models for Hierarchical Topic Detection. In: Artificial Intelligence, vol.250, pp.105-124(2017).
11. Gong Ye Xiaoyan, Lin Peiguang, Ren Weilong, et al.: Thematic word extraction algorithm based on improved TF-IDF algorithm and co-occurrence words. In: Journal of Nanjing University: Natural Science Edition, vol.53, No.6, pp.1072-1080(2017).
12. Han Pu, Wang Dongbo, Liu Yanyun, et al.: Research on the Influence of Part of Speech on the Clustering of Chinese and English Texts. In: Chinese Journal of Information, vol.27, No.2, pp.65-73(2013).
13. Yang W.Y., Cao W, Chung T S, et al.: Applied Numerical Methods Using MATLAB. In: J. Wiley, pp.201-202(2004).
14. Mimno D, Wallach H M, Talley E, et al.: Optimizing Semantic Coherence in Topic Models. In: EMNLP, pp.262-272(2011).