

Discover Underlying Topics in Thai News Articles: A Comparative Study of Probabilistic and Matrix Factorization Approaches

Pimpitcha Pitichotchkphokhin¹, Piyawat Chuangkrud², Kongkan Kalakan³,

Boontawee Suntisrivaraporn⁴, Teerapong Leelanupab⁵ and Nont Kanungsukkasem⁶

^{1,2,5,6} Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand 10520

^{3,4} Siam Commercial Bank, Bangkok, Thailand 10900

Email: ¹pimpitcha.ptitichotchkphokhin@gmail.com, ²piyawatchuangkrud@gmail.com, ³kongkan.kalakan@scb.co.th,

⁴boontawee.suntisrivaraporn@scb.co.th, ⁵teerapong@it.kmitl.ac.th, ⁶nont@it.kmitl.ac.th

Abstract—Topic modeling is an unsupervised learning approach, which can automatically discover the hidden thematic structure in text documents. For text mining, topic modeling is a language-independent technique that disregards grammar and word order. Apart from semantic and structural issues, Thai language is typically considered more complex than others. Due to the lack of word delimiter and a surfeit of composite words. Errors from word tokenization can create significant problems for any post processes of text, such as document retrieval, sentiment analysis, machine translation, etc., adversely decreasing the performance of text applications. Despite a strong correlation between word ordering and semantic meaning, topic modeling has been widely reported that it can extract latent information, aka. latent topic or latent semantic, encoded in documents. Although there were few previous research works on studying topic modeling in Thai language, they mostly focused on upstream processes of Natural Language Processing (NLP) in, for example, applying a refined stop-word list to, or adding N-gram on a single specific topic modeling method. To our knowledge, this paper is the first to explore different topic modeling approaches, i.e., Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorization (NMF), in Thai Language to compare their coherence. We also employ and compare a set of state-of-the-art evaluation metrics based on Topic Coherence.

Keywords—Topic Modeling; Latent Dirichlet Allocation; Non-negative Matrix Factorization; Coherence; Thai News

I. INTRODUCTION

One of the difficulties for programming a computer to process and understand human language is the symbol grounding problem. This problem is related to the semantic gap of how words (symbols) are interpreted to their meanings. One of the interpretation problems is ambiguity, e.g.:

- Lexical ambiguity, which occurs when a word or phrase in a content has more than one meaning; and
- Referential ambiguity, which occurs when a word or phrase in a content can refer to more than one thing.

Many approaches to feature extractions have been proposed to deal with those ambiguities by transforming human language, which is unstructured data, into structured data or useful features for subsequent analysis. This extraction process

of text is part of the well-known field, so-called text mining. Some techniques require strenuous efforts and a large amount of training data to create effective models, but others do not need to be supervised and can extract useful information, e.g., latent topic or latent semantic, which provides valuable insight of those human language underlying in documents, so-called topic modeling.

In Natural language processing (NLP), one of the apparent benefits of topic modeling is a language-independent technique that ignores grammar and word order. Even though Latent Dirichlet Allocation (LDA) [1] seems to be the state-of-the-art topic model, here have been many comparison studies about topic modeling for various kinds of utilization. We found that some studies compared the same approaches but showed inconsistent results. For instance, Bergamaschi *et al.* [2] reported that Latent semantic analysis (LSA) was better than LDA in supporting the suggestion of similar plots of movies on two hundred thousand plots of movie databases. In contrast, the results from another comparative study in automatic bug localization that compared LDA with LSA by Lukins *et al.* [3] demonstrated that LDA-based approach was more effective than the LSA approach.

In general, Thai language is more challenging than English in NLP because it is harder to identify sentence boundaries and grammars [4]. Furthermore, studies on topic modeling in Thai corpus was previously little examined. Most research focused on upstream processes of NLP and also on just a single method of topic modeling. For instance, Nararatwong *et al.* [5] simply applied a refined stop-word list as a text pre-processing step for improving topic extraction by LDA in Thai tweets. Asawaroengchai *et al.* [6] added N-gram as input to LDA, renamed Topic N-grams model (TNG), and compared it with traditional LDA. Accordingly, it would be interesting to see how much differently topic modeling techniques perform on Thai language. We therefore select two topic modeling approaches from two different theoretical bases in our study. One is grounded on linear algebra and the other is on generative statistical model. To our knowledge, this paper is the first effort to compare two distinct topic modeling methods, i.e., Non-negative Matrix Factorization (NMF) and LDA, with the purpose of topic extraction in Thai News Corpus. We are also

the first to use a set of the state-of-the-art evaluation metrics, Topic Coherence [7], to evaluate the different topic modeling approaches on a topic mining task in Thai language. Without the requirement of ground truth data or human assessors, Topic Coherence can quantify the results from topic modeling in terms of semantic coherence. Formally, this study aims to mainly answer a following research question:

“When applying NMF and LDA to Thai News Corpus, which topic models produce more coherent topics?”

II. RELATED WORKS

A. Latent Dirichlet Allocation

Latent Dirichlet allocation (LDA), introduced by Blei *et al.* [1], is a generative probabilistic model designed to extract topics from a corpus, learning from the relationship between words, topics, and documents. Basically, the LDA model was developed based on knowledge from previously proposed models, i.e., Naive Bayes, Mixture of Unigram and Probabilistic Latent Semantic Analysis (PLSA). LDA outputs are a mixture of topics lying in each document, and a probabilistic distribution over words in each topic.

LDA relaxes three assumptions:

- 1) the order of documents are not important.
- 2) the order of words are not important.
- 3) the number of topics is known and constant.

Recently, Kanungsukkasem *et al.* [8] further formalized the assumptions of the LDA model that a corpus of D documents contains V vocabulary words, and document d consists of N_d word tokens, $(w_{d,1}, \dots, w_{d,N_d})$. Each of the n -th word token $w_{d,n}$ in document d is assigned as an index in the vocabulary, $w_{d,n} \in \{1, \dots, V\}$. The final results from LDA are vocabulary word distributions β_k , each of which represents a latent topic $k \in \{1, \dots, K\}$, and topic distributions θ_d per document d . θ_d is calculated from $z_{d,n}$, the topic number of word n in document d . Additionally, η and α are the hyperparameters of the Dirichlet distribution.

The joint distribution of the hidden and observed variables for LDA is:

$$P(\beta, \theta, z, w | \alpha, \eta) = \prod_{k=1}^K P(\beta_k | \eta) \prod_{d=1}^D \left(P(\theta_d | \alpha) \times \left(\prod_{n=1}^{N_d} P(z_{d,n} | \theta_d) P(w_{d,n} | \beta, z_{d,n}) \right) \right) \quad (1)$$

Given the words in documents, the value of hidden variables in the model can be estimated by computing the posterior distribution. An approximation of the posterior can be computed by many approximate inference algorithms, e.g., Markov Chain Monte Carlo (MCMC), expectation propagation and variation Bayes. Blei *et al.* [9] chose Variational Bayes to infer and shape the variables in the model.

B. Non-Negative Matrix Factorization

Non-Negative Matrix Factorization (NMF) is one of the feature selection approaches used to cope with high dimensional dataset and can be exploited in other domains including

topic modeling. It was first introduced by Paatero and Tapper in 1994 [10], and popularized by Lee and Seung in 1999 [11]. Later, Gaussier and Goutte [12] applied NMF and PLSA to generate document clustering models, which is similar to topic modeling. Afterwards, NMF has been used for topic modeling purpose in many studies. Lee and Seung defined NMF:

$$X \approx WH \quad (2)$$

where X is an $n \times m$ matrix of the number of word tokens of each of m word types (vocabulary) in n documents, approximately factorized to multiplication of matrix W and matrix H . W is an $n \times r$ matrix of the number of word tokens of each of r topics in n documents, and H is an $r \times m$ matrix of the number of word tokens of each of m word types in r topics. Basically, r is chosen to be smaller than n or m so that W and H are smaller than the original matrix X . The matrix H can be compared to a vocabulary word distribution β_k of LDA whereas, although not being identical due to probabilistic nature, W can be likened to a topic distribution θ_d of LDA.

C. Topic Coherence

Latent topics have been evaluated manually by human assessors, called human topic ranking. In order to shift from user-centered to system-oriented evaluation, gold standard to which such rankings are referred for measurement needs to be derived from either direct pooling assessment or crowdsourcing. As Newman *et al.* [13] introduced topics represented by a set of words and get evaluated by humans, it has become a baseline to measure topic coherence with other automatic topic ranking. The coherence is calculated based on pointwise mutual information (PMI¹) measuring association between word w_i and word w_j by human ratings.

The UCI-coherence is used to evaluate topic coherence by taking a set of N top words of a topic and summing up the PMI of every word pair of all N top words.

$$C_{UCI} = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^N PMI(w_i, w_j) \quad (3)$$

Besides, PMI can be estimated based on word co-occurrence constructed by a sliding window moving over any external reference corpus, e.g., Wikipedia, as precomputed in Gensim² and becomes a baseline of system-centered evaluation.

Mimno *et al.* [7] introduced the UMASS-coherence that can compute the coherence score without requiring any external corpus like Wikipedia. Rather than divided by the product of probabilities of both words as calculated in PMI, the joint distribution in UMASS is divided by the probability of one word $P(w_j)$. Also, the size of a sliding window of UMASS is commonly equal to the document length. That is, UMASS defines the coherence score based on the number of documents containing words w_i and w_j within an original corpus.

$$C_{UMASS} = \frac{2}{N(N-1)} \sum_{i=2}^N \sum_{j=1}^{i-1} \log \frac{P(w_i, w_j) + \epsilon}{P(w_j)} \quad (4)$$

¹ $PMI(w_i, w_j) = \log((P(w_i, w_j) + \epsilon)/(P(w_i)P(w_j)))$; where ϵ is added to avoid zero in logarithm.

² <https://radimrehurek.com/gensim/models/coherencemodel.html>

Aletras and Stevenson [14] introduced another coherence measure to automatically evaluate a topic model by distributional semantic similarity methods. Instead of PMI in UCI-coherence, they used other similarities, i.e., Cosine similarity, Dice coefficient and Jaccard coefficient, to determine topic coherence. Besides, instead of raw co-occurrence count, the vector were generated by two approaches, i.e., PMI and Normalized PMI (NPMI) for each term. Their evaluation showed that their measures outperformed UCI-coherence and changing from PMI to NPMI in UCI-coherence boosted its performance, later called NPMI-coherence.

$$NPMI(w_i, w_j) = \frac{PMI(w_i, w_j)}{-\log(P(w_i, w_j) + \epsilon)} \quad (5)$$

The latest coherence measure, CV-coherence, combines the indirect cosine similarity with the NPMI and the boolean sliding window [15]. Note that, like the UCI-coherence, both NPMI-coherence and CV-coherence employ Wikipedia as an external source for semantic evaluation.

D. Comparative Study

Lukins *et al.* [3] introduced automatic bug localization systems, exploiting LDA and LSI for extracting latent topics, regarded as hidden crucial information, from source code. Such information is then used to identify the error and provide comments. In comparison with the error checking cases from Mozilla³ and Eclipse. The results showed that LDA was more effective than LSA, in which the first ten analyses showed that LSA found 3 errors from 8 cases but LDA found all 8 errors.

Bergamaschi *et al.* [2] compared two content-based movie recommender systems, exploiting LSA or LDA for dimension reduction of keyword-based features. For each movie, its feature vector of keywords was constructed by tf-idf weighting from 220,000 keywords of a IMDb database. By applying LSA and LDA, the movie-by-keyword matrix can be transformed to the movie-by-topic matrix with the topic column of 500 and 50 long, respectively. In their work, LSA outperformed LDA in terms of precision with respect to user preference.

III. EXPERIMENTAL DESIGN

A. Data Preparation

Thairath online is one of the most popular daily Thai language newspapers in Thailand and its content is in the form of formal written language. Thus, we assume that the dataset from Thairath online is in a good format to be tokenized and formally categorized into topics. By adhering to the best practices of reproducible research, we did not crawl data by ourselves, but used a subset of public thairath-228k dataset, containing 105,491 news articles from Thairath online. The full thairath-228k dataset, which contains 228,937 articles from October 1, 2014 to October 10, 2019, was subset to an acceptable size for data exploration and research. Being an experimental dataset, the subset is available on Medium and our archived websites⁴.

³An open-source Web browser, later named and known as Firefox.

⁴The experimental dataset has been archived and available for reproducing the results reported in this paper. Urls of both original and archived datasets will be given upon acceptance in a camera ready version.

As topic models place importance on word and need input in the form of number, ‘bag-of word (BoW)’ is the feature needed to be extracted from text. However, contents in our dataset contain both Thai and English words. In order to tokenize the words from the dataset, we selected one of the word segmentation algorithms from PyThaiNLP⁵, i.e., *newmm*, *longest*, *attacut*, *icu* and *deepcut*. As *deepcut* is a segmentation technique using deep learning algorithm and more efficient than others for cutting the specific words, we chose *deepcut* to tokenize our dataset and get both Thai and English word tokens. We get a list of common stop words in various languages by library stopwords to cutoff English words. After the tokenization process, we use *PorterStemmer* from NLTK⁶, the popular stemming algorithm to reduce inflected forms of words to a derivation form, to reduce each English word token to its root word.

B. Experimentation and Evaluation

We estimated latent topics from LDA and NMF by using *LDAModel* and *Nmf* classes in library *Gensim*. Three inputs, i.e., a set of tokenized words, a set of dictionary and the number of topics, are necessary for topic modeling to generate latent topics. Besides three parameters used in LDA and NMF were set to default as defined in the *Gensim* library, i.e., setting the maximum number of iterations for inferring the topic distribution by 50. We varied the different numbers of topics (i.e., 5, 10, 12, 15, 20, 25 and 30) to find an appropriate number of topics that give the best result.

We evaluated latent topics from NMF and LDA using system-centered evaluation by a set of the-state-of-the-art measures for topic modeling, so-called *Topic Coherence*. The library *Gensim* provides the suite for the evaluation of topic coherence in a *CoherenceModel* class. In our experiment, we used four coherence measures as dependent variables (i.e., UMASS, UCI, NPMI and CV) to examine which measurement is strong to show and explain two experimental topic models (i.e., model as an independent variable). Four controlled variables were composed of:

- 1) the Thairath dataset as a corpus;
- 2) a set of unique word tokens tokenized by *deepcut* as a dictionary;
- 3) default settings⁷ of sliding window sizes depending on coherence measures; and
- 4) the number of top words (i.e., 20) in each topic.

During the experiment, we discovered that the runtime ranking of the coherence measures is different from the ranking in the study by Rder *et al.* [15]. Thus, we set another experiment to confirm the runtime ranking by the different number of top words, i.e., 20, 50 and 100 (the same number of top words as set in the study of Rder *et al.*)

IV. RESULTS AND DISCUSSION

In this section, we present the results of our comparative study of the generative statistical and linear algebra models,

⁵<https://www.thainlp.org/pythainlp/docs/2.0/>

⁶Natural Language Toolkit, <https://www.nltk.org/howto/stem.html>

⁷110 for CV, 10 for NPMI and UCI and N/A for UMASS.

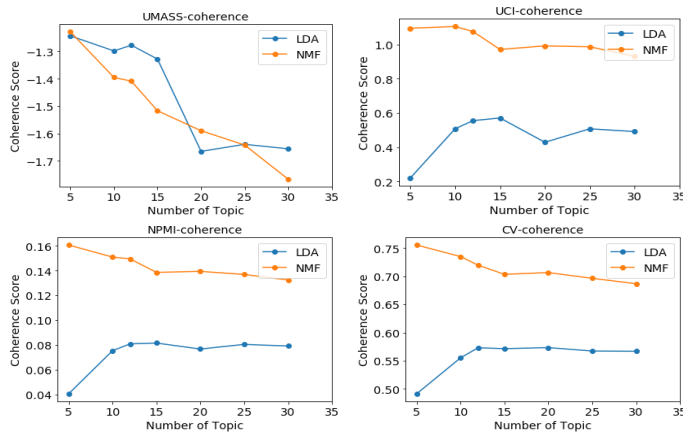


Fig. 1. The average coherence scores of LDA and NMF as evaluated by four different metrics (i.e., UMASS, UCI, NPMI and CV) when varying the number of topics $k \in \{5, 10, 12, 15, 20, 25, 30\}$

i.e., LDA and NMF, respectively, using four different coherence scores in Fig. 1. In the case of UMASS-coherence, topics from LDA in a various number of topics are found to be most coherent, which is inconsistent with the results from UCI-coherence, NPMI-coherence and CV-coherence. This obtained result is unsurprising since the theoretical ground of UMASS-coherence differs from those of the other three coherence metrics, which rely on an external corpus and their norm derived from two words. Additionally, the results from all measures show that the suitable number of topics in LDA should be around 12 and 15 topics because of the peak coherence scores, while the scores of NMF are in the downward form without any peak, leading to difficulty in the identification of the appropriate number of topics.

As we found out that UMASS is the fastest, followed by CV and both NPMI and UCI when N is 20 top words which is different from the study by Rder *et al.* [15] that CV is faster than UCI and NPMI. We experiment with a various number of top words, i.e., 20, 50 and 100, and the runtimes of each coherence are:

- 20 top words; UMASS: 12.2s, UCI: 184s, NPMI: 174s, CV: 401s.
- 50 top words; UMASS: 32.4s, UCI: 295, NPMI: 294, CV: 821s,
- 100 top words; UMASS: 63s, UCI: 373s, NPMI: 401s, CV: 1,485s,

The results show that the runtimes of UMASS are consistent with Rder *et al.* but runtimes of CV become the slowest instead of the second place.

V. CONCLUSION

In this work, we have described an analysis of topic coherence on two topic models, i.e., the probabilistic technique represented by LDA and matrix factorization technique represented by NMF on Thai corpus. The experiment shows that NMF has produced more coherent topics than LDA approach in 3 out of 4 measures, i.e., UCI, NPMI and CV. However,

LDA has produced more coherent topics than NMF in UMASS measurement. All of 4 coherent measurements have shown their peak coherence scores of LDA, manifesting the suitable number of topics, while those of NMF have not.

In conclusion, the coherence measures are worthy to be a tool to select the number of topics for LDA. However, as the coherence score is the similarity score pre-computed from the ranking by human assessors or word co-occurrences on Wikipedia in the past, the human judgments and data from Wikipedia in the past might not be suitable for Thai corpus.

In our future work, instead of using the basic precomputed topic coherence measures provided by an open-source library, we will implement our own coherence measures rating by Thai human and word co-occurrence from multiple sources of Thai formal articles to be more suitable for Thai corpus.

REFERENCES

- [1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," in *Advances in neural information processing systems*, 2002, pp. 601–608.
- [2] S. Bergamaschi and L. Po, "Comparing LDA and LSA topic models for content-based movie recommendation systems," in *Proceedings of the International conference on web information systems and technologies*, 2014, pp. 247–263.
- [3] S. K. Lukins, N. A. Kraft, and L. H. Etzkorn, "Source code retrieval for bug localization using latent dirichlet allocation," in *Proceedings of the 15th Working Conference on Reverse Engineering*, 2008, pp. 155–164.
- [4] S. Singkul and K. Woratpanya, "Thai dependency parsing with character embedding," in *Proceedings of the 11th International Conference on Information Technology and Electrical Engineering*, 2019, pp. 1–5.
- [5] R. Nararatwong, R. Legaspi, N. Cooharajanane, H. Okada, and H. Maruyama, "Solving the difficult problem of topic extraction in thai tweets," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 8, no. 6, pp. 141–145, 2016.
- [6] C. Asawaroengchai, W. Chaisangmongkon, and D. Laowattana, "Probabilistic learning models for topic extraction i thai language," in *2018 5th International Conference on Business and Industrial Research*, 2018, pp. 35–40.
- [7] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011, pp. 262–272.
- [8] N. Kanungsukkasem and T. Leelanupab, "Financial latent dirichlet allocation (finlda): Feature extraction in text and data mining for financial time series prediction," *IEEE Access*, vol. 7, pp. 71 645–71 664, 2019.
- [9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.
- [10] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.
- [11] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.
- [12] E. Gaussier and C. Goutte, "Relation between pls and nmf and implications," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 601–602.
- [13] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, "Automatic evaluation of topic coherence," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010, pp. 100–108.
- [14] N. Aletras and M. Stevenson, "Evaluating topic coherence using distributional semantics," in *Proceedings of the 10th International Conference on Computational Semantics*, 2013, pp. 13–22.
- [15] M. Röder, A. Both, and A. Hinneburg, "Exploring the space of topic coherence measures," in *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, 2015, pp. 399–408.