STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LVIII**, Number 2, 2013

# TEXT REPRESENTATION AND GENERAL TOPIC ANNOTATION BASED ON LATENT DIRICHLET ALLOCATION

DIANA INKPEN[(1)] AND AMIR H. RAZAVI[(2)]

Abstract. We propose a low-dimensional text representation method for topic classification. A Latent Dirichet Allocation (LDA) model is built on a large amount of unlabelled data, in order to extract potential topic clusters. Each document is represented as a distribution over these clusters. We experiment with two datasets. We collected the first dataset from the FriendFeed social network and we manually annotated part of it with 10 general classes. The second dataset is a standard text classification benchmark, Reuters 21578, the R8 subset (annotated with 8 classes). We show that classification based on the LDA representation leads to acceptable results, while combining a bag-of-words representation with the LDA representation leads to further improvements. We also propose a multi-level LDA representation that catches topic cluster distributions from generic ones to more specific ones.

## 1. Introduction

In order to improve the performance of text classification tasks, we always need informative and expressive methods to represent the texts [14] [16]. If we consider the words as the smallest informative unit of a text, there is a variety of well-known quantitative information measures that can be used to represent a text. Such methods have been used in a variety of information

Received by the editors: June 1, 2013.

2010 *Mathematics Subject Classification.* 62Fxx Parametric inference, 62Pxx Applications.

1998 *CR Categories and Descriptors.* code [**I.2.7 Natural Language Processing**]: Subtopic – *Text analisys* code [**H.3.1 Content Analysis and Indexing**]: Subtopic – *Linguistic processing*;

*Key words and phrases.* automatic text classification, topic detection, latent Dirichlet allocation.

This paper has been presented at the International Conference KEPT2013: Knowledge Engineering Principles and Techniques, organized by Babeş-Bolyai University, Cluj-Napoca, July 5-7 2013.

extraction projects, and in many cases have even outperformed some syntax-based methods. There are a variety of Vector Space Models (VSM) which have been well explained and compared, for example in [18]. However, these kinds of representations disregard valuable knowledge that could be inferred by considering the different types of relations between the words. These major relations are actually the essential components that, at a higher level, could express concepts or explain the main topic of a text. A representation method which could add some kind of relations and dependencies to the raw information items, and illustrate the characteristics of a text at different conceptual levels, could play an important role in knowledge extraction, concept analysis and sentiment analysis tasks.

In this paper, the main focus is on how we represent the topics of the texts. Thus, we select a LDA topic-based representation method. We also experiment with a multi-level LDA-based topic representation. Then, we run machine learning algorithms on each representation (or combinations), in order to explore the most discriminative representation for the task of text classification, for the two datasets that we selected.

## 2. Related work

In the most text classification tasks, the texts are represented as a set of independent units such as unigrams / bag of words (BOW), bigrams and/or multi-grams which construct the feature space, and the text is normally represented only by the assigned values (binary, frequency or term TF-IDF[1]) [17]. In this case, since most lexical features occur only a few times in each context, if at all, the representation vectors tend to be very sparse. This method has two disadvantages. First, very similar contexts may be represented by different features in the vector space. Second, in short instances, we will have too many zero features for machine learning algorithms, including supervised classification methods.

Blei, Ng and Jordan proposed the Latent Dirichlet Allocation (LDA) model and a Variational Expectation-Maximization algorithm for training their model. LDA is a generative probabilistic model of a corpus and the idea behind it is that the documents are represented as weighted relevancy vectors over latent topics, where a topic is characterized by a distribution over words. These topic models are a kind of hierarchical Bayesian models of a corpus [2]. The model can unveil the main themes of a corpus which can potentially be used to organize, search, and explore the documents of the corpus. In LDA models, a *topic* is a distribution over the feature space of the corpus and each document can be represented by several topics with different weights.

[1]term frequency / inverse document frequency

The number of topics (clusters) and the proportion of vocabulary that create each topic (the number of words in a cluster) are considered as two hidden variables of the model. The conditional distribution of these variables, given an observed set of documents, is regarded as the main challenge of the model.

Griffiths and Steyvers in 2004, applied a derivation of the Gibbs sampling algorithm for learning LDA models [9]. They showed that the extracted topics capture a meaningful structure of the data. The captured structure is consistent with the class labels assigned by the authors of the articles that composed the dataset. The paper presents further applications of this analysis, such as identifying *hot topics* by examining temporal dynamics and tagging some abstracts to help exploring the semantic content. Since then, the Gibbs sampling algorithm was shown as more efficient than other LDA training methods, e.g., variational EM and Expectation-Propagation [12]. This efficiency is attributed to a famous attribute of LDA namely, "the conjugacy between the Dirichlet distribution and the multinomial likelihood". This means that the conjugate prior is useful, since the posterior distribution is the same as the prior, and it makes inference feasible; therefore, when we are doing sampling, the posterior sampling become easier. Hence, the Gibbs sampling algorithms was applied for inference in a variety of models which extend LDA [19], [7], [4], [3], [11].

Recently, Mimno et al. presented a hybrid algorithm for Bayesian topic modeling in which the main effort is to combine the efficiency of sparse Gibbs sampling with the scalability of online stochastic inference [13]. They used their algorithm to analyze a corpus that included 1.2 million books (33 billion words) with thousands of topics. They showed that their approach reduces the bias of variational inference and can be generalized by many Bayesian hidden-variable models.

## 3. Datasets

The first dataset that we prepared for our experiments consists in threads from the FriendFeed social network. We collected main postings (12,450,658) and their corresponding comments (3,749,890) in order to obtain all the discussion threads (a thread consists in a message and its follow up comments). We filtered out the threads with less than three comments. We were left with 24,000 threads. From these, we used 4,000 randomly-selected threads as background source of data, in order to build the LDA model. We randomly selected 500 threads and manually annotated them with 10 general classes[2], to use as training and test data for the classification. The 10 classes are: *consumers,*

[2]We used only one annotator, but we had a second annotator check a small subset, in order to validate the quality of annotation. In future work, we plan to have a second annotator label all the 500 threads.

| Class | No. of Training Docs | No. of Test Docs | Total |
|---|---|---|---|
| Acq | 1596 | 696 | 2292 |
| Earn | 2840 | 1083 | 3923 |
| Grain | 41 | 10 | 51 |
| Interest | 190 | 81 | 271 |
| Money-fx | 206 | 87 | 293 |
| Ship | 108 | 36 | 144 |
| Trade | 251 | 75 | 326 |
| Crude | 253 | 121 | 374 |
| Total | 5485 | 2189 | 7674 |

TABLE 1. Class distribution of training and testing data for R8.

*education, entertainment, life_stories, lifestyle, politics, relationships, religion, science, social_life* and *technology*.

The second dataset that we chose for our experiments is the well-known R8 subset of the Reuters-21578 collection (excerpted from the UCI machine learning repository), a typical text classification benchmark. The data includes the 8 most frequent classes of Reuteres-21578; hence the topics that will be considered as class labels in our experiments are *acq, crude, earn, grain, interest, money, ship* and *trade*.

In order to follow the Sebastiani's convention [16], we also call the dataset R8. Note that there is also a R10 dataset, and the substantial difference between R10 and R8 is that the classes *corn* and *wheat*, which are closely related to the class *grain*, were removed. The distribution of documents per class and the split into training and test data for the R8 subset is shown in Table 1.

## 4. Method

We trained LDA models for each of the two datasets: one model on 4000 threads from FriendFeed and one model on all the R8 text data. LDA models have two parameters whose values need to be chosen experimentally: the number of topic clusters and the number of words in each cluster. We experimented with various parameter values of the LDA models.

For the first dataset, the best classification results were obtained by setting the number of cluster topics to 50, and the number of words in each cluster to maximum 15.

In LDA models, polysemous words can be member of more than one topical cluster, while synonymous words are normally gathered in the same topics. An example of LDA topic cluster for the first model is: "Google", "email",

"search", "work", "site", "services", "image", "click", "page", "create", "contact", "connect", "buzz", "Gmail", "mail". This could be labeled as *Internet*.

As mentioned, our 500 threads were manually annotated with the 10 generic classes. These classes, enumerated in section 3, are a manually generalized version of the 50 LDA clusters into the 10 generic categories. For the above example, the annotator placed it under the *technology* and *social_life* categories. The classification task is therefore multi-class, since a thread can be in more than one class. We trained binary classifiers from Weka [20] for each class, and averaged the results over all classes.

The manual mapping of LDA clusters into generic classes would allow us to automatically annotate more training data form the FriendFeed dataset, in our future work. Since each document has LDA clusters that were associated to it during the Gibbs sampling process, the generic classes for these clusters can be obtained, and one or more labels can be assigned to the document. Only the labels with high LDA weights will be retained. If the weights are low for all labels, the document would not be added to the training data. If more than one label has high weight, the document would have multiple labels. This process would allow us to add a large amount of training data, perhaps with some noise. For more details see [15].

For the classification task, we chose several classifiers from Weka: Naive Bayes (NB) because it is fast and works well with text, SVM since it is known to obtain high performance on many tasks, and decision trees because we can manually inspect the learned tree.

We applied these classifies on simple bag-of-words (BOW) representation, on LDA-based representations of different granularities, and on an integrated representation concatenating the BOW features and the LDA features. The values of the LDA-based features for each document are the weights of the clusters associated to the document by the LDA model (probability distributions).

## 5. Experiments and Results

The results on the first dataset are presented in Table 2. After stopword removal and stemming, the bag-of-words (BOW) representation contained 6573 words as features (TF-IDF values). The lower-dimensional representation based on LDA contained 50 features, whose values are the weights corresponding to the topic clusters. For the combined representation (BOW integrated with the LDA topics) the number of features was 6623.

We observed that the 10 class labels (general topics) are distributed unevenly over the dataset of 500 threads, in which we had 21 threads for the class *consumers*, 10 threads for *education*, 92 threads for *entertainment*, 28 threads

| Representation / Classifier | Accuracy |
|---|---|
| BOW(TF-IDF)/ CompNB | 77.22% |
| LDA Topics / Adaboost (j48) | 69.32% |
| BOW(TF-IDF)+LDA / SVM(SMO) | 80.00% |

TABLE 2. Results on the FriendFeed dataset.

for *incidents*, 90 threads for *lifestyle*, 27 threads for *politics*, 58 threads for *relationships*, 31 threads for *science*, 49 threads for *social_activities*, and 94 threads for *technology*. Thus, the baseline of any classification experiment over this dataset may be considered as 18.8%, for a trivial classifier that puts everything in the most frequent class, *technology*. However, after balancing the above distribution through over/under sampling techniques, the classification baseline lowered to 10%.

On this dataset, we conducted the classification evaluations using stratified 10-fold cross-validations (this means that the classifier is trained on nine parts of the data and tested on the remaining part, then this is repeated 10 times for different splits, and the results are averaged over the 10 folds). We performed several experiments on a range of classifiers and parameters for each representation, to check the stability of a classifier's performance. We changed the *seed*, a randomization parameter of the 10-fold cross-validation, in order to avoid the accidental over-fitting.

For the BOW representation, the best classifier was Complement Naive Bayes (a version of NB that compensates for data imbalance), with an accuracy of 77.22%. Using the low-dimensional LDA representation, the accuracy goes down, but it has the advantage that the classifiers are faster and other classifiers could be used (that do not usually run on high-dimensional data). Combining the two representations achieved the best results, 80% accuracy.

The results on the second dataset, R8, are shown in Table 3. We experimented with several parameters for the LDA model: 8, 16, 32, 64, 128, and 256 for the number of clusters (therefore we build 6 models). We chose 20 words in each cluster. The reason we started with 8 clusters is that there are 8 classes in the annotated data. We experimented with combinations of the models in the feature representation (a multi-leveal LDA-based representation), leaving up to the classifier to choose an appropriate level of generalization.

After stopword removal and stemming, the BOW representation (TF-IDF values) contained 17387 words as the feature space. We experimented with each LDA representation separately, without good results; therefore we chose a combined 6-level representation (corresponding to the LDA models with 256, 128, 64, 32, 16, 8 clusters). For the integrated representation BOW with LDA

| Representation / Classifier | Accuracy |
|---|---|
| BOW / SVM | 93.33% |
| LDA Topics / SVM | 95.89% |
| LDA+BOW / SVM | 97.03% |
| BOW / NB | 95.20% |
| LDA Topics / NB | 94.61% |
| LDA+BOW / NB | 95.52% |
| BOW / DT | 91.54% |
| LDA Topics / DT | 91.78% |

TABLE 3. Results on the R8 dataset.

topics we had 17891 features (256 + 128 + 64 + 32 + 16 + 8 = 504, plus the 17387 words).

The average classification accuracy is very high, compared to a baseline of 51% (of a simplistic 8-way classifier that always chooses the most frequent class, *earn* in this dataset). The SVM and NB classifiers achieved the best results. These values are in line with state-of-the art results reports in the literature. We can compare our results with other reported classification results of the same dataset. According to the best of our knowledge, the accuracy of our integrated representation method on the Reuters R8 dataset, 97%, is higher than any simple and combinatory representation method from related work, which reports accuracies of 88%–95% [6], [1], [5], while 96% was reached with SVM on a complex representation method based on kernel functions and Latent Semantic Indexing [21].

For SVM, the LDA-based representation achieved better accuracy (95.89%) than the BOW representation (93.33%). This is due to the multi-level representation. When we experimented with each level separately, the accuracies dropped considerably. The best results over all the experiments were for SVM with the combined BOW and LDA-based representation.

## 6. Conclusions and Future Work

As our experimental results show, we can achieve good classification results by using a low-dimensional representation based on LDA. This representation has the advantage that allows the use of classifiers or clustering algorithms that cannot run on high-dimensional feature spaces. By using a multi-level representation (different generalization levels) we achieved better results than the BOW representation on the second dataset. In future work, we plan to test the multi-level representation on the first dataset, to confirm our hypothesis that it is better to let classifiers choose the appropriate level of generalization.

The combined BOW and LDA features representation achieved the best classification performance, and it can be used when there memory is not a concern, for classifiers that are able to cope with the large vector spaces.

Our results show that the first dataset is more difficult to classify than the second dataset. The reason is that it consists in social media texts, which are very noisy. In future work, we plan to experiment with more training data for the FriendFeed dataset (automatically annotated via the mapping of LDA clusters into the 10 classes), and to design representation and classification methods that are more appropriate for this kind of data.

## Acknowledgments

The authors wish to thank Ontario Centres of Excellence and to The Natural Sciences and Engineering Research Council of Canada for the financial support. We thank Lana Bogouslavski for annotating the FriendFeed data and Dmitry Brusilovsky for his insights in the project.

## References

[1] Charu C. Aggarwal and Peixiang Zhao. 2012. Towards graphical models for text processing; Knowledge Information Systems, DOI 10.1007/s10115-012-0552-3; Springer-Verlag London.

[2] David M. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum. Hierarchical topic models and the nested Chinese restaurant process. In Proceedings of the conference on Neutral Processing Information Systems NIPS 2003.

[3] David M. Blei and J. McAulie. 2007. Supervised topic models. In Proceedings of the conference on Neutral Processing Information Systems NIPS 2007.

[4] David M. Blei, Andrew Ng, and Michael Jordan. Latent Dirichlet Allocation. 2003. Journal of Machine Learning Research, 3: 993–1022.

[5] Ana Cardoso-Cachopo, Arlindo L. Oliveira. 2007. Combining LSI with other Classifiers to Improve Accuracy of Single-label Text Categorization. In Proceedings of the first European Workshop on Latent Semantic Analysis in Technology Enhanced Learning, EWLSATEL 2007.

[6] Yen-Liang Chen and Tung-Lin Yu. 2011. News Classification based on experts' work knowledge. In Proceedings of the 2nd International Conference on Networking and Information Technology IPCSIT 2011, vol.17 ; IACSIT Press, Singapore.

[7] Andrew McCallum and X. Wang. 2005. Topic and role discovery in social networks. In Proceedings of IJCAI 2005.

[8] J. R. Firth et al. 1957. Studies in Linguistic Analysis. A synopsis of linguistic theory, 1930–1955. Special volume of the Philological Society. Oxford: Blackwell.

[9] Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. In Proceedings of the National Academy of Sciences, 101 (Suppl 1), 5228–5235.

[10] Gregor Heinrich. 2004. Parameter estimation for text analysis, Technical Report (For further information please refer to JGibbLDA at the following link: http://jgibblda.sourceforge.net/)

[11] Wei Li and Andrew McCallum. 2006. Pachinko allocation: Dag-structured mixture models of topic correlations. In Proceedings of ICML 2006.

[12] Thomas Minka and John Lafferty. 2002. Expectation propagation for the generative aspect model. In Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence UAI 2002. https://research.microsoft.com/minka/papers/aspect/minka-aspect.pdf.
[13] David Mimno, M. Hoffman, and David M. Blei. 2012. Sparse stochastic inference for latent Dirichlet allocation. In Proceedings of International Conference on Machine Learning ICML 2012.
[14] Xiaoshan Pan and Hisham Assal. 2003. Providing context for free text interpretation. In Proceedings of Natural Language Processing and Knowledge Engineering, 704–709.
[15] Amir H. Razavi and Diana Inkpen. 2013. General Topic Annotation in Social Networks: A Latent Dirichlet Allocation Approach. In Proceedings of the 26th Canadian Conference on Artificial Intelligence (AI 2013), Regina, SK, Canada.
[16] Fabrizio Sebastiani. 2006. Classification of text, automatic. In Keith Brown (ed.), The Encyclopedia of Language and Linguistics, Volume 14, 2nd Edition, Elsevier Science Publishers, Amsterdam, NL, 457–462.
[17] Karin Spark Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 28 (1): 11–21.
[18] Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics, Journal of Artificial Intelligence Research (JAIR), 37, 141–188.
[19] Xuerui Wang and Andrew McCallum. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In Proceedings of ACM SIGKDD conference on Knowledge Discovery and Data Mining KDD 2006.
[20] Ian H. Witten and Eibe Frank. 2005. Data Mining: Practical Machine Learning Tools and Techniques, 2nd edition, Morgan Kaufmann, San Francisco.
[21] Man Yuan, Yuan Xin Ouyang and Zhang Xiong. 2013. A Text Categorization Method using Extended Vector Space Model by Frequent Term Sets. Journal of Information Science and Engineering 29, 99–114.

(1) University of Ottawa, School of Electrical Engineering and Computer Science, Ottawa, ON, Canada, K1N 6N5
*E-mail address*: Diana.Inkpen@uottawa.ca

(2) University of Ottawa, School of Electrical Engineering and Computer Science, Ottawa, ON, Canada, K1N 6N5
*E-mail address*: araza082@eecs.uottawa.ca