Hidden Topics Modeling Approach for Review Quality Prediction and Classification

Hoan Tran Quoc, Hideya Ochiai, Hiroshi Esaki
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, Japan
zoro@hongo.wide.ad.jp, jo2lxq@hongo.wide.ad.jp, hiroshi@wide.ad.jp

***Abstract*—The automatic assessment of online review's quality is becoming important with the number of reviews increasing rapidly. In order to help determining review's quality, some online services provide a system where users can evaluate or feedback the helpfulness of review as crowdsourcing knowledge. This approach has shortcomings of sparse voted data and richer-get-richer problem in which favor reviews are voted frequently more than others. In this work, we use Latent Dirichlet Allocation (LDA) method to exploit hidden topics distribution information of all reviews and propose supervisor prediction model based on probabilistic meaning of the review's quality. We also propose a deep neural network to classify the review in quality and validate our proposals within some real reviews datasets. We demonstrate that using hidden topics distribution information could be helpful to improve the accuracy of review quality prediction and classification.**

***Keywords*-review quality prediction; hidden topics modeling; LDA; deep neural network;**

I. Introduction

The big volume of online reviews today makes the process of extracting helpful information becoming more and more difficult. Users are encountering with mind confusing problem to find the interesting and helpful opinion in mixtures of unhelpful or highly subjective and misleading information. To deal with this problem, some review portal sites are providing a mechanism where users can evaluate or rate the helpfulness of a review (e.g. Amazon.com and Yelp.com). However, the disadvantage of provided mechanism is the top reviews attract more and more rating while more recent reviews are rarely read and thus not rated [1]. It is thus highly desirable to develop robust and reliable methods to evaluate the quality of reviews automatically.

In this paper, we propose a probabilistic definition for quality of review and investigate how the hidden topics distribution information extracted from reviews can help improve the accuracy of supervisor quality predictor and classifier. To the best of our knowledge, this is the first time review quality is modeled as a probabilistic and statistical model. Furthermore, this is also the first time that textual features and hidden topics distribution features over all reviews are combined for assessing review quality.

In generally, our simple idea is that all reviews discussed the same number of topics but with the different proportion of topics in each review. We call this kind of topics as hidden topics and use Latent Dirichlet Allocation (LDA) [8] method to extract these hidden topics and their proportion in each review. We have an intuition that two reviews with the same topics proportion may have the same quality. We formulate our intuition by display each review as a vector of features that are the topics proportions. Then we propose a logistics model for quality prediction and a deep neural network (RVDeepNet) for quality classification. Finally, we demonstrate that topics proportion features could be helpful to improve the accuracy of predictor and classifier within real review data from some online review portals.

II. Related Works

Most of previous works in [1] [2] [3] [4] [5] has addressed the solution to problem of review's quality evaluation by treating each review as a stand-alone text document, extracting statistical textual features from the text and proposing a function based on these features for predicting review quality as the user-generated helpfulness vote proportion. Lu and Tsaparas [6] studied the quality by incorporating social context information. This approach is promising for the increasing of social relationships between reviewers. However, for general review portals, social context may be lacked or untrusted. In addition to statistical textual features, there is much more information available as hidden and latent topics in review's text. In our approach, we plan to apply techniques for general type of review portals by combining textual and hidden topics distribution information to evaluate the quality of an individual review.

III. Target system

Our target system is described in figure 1. The reviews which have number of votes higher than or equal ten (in explicit region) will be used as supervisor data. Our system solves for two main tasks: quality prediction and quality classification. The first one includes quality definition and quality inference, the second one includes quality classify

and class inference for the reviews that do not have enough votes (in the buried region).

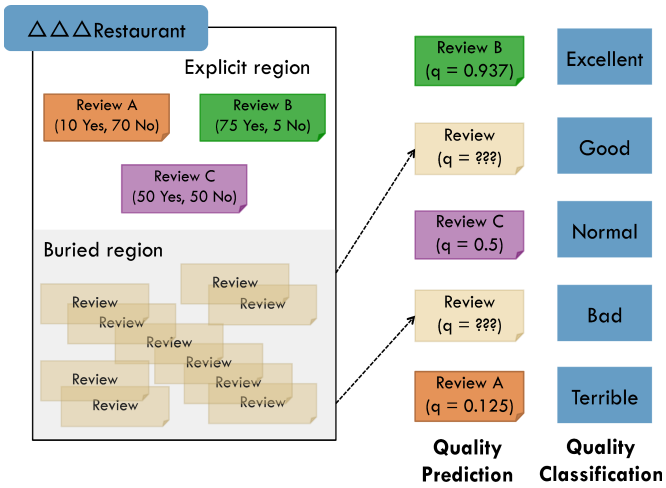

Figure 1. Target system with review quality prediction and classification tasks. System uses review in the explicit region as supervisor data and evaluates quality for the reviews that do not have enough votes in the buried region.

A. Quality Prediction Based Estimation

For each review r_i, let p_i be the probability for one vote of r_i to be a helpful vote. We can assume that p_i depends on features vector $\mathbf{f_i}$ and common parameters vector $\mathbf{w}$ for all reviews. We define the quality of review i as p_i.

Definition 1. *Quality of review r_i is defined as the probability that one vote for this review to be a helpful vote.*

The vote set of review r_i is defined as $V_i = \{v_i^{(1)}, v_i^{(2)}, ..., v_i^{(N_i)}\}$ where each $v_i^{(k)}$ is a random variable taking value of 0 (unhelpful vote) or 1 (helpful vote). The distribution of v_i^k (given p_i) follows the Bernoulli distribution $p(v|p_i) = p_i^v(1-p_i)^{1-v}$. We denote the number of votes and number of helpful votes for review r_i as N_i and h_i. We can assume that each vote is independent event, then the distribution of V_i given p_i is followed by the below distribution.

$$p(V_i|p_i) = \prod_k p(v_i^{(k)}|p_i) = p_i^{h_i}(1-p_i)^{N_i-h_i} \quad (1)$$

In the formula above, for the simple prediction, we assume that each pair of $\mathbf{f}_i$ and $\mathbf{w}$ defines a unique value of p_i. Because p_i is the quantitative probability value defined in range [0, 1], it could be displayed as the logistic function:

$$p_i = logistic(z_i) = \frac{1}{1+exp(-z_i)} \quad (2)$$

where z_i is expressed as linear combination from features space: $z_i = \mathbf{w}^T\mathbf{f_i}$.

If we can inference the common weights vector $\mathbf{w}$, we can calculate the quality from review features vector $\mathbf{f}$ by (2). The maximum likelihood function and optimization solution $\hat{\mathbf{w}}$ will be defined as the following formulas.

$$\begin{aligned} \hat{\mathbf{w}} &= argmax_{\mathbf{w}} logL(\mathbf{w}) \\ L(\mathbf{w}) &= \prod_i p(V_i|p_i) = \prod_i p_i^{h_i}(1-p_i)^{N_i-h_i} \\ logL(\mathbf{w}) &= \sum_i \{h_i log(p_i) + (N_i - h_i)log(1-p_i)\} \end{aligned} \quad (3)$$

Notice that p_i could also be predicted by following linear model.

$$p_i \sim q_i = \frac{Number\_of\_helpful\_votes}{Number\_of\_votes} = \mathbf{w}^T\mathbf{f}_i \quad (4)$$

Parameter vector $\mathbf{w}$ is obtained by minimizing the quadratic loss $\sum_{i=1}^{n}(q_i - \mathbf{w}^T\mathbf{f}_i)^2$. However, the linear model has the disadvantage that the linear combination could be any real value while p_i has to be in restricted range [0, 1]. Moreover, using only the helpfulness ratio from user feedback votes could not tell us how the votes for the review are produced. For example, "3 out of 10 people found the review helpful" may not be the same as "300 of 1000 people found the review helpful" although they have the same helpful votes ratio as 0.3.

B. Quality Classification Based Estimation

In this sub-section, we solve the classification problem of review quality in target system. As ground truth labeled data, we define five categories of quality that represent the different helpful votes ratio's ranges from user feedback votes (table I). We also propose RVDeepNet as a deep convolutional neural network for classification task with the input is 8 x 8 numeric matrix computed from features vector representation of each review. The input is connected to 9 hidden layers, including three 2 x 2 convolutional layers and three 2 x 2 max-pooling layers, three Rectified Linear Unit (ReLU) layers. The 10-th hidden layer is a fully connected layer with 256 dimensional input vector and 5 output neurons for classification task. We train RVDeepNet using ADADELTA optimizer with default parameters in [7] to minimize the cross entropy between the network output and ground truth labels. The detail of RVDeepNet is described in figure 2.

Table I
FIVE CATEGORIES OF REVIEW QUALITY

Review categories	Helpful votes ratio r
Terrible	$0 \leq r < 0.2$
Bad	$0.2 \leq r < 0.4$
Normal	$0.4 \leq r < 0.6$
Good	$0.6 \leq r < 0.8$
Excellent	$0.8 \leq r \leq 1$

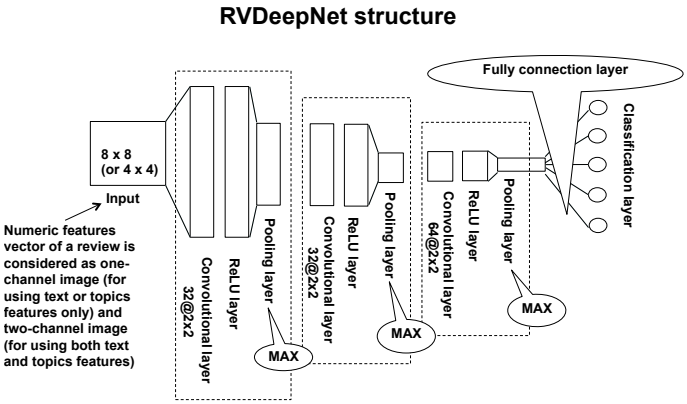

Figure 2. RVDeepNet structure: the features vector of each review is rearranged and normalized as one-channel channel image with size 8 x 8 (for topics features only) and 4 x 4 (for text features only), or two-channel image for combination of both topics and text features. The network has three 2 x 2 convolutional layers and three 2 x 2 max-pooling layers, three Rectified Linear Unit (ReLU) layers and one fully connected layer which generates 5 output neurons for classification task.

C. Incorporating Hidden Topics Distribution Features

The text of a review provides rich information about its quality. Lu and Tsaparas [6] grouped the features for review's quality predictor into three different types.

Text-statistic features: The aggregate statistical features over the text, such as the review's length, the average length of a sentence, or the richness of the vocabulary.

Syntactic features: The statistical features based on the Part-Of-Speech (POS) tags of the words in the text, such as percentage of nouns, adjectives, punctuations, etc.

Sentiment features: The features that take into account the positive or negative sentiment of words in the review.

However, they are not enough to reveal the content of review which is the major factor to evaluate its quality. For example, consider two following reviews for a restaurant.

> **Review 1**: *This place is one of the best spots in this area. I came here as my first date. He was very sweet to me. We talked a lot about school, summer holidays, the newest movies of Tom Cruise. He's very intelligent and as he was speaking, I felt dizzy and hot. I could no longer focus on his words. I controlled myself to not say something stupid. It's wonderful day with me.*

> **Review 2**: *We ordered the omakase and truly enjoyed each dish and experienced topped off wagu beef that I'm still dreaming about! All the nigiri/sashimi was super tasty and fresh!*

We call these types of features proposed in [6] as "explicit textual features". In the estimator using only explicit textual features, review 1 is evaluated with higher quality than review 2 for abundant words and plenty of positive opinions. However, review 1 mentions only about reviewer's boyfriend without useful information about restaurant. Review 2 with keywords like "omakase", "wagu beef", "nigiri/sashimi", "fresh", "tasty" is representative review that helps reader understand the characteristics of restaurant. Review 2 is better in this circumstance.

Quality of a review depends on the content of this review. The reviews which discussed the same things or the same opinions about the same topics may have the same quality. A review could discuss one major topic or mixture of topics. It's more appropriate to use the distribution of topics in each individual review as features in quality evaluation. To figure out the distribution of topics in each individual review and in all reviews community, we use Latent Dirichlet Allocation (LDA) method for topic modeling.

1) Latent Dirichlet Allocation: Latent Dirichlet Allocation (LDA) [8] is a Bayesian generative model that describes how the documents in a dataset were created. It is used as an unsupervised method to discover the underlying topics covered by a text document. LDA assumes that a corpus of text documents is just a collection of topics where each topic has some particular probability for generating a particular word. The particular probability is determined by looking at each training document as a "bag of words" from a distribution selected by Dirichlet process.

Traditional LDA can be represented by plate diagram (Figure 3) of graphical model that defines the pattern of conditional dependence between random variables. Unshaded and shaded circles display for latent random variables and observed random variables respectively. Edges represent dependences between variables, and the rectangular plates indicate repetition. The parameters used in LDA model are summarized in table II.

Table II
PARAMETERS USED IN LDA MODEL

Symbols	
α	Dirichlet parameter
η	Dirichlet parameter
θ_d	Topic proportion for d^{th} document
z_{dn}	Topic assignment of n^{th} word in d^{th} document
w_{dn}	n^{th} word in d^{th} document
β_k	Distribution of terms in k^{th} topic
N	Number of words per document
D	Number of documents
K	Number of topics
V	Vocabulary set

LDA generative model describes how each document obtained its words. Each topic β_i is defined as a multinomial distribution over a word dictionary with $|V|$ words drawn from a Dirichlet process $\beta_i \sim Dirichlet(\eta)$. The LDA generative process for a document d and number K of topics is described as following steps in [8].

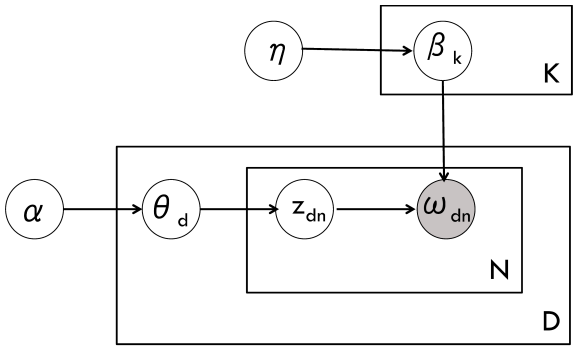

Figure 3. LDA plate diagram

LDA Generative Process

1) Randomly choose number of words N for document d from Poisson distribution: $N \sim Poisson(\lambda)$
2) Randomly choose a distribution over topics for document d from Dirichlet process: $\theta_{\mathbf{d}} \sim Dir(\alpha)$. $\theta_{\mathbf{d}}$ is the parameter for multinomial distribution.
3) For each n^{th} word in the document d:
   - Randomly choose topic z_{dn} from the distribution over topics: $z_{dn} \sim Multinomial(\theta_{\mathbf{d}})$
   - Randomly choose a word w_{dn} from one of $|V|$ words in the corresponding topic. The selected probability is defined as $p(w_{dn}|z_{dn}, \beta)$. β is a $K \times V$ matrix whose row $\beta_i \sim Dirichlet(\eta)$ and β_{ij} is the probability that j^{th} word in vocabulary assigned to topic i.

Suppose we have a set of documents and some fixed number of K topics to discover and we do not know K topic distributions for our corpus. We use LDA process to learn the topic representation of each document and the words associated to each topic that best fit the corpus. The only observed variable is the bag of words $\{w_{dn}\}$, we want to learn latent variables: β_k (distribution over vocabulary set for topic k) and θ_{dk} (topic proportion of topic k in document d). In this paper, we do not go in details for solving methods but using the variational Bayesian method and online learning approach in [9] to discover latent topics distribution where reviews were processed in "batches" and the topic model was updated incrementally after processing each batch.

2) Extracting Features from Hidden Topics Distribution: After getting topics distribution θ_d as K-dimensional vector for each review d, we realized that for a new review that does not mention about any of K topics, the distribution elements are almost the same (as $1/K$). We make a post-processing for θ_d, that is multiply $\theta_{\mathbf{d}}$ to standard deviation of all elements in $\theta_{\mathbf{d}}$, and then normalize vector $\theta_{\mathbf{d}}$ with its maximum element. Finally, we combined this K-dimensional vector $\theta_{\mathbf{d}}$ to the explicit textual features. The features for our estimator are summarized in Table III.

Table III
EXPLICIT TEXTUAL FEATURES AND HIDDEN TOPICS DISTRIBUTION FEATURES

Feature Name	Type	Feature Description
EXPLICIT TEXTUAL FEATURES proposed in [6]		
NumToken	Text-Stat	Total number of tokens
NumSent	Text-Stat	Total number of sentences
UniqWordRatio	Text-Stat	Ratio of unique words
SentLen	Text-Stat	Average sentence length
POS:NN	Syntactic	Ratio of nouns
POS:JJ	Syntactic	Ratio of adjectives
POS:COMP	Syntactic	Ratio of comparatives
POS:V	Syntactic	Ratio of verbs
POS:RB	Syntactic	Ratio of adverbs
POS:FW	Syntactic	Ratio of foreign words
POS:CD	Syntactic	Ratio of numbers
PosSEN	Sentiment	Ratio of positive words
NegSEN	Sentiment	Ratio of negative words
HIDDEN TOPICS DISTRIBUTION FEATURES		
θ_d (K-dim)	Top-Stat	Topics distribution in review d

IV. EXPERIMENTS AND RESULTS

In this section, we present the experimental evaluation with real online review datasets for our proposals. The experiments are performed with the review data from the Yelp Dataset Challenge, TripAdvisor Dataset provided in [10] [11]. The data have users' votes as helpful or unhelpful votes for each review. We used MongoDB to store data and Python scripts for analysis. Specifically, we used the Gensim Python Library, which is a topic modeling tool for documents and Numpy, Scikit-learn package [12], Chainer framework [13] for computation.

A. Prediction Performance with Features Design

We test our fitting models and our proposed features with the review data that has number of votes for each review higher than or equal ten. The learning target is helpful votes proportion for each review. We evaluate the prediction performance by randomly split the data equally into training set (R_{train}) and testing set (R_{test}). The test data size is fixed, while the training data size is reduced to different proportions (25%, 50%, 75%, 100%) to study the effect of training data size on the prediction performance. Twenty independent random splits are conducted to examine the median of evaluation metrics.

We compare the design of features for review quality prediction by considering the following combinations.

Linear-Text-Only: Linear fitting model using only explicit textual features.

Linear-Topics-Only: Linear fitting model using only hidden topics distribution features.

Linear-Text-Topics: Linear fitting model using both explicit textual features and hidden topics distribution features.

Logistic-Text-Only: Logistic fitting model using only explicit textual features.

Logistic-Topics-Only: Logistic fitting model using only hidden topics distribution features.

Logistic-Text-Topics: Logistic fitting model using both explicit textual features and hidden topics distribution features.

The effectiveness of features design in fitting model is evaluated by Mean Absolute Error (MAE) metric between predicted values set $\{Q(\mathbf{r}_i), i = 1, 2, ...\}$ and learning target set $\{q_i, i = 1, 2, ...\}$ over the test set.

$$MAE(R_{test}) = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \mid Q(\mathbf{r}_i) - q_i \mid$$

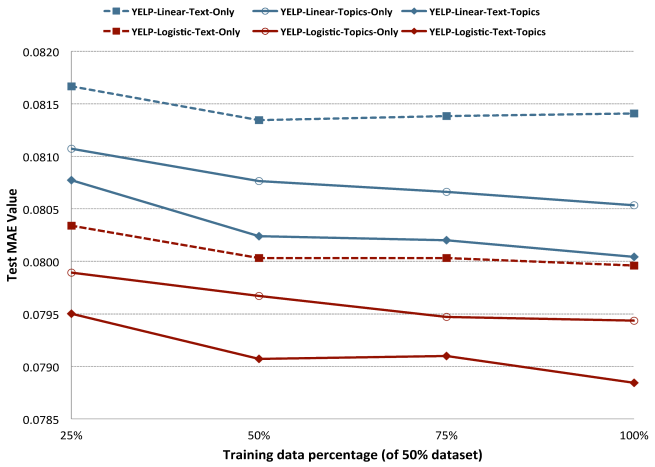

Figure 4. Prediction Performance for Yelp Dataset Challenge [10]. Incorporating hidden topics distribution features shows significant improvements in MAE values especially when the training data is sufficient.

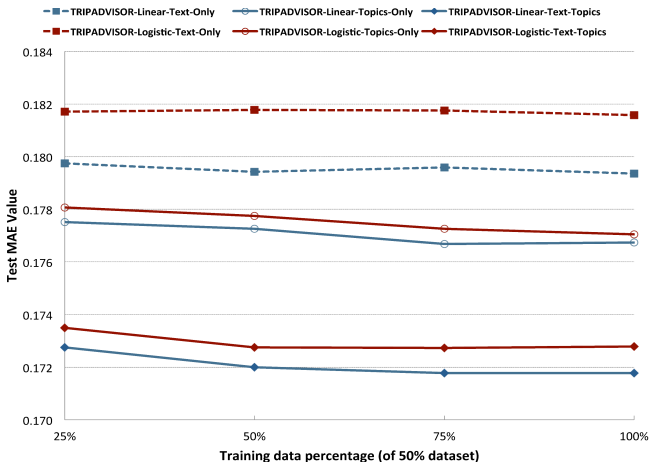

Figure 5. Prediction Performance for TripAdvisor Dataset [11]. Incorporating hidden topics distribution features shows improvements in MAE value. However, the MAE values are still high even when the training data size is increased.

The results of experiments for Yelp Dataset Challenge and TripAdvisor dataset are summarized in figures 4, 5 with the median of MAE for each model and each feature design. For Yelp Dataset Challenge, incorporating hidden topics distribution features shows significant improvements over the explicit textual features baseline especially when the training data is sufficient. Incorporating hidden topics distribution features also shows improvements in MAE values for TripAdvisor Dataset. However, the MAE values are still high even when the training data size is increased.

B. Classification Performance with Features Design

In this sub-section, we test different classification algorithms with different features designs (as using explicit textual features only, hidden topics distribution features only or combination of them). We use the set of review data that has number of votes for each review higher than or equal ten for evaluation. We evaluate the classification performance by randomly split the data into 80% for training (R_{train}) and 20% for testing (R_{test}). Twenty independent random splits are conducted to examine the median of evaluation metrics.

The effectiveness of features design in classification algorithms is evaluated by accuracy percentage when predicting category for the review in the test set. The results of experiments for Yelp Dataset Challenge and TripAdvisor Dataset are summarized in figures 6, 7 with the median of accuracy for each algorithm and each feature design.

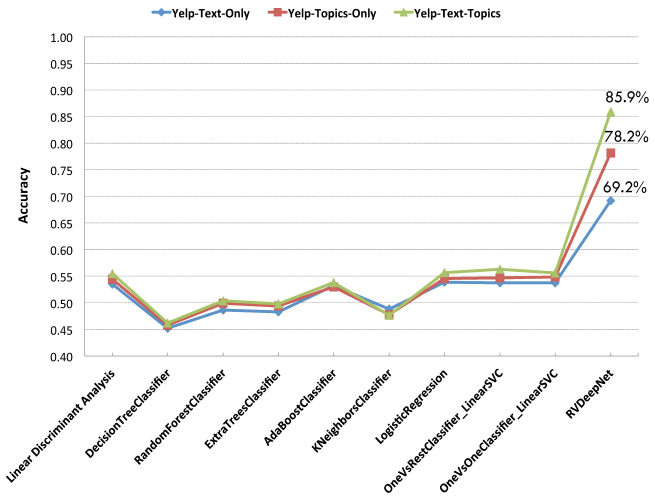

Figure 6. Classification Performance for Yelp Dataset Challenge [10]. Adding hidden topics distribution as new features for classification task does not show improvements in almost conventional classification algorithms but significant improvement in our proposed RVDeepNet.

Adding hidden topics distribution as new features for classification problem does not show improvements in almost conventional classification algorithms but significant improvement in our proposed RVDeepNet. It shows that our RVDeepNet is better for classification task and our proposed hidden topics distribution features represent the essential quality of review better than the explicit textual features.

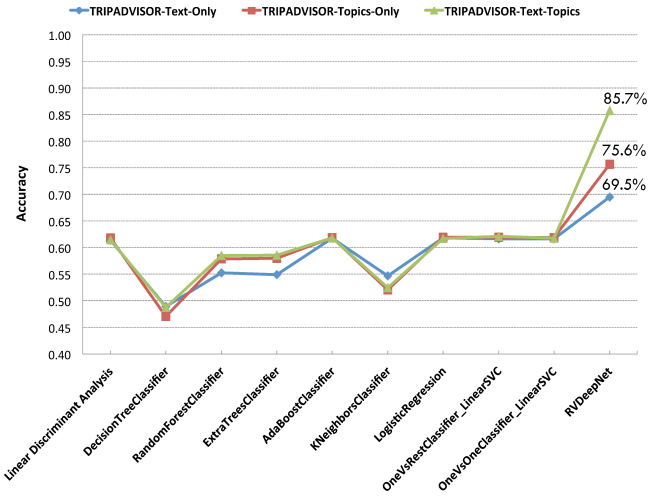

Figure 7. Classification Performance for TripAdvisor Dataset [11]. Adding hidden topics distribution as new features for classification task does not show improvements in almost conventional classification algorithms but significant improvement in our proposed RVDeepNet.

V. Conclusion and Future Work

In this paper, we studied the problem of automatically determining and classifying review quality by using hidden topics distribution information in online review dataset. We proposed a probability definition for review quality and applied logistics fitting model for prediction that caught the true probabilistic meaning of review quality. For classification task, we proposed a deep convolutional neural network (RVDeepNet) that is better than other conventional classification algorithms. We also proposed the hidden topics distribution information in each review to represent each review as features vector. We demonstrated that hidden topics distribution which captured the essential content of review could be helpful to improve the accuracy of prediction and classification problem (compared with previous methods). The probabilistic prediction, classification model and represented features we proposed are quite generalizable and applicable for quality evaluation in real online review dataset or other user-generated contents.

As future work, hidden topics distribution information can be enhanced with reviewer opinion or attitude as: "good service", "delicious raw fish", "great table for family", etc. The relation between the topics distribution in each review with the item's characteristics that user reviewed also is useful information for prediction and classification tasks.

Although user votes can be used as ground-truth data, this kind of data has some biases described in [1]. Therefore, we plan to develop a new ground-truth by proposing a specification on quality of reviews. Moreover, we plan to extend and evaluate our method to other review datasets and implement in our self-built recommendation system.

References

[1] J. Liu, Y. Cao, C.-Y. Lin, Y. Huang, and M. Zhou, "Low-quality product review detection in opinion summarization," in *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007, pp. 334-342.

[2] S. M. Kim, P. Pantel, T. Chklovski, and M. Pennacchiotti, "Automatically assessing review helpfulness," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, pp. 423-430, Stroudsburg, PA, USA, 2006.

[3] A. Ghose and P. G. Ipeirotis, "Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics," in *IEEE Trans. on Knowl. and Data Eng.*, vol. 23(10), pp. 1498-1512, 2011.

[4] Y. Liu, X. Huang, A. An, and X. Yu, "Modeling and predicting the helpfulness of online reviews," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM)*, IEEE Computer Society, pp. 443-452, Washington, DC, USA, 2008.

[5] O. Tsur and A. Rappoport, "Revrank: A fully unsupervised algorithm for selecting the most helpful book reviews," in *E. Adar, M. Hurst, T. Finin, N. S. Glance, N. Nicolov, and B. L. Tseng, editors*, ICWSM, The AAAI Press, 2009.

[6] Y. Lu, P. Tsaparas, A. Ntoulas, and L. Polanyi, "Exploiting social context for review quality prediction," in *Proceedings of the 19th International Conference on World Wide Web (WWW)*, ACM, pp. 691-700, New York, NY, USA, 2010.

[7] M. D. Zeiler, "ADADELTA: An Adaptive Learning Rate Method", CoRR abs/1212.5701, 2012.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," in *The Journal of Machine Learning research*, vol. 3, pp.993-1022, 2003.

[9] M. Hoffman, F. R. Bach, and D. M. Blei, "Online learning for latent dirichlet allocation," in *J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, Advances in Neural Information Processing Systems 23*, Curran Associates, Inc, pp. 856-864, 2010.

[10] Yelp dataset challenge, http://www.yelp.com/dataset_challenge, April, 2015.

[11] K. A. Ganesan, C. X. Zhai, "Opinion-Based Entity Ranking," in *Information Retrieval*, Springer Link, vol. 15, issue 2, pp. 116-150, 2012.

[12] Pedregosa et al, "Scikit-learn: Machine Learning in Python," in *JMLR 12*, pp. 2825-2830, 2011.

[13] Chainer, A Powerful, Flexible, and Intuitive Framework of Neural Networks, http://chainer.org/, July, 2015.