Short Text based Cooperative Classification for Multiple Platforms

Mingzhu Li
School of Computer Science and Technology
Shandong University
Jinan, China
limingzhu7@gmail.com

Lihua Chen
Shandong Dazhong Information Industry Co., Ltd
Jinan, China
chenlihua@dzwww.com

Tianyuan Liu
School of Software College
Shandong University
Jinan, China
zodiacg@foxmail.com

*Yuqing Sun
School of Software College
Shandong University
Jinan, China
sun_yuqing@sdu.edu.cn

***Abstract*—With the popularity of electronic commerce, there are increasing requirements on product comparison services, which collect the similar products information on different platforms for a user reference. Since there are a large quantity of products on each platform, it is necessary to classify the products based on their short descriptions and to learn the relationships between the different categories on multiple platforms. In this paper, we propose the Rectified Topic Classification model to classify products into hierarchical categories based on their short text descriptions. We adopt the topic model to capture the latent features of products from the noisy short descriptions generated by merchants. To reduce the uncertainty of the inferring topic features of a new product, we invoke the topic model several times to get a set of probabilistic feature results and adopt the convolutional neural network for classification. To learn the correlations between two platform categories, the mapping matrix is learned by using a set of seed products. We crawled several real datasets from popular e-commerce platforms and perform experiments to verify our methods. The results show that our method outperforms the related methods.**

***Index Terms*—product classification, topic model, hierarchy**

I. Introduction

There are many E-commerce platforms that provide convenient services for customers. To help users choose their preferred products, there are increasing requirements on product comparison services, which collect the similar products information on different platforms. Since there are a large quantity of products on each platform and their categories are different, it is necessary to classify the products against their platform categories and to learn the relationships between categories.

There are many new products released online everyday. Since a product description often contains much information and pictures, it is impossible to classify a product by the whole description, especially for multiple e-commence platforms. Instead, the comparison service only consider the title of a product for classification, which is often the short text containing some technical key words. For example, a cell phone title 'Apple iPhone 7 128 GB Unlocked, Black US Version' involves a few topics like the brand, technical settings, version and etc. The product description texts are submitted by merchants without any restrain on text styles. There may exist some unstandard words that make the classification difficult. In this paper, we would solve the problem of short text based cooperative classification for products on multiple platforms.

There is another challenge for this problem. The product categories on a platform are often hierarchically organized and different with others, which resides on two aspects: categories and structures. That is to say the same product might be classified into two categories on different platforms, and the products in the same category on two platform are not exactly the same. To achieve cross platform product comparison, it is necessary to learn the relationships between platform-dependent categories.

A basic task of text classification is to learn the representation of text feature. There are usually two ways: the direct key word vector against a vocabulary [1] [2], and the embedding method that model a text as distributed representations [3]. Since the title of a product often contains new user-defined and technical key words, the vocabulary increases very fast such that a product vector becomes high dimensional and sparse, or the embedding vectors need re-training, which results in the classification on product short text description difficult.

In this paper, we propose the Rectified Topic Classification (RTC) model to classify products into hierarchical categories based on their short text descriptions. Considering the noisy and non-standard text descriptions, we adopt the topic model to capture latent features for products. To reduce the uncertainty of the generated topics for an instance, we invoke topic model several times to get a set of probabilistic distributions on topics. And the invocation results for an instance are reshaped as an image to feed the convolutional neural network for the category inference. We introduce the mapping matrix to learning the correlations between two platform categories, which is learned by using a few of seed products.

The rest of this paper is organized as follows. In section II, we review the related works. Section III presents the problem definition and the model description. Experiments are performed on four real datasets, presented in section IV.

* Corresponding author. This work is supported by the National Natural Science Foundation of China (91646119), the Major Project of NSF Shandong Province(ZR2018ZB0420), the National Key Technology Research and Development Program(2018YFC0831401), the National Development and Reform Commission Project (2017XXH001-04), the Key Research and Development Program of Shandong province(2017GGX10114) and SAICT Expert Program.

Finally, section V concludes the paper.

II. Related Works

A. Text based Classification

The most related work is the text classification. Naive Bayes is widely used for text classification, which is based on the conditional probabilities of features belonging to a class [4]. The support vector machine (SVM) is also suitable for text classification on account of the kernel functions and linear threshold functions [5] [6] [7]. CNN is applied for automatic text categorization of IT tickets [8]. Documents are usually represented as feature vectors against the vocabulary of key words [1] [2]. The embedding methods, for example, the doc2vec method [3], are also adopt for text representation. Since in this problem the titles of products often contain many new personalized and technical key words, the vocabulary increase very fast such that the vector for a product becomes high dimensional and sparse. In addition, there are inadequate product attributes in the titles, such as usage and material, such that these methods are not applicable.

The multi-label classification problem is also related to our work, which usually learns a binary classifier for each class, and make the final decision by integrating all the binary classifiers' advice. A general framework is proposed, in which a binary classifier is constructed for each label and the correlation information among multiple labels is captured by a low-dimensional subspace shared among all labels [9]. But it is impractical to develop binary classifiers when there are a large number of classes involved. A parametric mixture model is developed to label web pages [10]. ML-KNN model combines k-Nearest Neighbor and the MAP to solve the multi-label text classification [2].

In the web page categorization, external knowledge such as the relevance of linked page to the linking page [11], and hypertext structure and hyperlinks [12], are usually involved to improve the model effectiveness. But there is no external knowledge we can utilize in product short text descriptions such that they are not suitable for our problem.

B. Topic Models for Text Classification

The topic models are also related to our work. The latent dirichlet allocation (LDA) is a dimensionality reduction method for text representation [13]. LDA generates the latent topics distribution of a document via Gibbs sampling and the Dirichlet distribution, which represents the document in a semantic space. It infers a new document as a probabilistic distribution over the topics [14] [15]. Supervised topic model constructs implicit correlations between labels and latent topics to categorize documents [16]. For disease categorization based medical text, a 'universal dataset' is involved to discover a rich set of hidden topics with LDA, and a classifier is developed on both labeled training data and hidden topics [17]. For multi-label text classification, the Labeled-LDA method constrains the latent dirichlet allocation by defining a one-to-one correspondence between latent topics and user tags [18], but it can not handle the hierarchically organized categories.

C. Cross-domain Recommendation and Classification

Cross-domain recommendation is also related to our work. EMSDR algorithm learns the user relation matrix and latent factor matrix between users and items for cross-domain movie recommendation [19]. A two-layer convolutional neural network is proposed for cross-domain product review sentiment classification [20]. The hierarchical attention networks are adopted for cross-domain sentiment classification [21]. A domain adaptation technique is proposed to solve the cross-domain text classification [22]. And the most related studies to our work is the cross-domain multi-label text classification [23]. This study focuses on cross-domain text classification, where different label set occurred in multiple domains. But the hierarchical structures among categories are also neglected.

III. Hierarchical Classification Based on Product Short Text Description

In this section, we discuss how to classify a product against different hierarchical categories based on short text description. We would first present the notions and constraints in product classification. Then we present the Rectified Topic Classification model for short text based product classification. By using seed products, we learn the mapping matrix between two platform-dependent categories for cross domain inference.

A. Notions and Constraints

A product title is a sequence of words. Let **x**= $(w_1, w_2, ..., w_{|l|})$ denote a product title after word segmentation, where w_i is a word in the vocabulary, l is the length of the title.

Generally products are organized as the hierarchical categories on e-commerce platforms in the form of category trees, where each node is a category label and each branch represents the hierarchical relationships between a category and its sub-categories. Let H denote a category tree, and $|H|$ is the number of categories on H. Each category in H is assigned a sequential number. The classification of a product is the label path of categories from the root to a leaf in the category tree, which map to the labels from coarse-grained to fine-grained. For a product, the classification on the category tree is denoted by **y**= $(y_1, y_2, ..., y_{|H|})$, where $y_i = 1$ represents the product is assigned the ith category label on H, and $y_i = 0$ otherwise.

The label path for a product should satisfy two hierarchical constraints. The first is the inheritance constraint. Let y_c denote a sub-category of category y_i. Given a product, if it is not assigned the category label y_i, it should not be assigned any sub-category label y_c either. The second is the succession constraint. If a product is assigned the category y_c, it should be assigned its father category y_i. These constraints are formalized as follows:

Given a category label y_i and a sub-category y_c of y_i, the following conditions hold.

$$\begin{cases} if \quad y_i = 0, \quad then \quad y_c = 0 \\ if \quad y_c = 1, \quad then \quad y_i = 1 \end{cases} \tag{1}$$

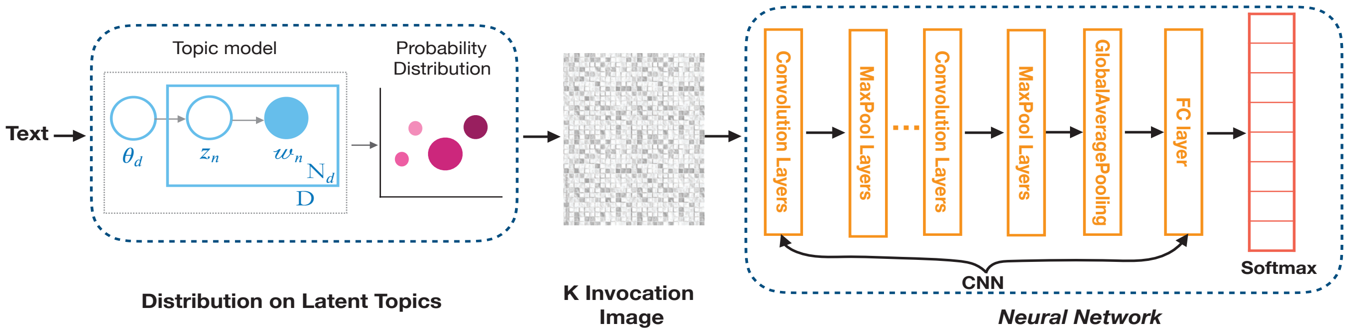

Fig. 1. The framework of RTC Model.

B. Problem Definition and the proposed Model

Definition 1 (Product Classification)**.** *Given a product short text description* $\boldsymbol{x}=(w_1, w_2, ..., w_{|l|})$*, and the category tree* H*, the Product Classification Problem is to find the label path on* H *for* x*, where* $\mathbf{y}=(y_1, y_2, ..., y_{|H|})$ *satisfying the hierarchical classification constraints.*

To solve the product classification problem, we propose the Rectified Topic Classification (RTC) model and present the framework in Figure 1. There are three parts in the RTC model. The first part is the topic representation model, by which we learn the latent features from short text descriptions of products. The second is the probabilistic representation part. Since the topic model is a generative model, it is uncertain on the generated topics by once invocation. Thus we invoke the topic model k-times for a new instance $\mathbf{x}$ to get a set of probabilistic distributions on topics so as to reduce the uncertainty. The invocation results are reshaped as an image for further analysis. The third part is the neural network to infer the categories of a product based on the probabilistic representation. We would present the details in the following discussion.

Firstly, we adopt the topic model to learn the latent features from product descriptions. Considering the flexible and ambiguous words, we adopt the Latent Dirichlet Allocation (LDA) method to extract latent topics features from the short text descriptions. LDA assumes that the documents are generated under the document-topic distributions θ_d and the word-topic distributions z_n [13]. Due to the limitation of the space, we omit the details of LDA. Readers are suggested to refer the document [13] for details.

After the convergence of parameters in the learning process, the document-word distributions and the word-topic distributions are used to infer the topics for a new document. This process is the same as the learning process. Generally, a given product title consists of a set of key words that describe different aspects of the product. These information involve a few topics. This short text is similar to a document in the LDA model. Thus we reduce the extraction of latent features from a product to a latent topics probability distribution, denoted by $\mathbf{t} = (t_1, ..., t_m)$, where m is the number of latent topics in LDA and $t_i \in [0, 1]$ is the probability for the ith topic.

The second part is to form the feature representation of products. Since LDA is a generative model, for a new sample, it probabilistically generates a topic inference result in once invocation. That is to say, for the same sample, we might get different results on topic inference by several invocations. To reduce the uncertainty of topic inference, we invoke the topic model k times for a sample and have the resulted topic distributions. Let $\mathbf{T} = (\mathbf{t}_1, ..., \mathbf{t}_k)' \in R^{k\times m}$ denote the latent topics probability matrix, and $\mathbf{t}_i = (t_1, ..., t_m)$ is the resulted topic vector of the ith invocation. We reshape the feature representation as an image to feed the convolutional neural network afterwards. Each element $\mathbf{T}_{i,j}(1 \le i \le k, 1 \le j \le m) \in \mathbf{T}$ is scaled to an integer in range $[0, 255]$. The scaled matrix is saved as a gray image and is further resized as an RGB image to feed the neural network afterwards.

The third part is the neural network for product classification, which consists of three steps. The first includes several convolution layers and max-pooling layers similar as the VGG16 network [24]. There are more than one 3×3 filters to capture more features with fewer parameters. Max-pooling layers are added to reduce the feature dimension. We adopt the weights pre-training on ImageNet. The second is the GloabalAveragePooling layers for learning the correspondence between a feature map and categories. The third is the several fully-connected layers with the softmax activation functions. The channels are set the same number of categories. The product inference results are the outputs of these softmax functions, denoted by $\hat{\mathbf{y}} = (\hat{y_1}, \hat{y_2}, ..., \hat{y}_{|C|})$, where $\hat{y_i}$ is the output of the ith softmax function, and $|C|$ is the number of leaf nodes in a category tree. We adopt the categorical cross-entropy as the loss function of the convolutional neural networks, formalized as:

$$loss = -\sum_{i=1}^{n}\sum_{j=1}^{|C|} y_j^i log\hat{y_j^i} \tag{2}$$

where n is the number of samples, and $y_j^i, \hat{y_j^i}$ are the true value and the predicted value on the jth category for product i, respectively.

With these three parts, we infer a product categories based on its short text description. The topics based representation part extracts the latent features of a product and overcomes the influence of noisy descriptions. The feature representation part gets rid of the uncertain inference results. By reshaping the invocation results, the neural network part infers the categories of products.

C. Category Mapping between Category Trees

To compare the same product on different platforms, we design the category mapping function between category trees. Let H^A, H^B denote the category tree on platform A and B, respectively. Given a product short text description **x**$=(w_1, w_2, ..., w_{|l|})$ on platform B, we predict its category distribution $\hat{\mathbf{y}} = (\hat{y_1}, \hat{y_2}, ..., \hat{y}_{|H^B|})$ against H^B by our model. The reason we choose the probability distribution as the form of $\hat{y}$ for mapping learning is that it contains more useful information than one-hot category representation. In order to find the classification results $\mathbf{y}'= (y'_1, y'_2, ..., y'_{|H^A|})$ against H^A for **x**, we introduce the mapping matrix, denoted by $\mathbf{M}^{|H^A|\times|H^B|}$, to learn the correlations between the two platform categories, i.e $\mathbf{y}' = \mathbf{M} \cdot \hat{\mathbf{y}}$.

In order to learn the correlations between categories, we introduce the seed dataset that are known the same products on both platforms, denoted by O. Based on these seed products' corresponding classification information on both platforms, we have positive samples. Those products that do not exist on either platform are considered as negative samples.

Given the category trees H^A, H^B on platform A and B, the classification information of a seed product $c \in O$ is denoted by y_c^A and y_c^B respectively, where $y_c^A = (y_1^A, ..., y_{|H^A|}^A)$ and $y_c^A = (y_1^B, ..., y_{|H^B|}^B)$. And $y_{c'}^B$ is the label path of product $c' \notin O$ on platform B. According to the positive and negative mapping relationships, the mapping matrix **M** is learned via the following optimization objective:

$$\min \sum_{\substack{c \in O \\ c' \notin O}} \{\alpha \cdot d(y_c^A, M \cdot y_c^B) - \beta \cdot d(y_c^A, M \cdot y_{c'}^B) + \gamma \|M\|^2\} \tag{3}$$

where α, β and γ are the super parameters to adjust the global optimization objective, and $d(\boldsymbol{.})$ is a distance function. Suppose we adopt the subtraction of Frobenius norms (Euclidean distance is also suitable). The objective is formalized as follows:

$$\begin{aligned} d(y^A, M \cdot y^B) &= \|y^A - M \cdot y^B\|_F^2 \\ &= \sum_{i=1}^{|H^A|} [y_i^A - (M_{i,1} \cdot y_1^B + ... + M_{i,|H^B|} \cdot y_{|H^B|}^B)]^2 \\ &= \sum_{i=1}^{|H^A|} [y_i^A - (M_{i,.}) \cdot y^B]^2 \end{aligned} \tag{4}$$

The objective function minimizes the mapping distance of seed products classification and maximizes the distance of unrelated classification. The argument $\mathbf{M}^*$ is learned by optimizing the objective function 3.

$$\begin{aligned} M^* = argmin \sum_{\substack{c \in O \\ c' \notin O}} \{\alpha \cdot \|y_c^A - M \cdot y_c^B\|_F^2 \\ -\beta \cdot \|y_c^A - M \cdot y_{c'}^B\|_F^2 + \gamma \|M\|^2\} \end{aligned} \tag{5}$$

TABLE I
DESCRIPTIONS OF THE FOUR DATASETS

Dataset	Description	# Products	# Categories
Joybuy	The Joybuy dataset contains products organized by a 3-depth category tree.	379K	708
Amazon	The product in Amazon is assigned to more than one same-depth category.	44K	224
TMall	All the products in TMall are digital products.	9K	27
Joybuy Digital	This dataset only contains digital products on Joybuy.	5K	58

We solve this optimization problem by gradient descent algorithm [25]. And the related gradient function for **M** is formulated as follows:

$$\frac{\partial \|y^A - \mathbf{M} \cdot y^B\|^2}{\partial \mathbf{M}} = -2(y^A - \mathbf{M} \cdot y^B) \cdot y^B \tag{6}$$

IV. Experiments

A. Datasets and Baseline Methods

We evaluate our methods on four real datasets, which are crawled from three platforms: Joybuy, Amazon and TMall platforms, which are the world wild large e-commerce platforms. Each dataset contains short text descriptions and classification information of products. Table I describes the four datasets in detail.

Figure 2 presents the statistics on Joybuy and Amazon datasets, respectively, including the length of product descriptions and the product amount assigned the same class label. The length of products descriptions follows the long-tail distribution, and most products descriptions have no more than 20 words. The number of products in each class is imbalanced.

According to the above statistics, we remove the products satisfying the following restrictions:

1) Short text descriptions or corresponding classification information were missing
2) The number of products in a category < 150
3) The length of product text descriptions < 10

To evaluate the performance of our method on product classification, the baseline methods are the following:

1) Labeled-LDA: Labeled-LDA is a supervised topic model, which defines the correspondence between latent topics and labels to solve text classification.

2) Support Vector Machine: We adopt LDA to extract the latent topics as features and implement SVM to infer the categories of products.

B. Products Classification on a Single Platform

We first verify the product classification performance of the RTC model and other comparison methods. The adopted metrics are the accuracy and AUC-score. In the experiments of SVM, we extract latent topics from products descriptions as features to implement SVM for product classification. The

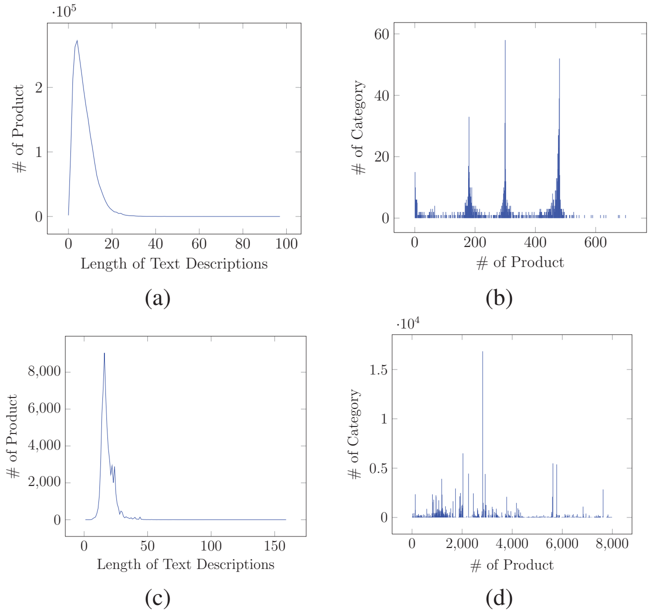

Fig. 2. Statistics on Joybuy and Amazon dataset. Figure (a) and (c) show the long-tail distribution of product descriptions length on Joybuy and Amazon dataset, respectively. Figure (b) and (d) give the statistics on product amount in each class on Joybuy and Amazon dataset, respectively.

number of latent topics on each dataset is set the same with our method.

The results are listed in table II, which shows that the RTC model outperforms other methods. The auc exceed 0.95 on all dataset. On the Joybuy Digital dataset, the accuracy of RTC model reaches nearly 0.78, which is 8.3% higher than SVM. These results show that the several topic model invocations enrich the product feature representation.

C. Evaluation on Category Mapping

To verify the mapping results between two category trees, we select the Joybuy Digital and TMall datasets for experiments. We aim to map the products on TMall to Joybuy, i.e. Joybuy is considered as platform A and TMAll is platform B. The classification information on Joybuy are treated as ground-truth. The seed dataset consists of the products on both platforms. Since for the same product x the quantity on A is more than on B, denoted by A^x and B^x, respectively, and these instances may be assigned different labels due to the providers' choices. So, we introduce a ranking metric to evaluate the category mapping performance, denoted by $Ranking^x$.

$$Ranking^x = \frac{1}{n}\sum_{i=1}^{n} rank_i \tag{7}$$

where n is the number of the same instances on platform A for sample x on B, and $rank_i$ is the label ranking for instance $i \in A^x$ in the mapping results of x. The accuracy of top-k $Ranking$ is illustrated in figure 3, which reflects the uncertain of labels for the same products.

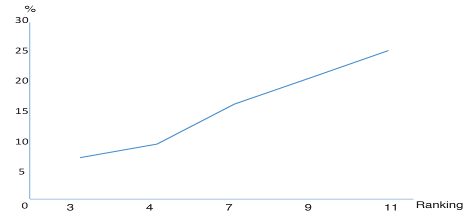

Fig. 3. The Accuracy of Top-k $Ranking$.

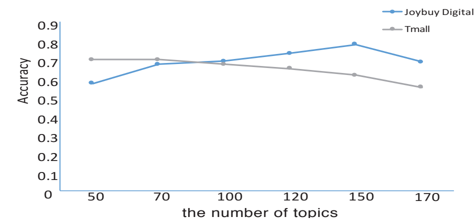

Fig. 4. Influence of parameters in RTC model on Joybuy Digital and TMall dataset. X-axis is the number of topics in LDA , and Y-axis is the accuracy of the RTC model results.

D. Influence of Parameter Settings

We experiment with different topic number setting in LDA, and analyze the influence to the product classification effectiveness. As shown in figure 4, the accuracy goes up with the increasing topic number on Joybuy Digital dataset, because more topics capture more efficient features from product description. And the accuracy decreases when the topic number is larger than 150. This may because some topics may involve redundant features. It is interesting to notice that, our method performs best with 150 and 70 topics on Joybuy Digtal dataset and TMall dataset, since more topics are needed to capture useful information for more categories.

E. Semantics Analysis on Latent Topics

For better understanding of latent topics features, we analyze the relations between representative words in the same topics. Table III shows the representative words on 5 topics. From the results, we can see the representative words in the same topic have close relationships. For example, in topic 4 of the Amazon dataset, Fujifilm is a famous brand of instant camera. SSD and CPU in are accessories of computers. The products and its brands information occur in the same topic frequently, such as the Apple and iphone, Nikon and camera. Further more, Inspiration is the computer family name of Dell. Kingston and WD are brand names of portable data storage products. They also shows the close correlations of words in the same topic.

V. Conclusion

In this paper, we solve the problem of cooperative classification based on products short descriptions for multiple platforms. We propose the Rectified Topic Classification model to classify a product based on the short text description, which captures the latent features from noisy short text generated by merchants and infer product categories based

TABLE II
PERFORMANCE COMPARISON OF PRODUCTS CLASSIFICATION.

Method	Tmall		Joybuy		Joybuy Digital		Amazon	
	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC	Accuracy	AUC
RTC	**0.698**	**0.953**	**0.552**	**0.99**	**0.778**	**0.981**	**0.330**	**0.970**
Labeled-LDA	0.221	0.60	0.055	0.527	0.460	0.662	0.137	0.563
SVM	0.350	0.662	0.003	0.500	0.713	0.821	0.065	0.526

TABLE III
REPRESENTATIVE WORDS IN LATENT TOPICS.

Dataset	Topic-1		Topic-2		Topic-3		Topic-4		Topic-5	
	Word-1	Word-2	Word-1	Word-2	Word-1	Word-2	Word-1	Word-2	Word-1	word-2
TMAll	Watch	Android	Xiaomi	mAh	Apple	iphone	Huawei	Honor	Nikon	Camera
Joybuy	WD	Element	HDMI.	Monitor	Kingston	WD	Travel	Cushion	Fuji	film
Joybuy Digital	Inch	i7	HP	Cartrige	Dell	Inspiration	USB	Win10	SSD	CPU
Amazon	Portable	Charger	Kindle	Paperwhite	Cervical	Neck	Fuji	camera	LED	Light

on the probabilistic feature representation. We introduce the mapping matrix to learn the correlations between two platform categories by using seed products. Experiments results on four real dataset show that our methods outperform the related methods. We also analyze the influence of parameter settings and the semantics of latent topics.

REFERENCES

[1] G. Mujtaba, L. Shuib, R. G. Raj, M. A. Al-Garadi, R. Rajandram, and K. Shaikh, "Hierarchical text classification of autopsy reports to determine mod and cod through term-based and concepts-based features," in *Industrial Conference on Data Mining*, pp. 209–222, Springer, 2017.

[2] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[3] H. Lee and Y. Yoon, "Engineering doc2vec for automatic classification of product descriptions on o2o applications," *Electronic Commerce Research*, vol. 18, no. 3, pp. 433–456, 2018.

[4] W. Zhang and F. Gao, "An improvement to naive bayes for text classification," *Procedia Engineering*, vol. 15, no. 1, pp. 2160–2164, 2011.

[5] S. Sun and X. Gu, "Support vector machine equipped with deep convolutional features for product reviews classification," in *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pp. 130–135, IEEE, 2017.

[6] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," *Machine learning: ECML-98*, pp. 137–142, 1998.

[7] H. Yao, B. Zhang, P. Zhang, and M. Li, "A novel kernel for text classification based on semantic and statistical information," *Computing and Informatics Computers and Artificial Intelligence*, vol. 37, no. 4, pp. 992–1010, 2018.

[8] J. Han and M. Akbari, "Vertical domain text classification: Towards understanding it tickets using deep neural networks.," pp. 8202–8203, 2018.

[9] S. Ji, L. Tang, S. Yu, and J. Ye, "Extracting shared subspace for multi-label classification," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 381–389, ACM, 2008.

[10] N. Ueda and K. Saito, "Single-shot detection of multiple categories of text using parametric mixture models," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 626–631, ACM, 2002.

[11] Y. Yang, S. Slattery, and R. Ghani, "A study of approaches to hypertext categorization," *Journal of Intelligent Information Systems*, vol. 18, no. 2-3, pp. 219–241, 2002.

[12] H.-J. Oh, S. H. Myaeng, and M.-H. Lee, "A practical hypertext catergorization method using links and incrementally available class information," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 264–271, ACM, 2000.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[14] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pp. 90–94, 2012.

[15] D. Quercia, H. Askham, and J. Crowcroft, "Tweetlda: supervised topic classification and link prediction in twitter," in *Proceedings of the 4th Annual ACM Web Science Conference*, pp. 247–250, ACM, 2012.

[16] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *Advances in neural information processing systems*, pp. 121–128, 2008.

[17] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi, "Learning to classify short and sparse text & web with hidden topics from large-scale data collections," in *Proceedings of the 17th international conference on World Wide Web*, pp. 91–100, ACM, 2008.

[18] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1*, pp. 248–256, 2009.

[19] T. Man, H. Shen, X. Jin, and X. Cheng, "Cross-domain recommendation: An embedding and mapping approach," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2464–2470, AAAI Press, 2017.

[20] X. Wei, H. Lin, Y. Yu, and L. Yang, "Low-resource cross-domain product review sentiment classification based on a cnn with an auxiliary large-scale corpus," *Algorithms*, vol. 10, no. 3, p. 81, 2017.

[21] K. Zhang, H. Zhang, Q. Liu, H. Zhao, H. Zhu, and E. Chen, "Hierarchical attention transfer network for cross-domain sentiment classification," pp. 5852–5859, 2018.

[22] H. S. Bhatt, D. Semwal, and S. Roy, "An iterative similarity based adaptation technique for cross-domain text classification.," in *CoNLL*, pp. 52–61, 2015.

[23] H. S. Bhatt, M. Sinha, and S. Roy, "Cross-domain text classification with multiple domains and disparate label sets," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 1641–1650, 2016.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.