Proceedings of the 2020 Pre-ICIS SIGDSA Symposium

Special Interest Group on Decision Support and Analytics (SIGDSA)

Winter 12-13-2020

Deep Learning-Based User Feedback Classification in Mobile App Reviews

Zhilei Qiao

Alan Wang

Alan Abrahams

Weiguo Fan

Deep Learning-Based User Feedback Classification in Mobile App Reviews

Completed Research Paper

Zhilei Qiao
University of Alabama at Birmingham
qiaozl@uab.edu

Alan Wang
Virginia Tech
alanwang@vt.edu

Alan Abrahams
Virginia Tech
abra@vt.edu

Weiguo Fan
University of Iowa
Weiguo-fan@uiowa.edu

Abstract

As online users are interacting with many mobile apps under different usage contexts, user needs in an app design process has become a critical issue. Existing studies indicate timely and constructive online reviews from users becomes extremely crucial for developers to understand user needs and create innovation opportunities. However, discovering and quantifying potential user needs from large amounts of unstructured text is a nontrivial task. In this paper, we propose a *domain-oriented deep learning approach* that can discover the most critical user needs such as app product new features and bug reports from a large volume of online product reviews. We conduct comprehensive evaluations including quantitative evaluations like F-measure a, and qualitative evaluations such as a case study to ensure the quality of discovered information, specifically, including the number of bug reports and feature requests. Experimental results demonstrate that our proposed supervised model outperforms the baseline models and could find more valuable information such as more important key words and more coherent topics. Our research has significant managerial implications for app developers, app customers and app platform providers.

Key Words: Online Reviews, Demand-side, Classification, Deep Learning, Mobile Apps

Introduction

Due to the strong competition in the mobile app industry, app quality has become an essential factor for apps to gain a competitive advantage in the mobile app market (Chen et al. 2014). The mobile app market is growing rapidly, with millions of apps and developers, billions of users, and billions of dollars in revenue. For example, the Apple App Store, one of the most competitive app markets, offered 500 apps upon its initiation in 2008 and had over 2.2 million apps by 2017 (Lai et al. 2018). Given the large volume of reviews available in the Apple App Store (Zhou et al. 2018), it is important for app developers to efficiently extract and understand user needs from user reviews (Aral et al. 2013; Chen et al. 2014; Nayebi et al. 2016).

Compared with the bug reporting and feature request mechanisms used in traditional software development, there are two outstanding challenges to extracting valuable user feedback from unstructured online reviews. First, only around one-third of app reviews contain objective statements (Abrahams et al. 2013; Law et al. 2017; Oh et al. 2013; Winkler et al. 2016). Second, manually processing a large volume of unstructured user reviews and extracting potential user needs from those reviews can be tedious. Thus, it is more efficient and desirable to automatically, rather than manually, extract user needs from unstructured online reviews.

To overcome these existing challenges and effectively extract user feedback from app reviews, this research proposes a deep learning-based opinion classification method to identify user needs from online reviews. The proposed method improves existing text classification methods by capturing the semantic context of words using deep text learning. In addition, in order to balance result interpretability and analysis granularity, the proposed method is implemented at the sentence level instead of the review or document level. The method helps mobile app developers automatically extract user needs from a large volume of online reviews with greater effectiveness than traditional machine learning algorithms.

The rest of the chapter proceeds as follows. Section 2 reviews related work, and Section 3 states the research objective. Section 4 presents the proposed domain-oriented, deep learning method for opinion mining, and Section 5 describes the experiment and evaluation results. The last section concludes the study's findings, discusses the limitations of the study, and makes suggestions for future work.

Related Work

In this section, the literature related to customer value co-creation, text classification for opinion mining, and text analytics of mobile app reviews, is reviewed.

Customer Value Co-Creation for Product Quality

Co-creation refers to a joint creation process that involves both the company and the customers (Prahalad and Ramaswamy 2000, 2004; Ramaswamy and Prahalad 2004). Presenting a holistic perspective of value co-creation, Prahalad and Ramaswamy (2000) document the transformation of customers from "passive listeners" to "active players" over time, which is the foundation of value co-creation.

In the mobile app industry, online customer reviews are an important channel through which customers and firms can communicate regarding product quality. In fact, online customer reviews have been a significant driving force in the evolution of several apps (Pagano and Maalej 2013; Qiao et al. 2018; Zhou et al. 2018). As the number of online app reviews increases at an unprecedented speed, many app firms seek to create business opportunities by discovering business values from the reviews (Chen et al. 2014; Maalej et al. 2017; Panichella et al. 2015; Di Sorbo et al. 2016). The content of online reviews is mostly unstructured text that is often difficult to manually analyze when the volume is large. Therefore, it is necessary to develop effective and efficient ways to automatically process a large volume of text-based user reviews and extract valuable user opinions for customer value co-creation.

Information Types in App Reviews

Maalej and Nabil (2015) categorize app reviews into four basic types: bug reports, feature requests, user experiences, and ratings. Bug reports refer to the problems with the app that should be fixed, such as an erroneous behavior, a performance issue, or an unexpected crash. Feature requests describe new features proposed by consumers, including new functions. User experiences are the documentation of the user's interaction with the app, while ratings are sentiment text represented by different numbers of stars. This study focuses on bug reports and feature requests because they contain specific user feedback and can be used to improve product design. User experiences and ratings are grouped together as other types because they are not directly related to the identification of user needs. Table 1 presents some examples of the different types of app reviews. As the examples indicate, a user review may consist of different types of information, with each sentence focusing on one specific information type. Therefore, app reviews should be analyzed at the sentence level.

Table 1 Examples of Different Types of App Reviews

No.	Review Content	Information Types
1.	The clock doesn't keep time like a regular clock.	Bug Reports
2.	Only problem I have with this game is that it crashes too much.	Bug Reports
3.	The connection for the game is kind of sucky.	Bug Reports

4.	I know that they already have that in Need for Speed Hot Pursuit, but I was hoping for something more diverse.	Feature Request
5.	Also Please add a multi player.	Feature Request
6.	Add Chevy cars and trucks too!	Feature Request
7.	I like this game a lot.	Other Types
8.	Great game!	Other Types

Sentence-Level User Feedback Classification

Several studies have proposed methods for sentence-level user feedback classification (Büschken and Allenby 2016; Kim 2014; Täckström and McDonald 2011). Liu (2012) explains that sentence-level classification is appropriate to classify objective information because document-level opinion mining is too coarse for applications, whereas the results of aspect-level or phrase-level opinion mining may be difficult to interpret. There are several studies in the sentence-level objective information classification literature (Liu et al. 2005; Moghaddam 2015; Stieglitz and Dang-Xuan 2013a; Wang et al. 2010). These approaches mostly focus on sentiment analysis and categorize a given text as either positive or negative. Although distinguishing the sentiment of user reviews can help customers make purchasing decisions, it is still challenging to capture objective sentences from these reviews that can provide developers specific suggestions for product improvement.

To address this issue, recent opinion mining research focuses on discovering objective sentences that describe specific product features (Mummalaneni et al. 2018; Wang et al. 2010b) and product defects (Abrahams et al. 2015) from user-generated content. These studies can be categorized into two types: rule-based and machine learning based. Rule-based methods, such as that proposed by Brun and Hagege (2013), manually formulate linguistic rules to extract opinion sentences from customer reviews. Some machine learning-based methods, such as those proposed by Moghaddam (2015) and Galvis, Carreño, and Winbladh (2013), utilize LDA or topic modeling to extract topics from online customer reviews in an unsupervised or semi-supervised manner. LDA, or topic modeling, uses a collection of keywords to represent each topic. However, it is difficult to evaluate the quality of the topics, and the topics (i.e., the collections of words) are difficult to interpret.

Other machine learning methods apply supervised classification algorithms to opinion classification tasks. They extract linguistic features, such as bags-of-words (Pang et al. 2002) and grammatical (e.g., Part-of-Speech Tagging), syntactical (e.g., noun phrases, verb phrases, prepositional phrases), and semantic features (e.g., word-sense) from text and apply classification algorithms, such as logistic regression, decision trees (DTs), multinmial naïve Bayes (MNB), and SVM. These methods consider that the linguistic features that can be extracted from a text are independent of each other. However, the feature extraction process ignores much of the contextual information embedded in sentence structures and word sequences. Thus, the deep learning technique has been introduced to text mining and natural language processing. One of the key components in deep text learning is word embedding, which is a language representation model that can capture the semantic and syntactic similarities between words. Deep learning captures the contextual information around words and the order between them and can help in the interpretation of textual data using a relatively holistic perspective. Moreover, deep learning has shown promising results in natural language processing applications, such as that of Stavrianou and Brun (2012). Last but not least, deep learning for user feedback classification is also understudied.

Deep Learning for Text Classification

Deep learning techniques perform better than traditional machine learning algorithms because they construct features in a hierarchical way: in other words, the higher-level features contain semantic connections extracted from the lower-level features such as words. The theoretical deep learning literature suggests that, in order to learn the various complex functions that can characterize high-level abstractions (e.g., image, language, and audio), researchers may need deep architectures (Bengio et al. 2007). Deep learning can use a broad collection of deep architectures (Bengio 2009), including graphical models with many levels of hidden variables (Hinton and Salakhutdinov 2006), neural networks with several hidden layers (Collobert and Weston 2008), and others (Zhu et al. 2009). The recent surge in deep learning and artificial intelligence research has demonstrated the superiority of deep learning over traditional machine learning techniques in computer vision (Ranzato et al., 2008; Lee et al., 2009; Mobahi et al., 2009), natural language processing (Collobert and Weston 2008; Weston and Besley 2008), and information retrieval (Salakhutdinov and Hinton 2007).

Two variants of deep learning techniques are widely used: a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN). A CNN can efficiently capture the contextual information around words. Specifically, a CNN is used to denote big context sizes, such as unigram, bigram (a two-word sequence), and trigram (a three-word sequence), and to extract salient features within larger contexts through convolution and max-pooling operations. However, a CNN does not consider the order of words in a sentence, which is important for understanding the semantics among the features. To solve this problem, an RNN views the input as a sequential structure and requires a series of linear operations (Wang et al. 2017). An RNN is well designed for sequence modeling. Long Short-Term Memory (LSTM), a variant of an RNN, provides an effective way of sequentially composing the semantic understanding in texts. The key units in LSTM are gates, which are implemented by a sigmoid function. Using sequence data, $\{w_1, \cdots, w_n\}$, the gates can help control how much new information, w_t, from the current step, t, is added, how many long memories from the previous step are needed to establish new memories, and how much information is needed as features to generate the output at the current step. In this way, LSTM can decide the amount of information that can pass through gates automatically and dynamically based on different inputs at different steps. The training process contains sequences that activate the next hidden layer using a previous time step as the input to the current layer to influence predictions at the current time step (Sak et al. 2014). Studies have shown that applying an RNN or a CNN to generic sentence classification demonstrates outstanding performance in terms of classification performance metrics (e.g., F-measure, precision and recall) (Gan et al. 2017; Wang et al. 2017).

Research Objectives

In this research, a deep learning-based user feedback classification framework is proposed for identifying information types from user reviews about mobile apps. Two types of information are emphasized: bug reports and new feature requests, which are helpful for customer value co-creation. Text classification will be conducted at the sentence level because it provides a good balance between analysis granularity and interpretability. It is expected that the deep learning-based approach will outperform existing text classification methods because deep learning can capture more semantic and contextual relationships between words. A comprehensive evaluation will be conducted to evaluate the proposed framework. The framework provides a useful and efficient way for mobile app developers to analyze user feedback from the large volume of online user reviews and maintain their competitive advantages.

Research Design

A Deep Learning-Based, Sentence-Level User Feedback Classification Framework

This section proposes a user feedback classification framework based on deep learning. Text documents, such as user reviews, must be preprocessed before the text analysis. Each review is segmented into individual sentences using a sentence tokenizer. A word tokenizer breaks each sentence into a sequenced collection of words. Punctuations and stop-words[1] are removed. All letters are converted into lower-case letters. The proposed user feedback classification framework consists of three main processing layers: a word-embedding layer, a CNN layer, and an RNN layer. Notations used in the framework are listed in Table 2 and a summarization of the computing process is mentioned in Algorithm 1. Figure 1 illustrates the major processes of the proposed framework.

Table 2 Notations in the Proposed Framework

Notation	Description
V	The vocabulary

[1] https://nlp.stanford.edu/software/

$\vert V \vert$	The vocabulary size
v_i	The word embedding word w_i
E	The word embedding matrix
w_i	The ith word in the input sentence
$\vert s \vert$	The length of the input sentence
S	The input sentence
dim_e	The dimension of word embedding
C_s	The output of convolution layer for the input sentence S
X_s	The output of max pooling for the input sentence S
R_s	The final semantic representation of input sentence S
L_s	The final label of input sentence S

Algorithm 1: Deep learning-based user feedback classification

```
Input: The input sentence contains a series of words: $[w_i, \cdots, w_{|s|}]$, where $w_i$ is chosen from a vocabulary $V$
    1. Represent $w_i$ using its word embedding $v_i$ by looking up word embedding matrix $E$. Define $S = [v_i, \cdots, v_{|s|}]$ as the input sentence embedding matrix $R$ with dimension $dim_e \times |s|$.
    2. Apply CNN to process S to get outputs of convolution $C_s$.
    3. Apply max pooling to process $C_s$ and get $X_s$.
    4. Apply LSTM to process $X_s$ and get $R_s$
    5. Apply argmax function to $R_s$ and get $L_s$ where $L_s[i] = argmax(R[i,:])$
Output: Return $L_s$
```

Experiments

In this section, the experiment used to evaluate the performance of the proposed user feedback classification framework to identify user needs in app user reviews is described. Baseline methods include several traditional text classification methods, such as SVM, k-nearest neighbors (KNN), and random forest (RF). For these baseline methods, the TF-IDF (term-frequency-inversed-document-frequency) vector space model is used as the text representation model, which is commonly used in text classification (Ramos 2003).

Data Description

The Apple App Store offers more than 2.8 million apps and is the largest app store in terms of its total generated revenue (Lai et al. 2018). 18,261,515 app reviews were collected for 4,602 game apps from their date of release to November 29, 2015. Each user review consists of review text content, review time, review rating, review title, app version, reviewer identity, and reviewer country. To overcome potential selection bias, five apps out of the 4,602 game apps and 3,000 user reviews made for the five apps were randomly selected. To obtain the ground truth of user feedback classification, 26 undergraduate and graduate students were recruited to label these reviews with 12,864 sentences. Each sentence could be labeled as three information types: bug fixes, feature requests, and others. The final dataset contained 6,915 sentences that were tagged by at least two taggers. The agreement rate was 79.9%, while the inter-rater reliability score was 74.5% which is fair good (Landis and Koch 1977). The feature request information type was 444 agreed sentence labels, which was the least across the three information types. Therefore, 444 sentences labeled as bug reports and 444 sentences labeled as other types were drawn in order to build a balanced evaluation data set.

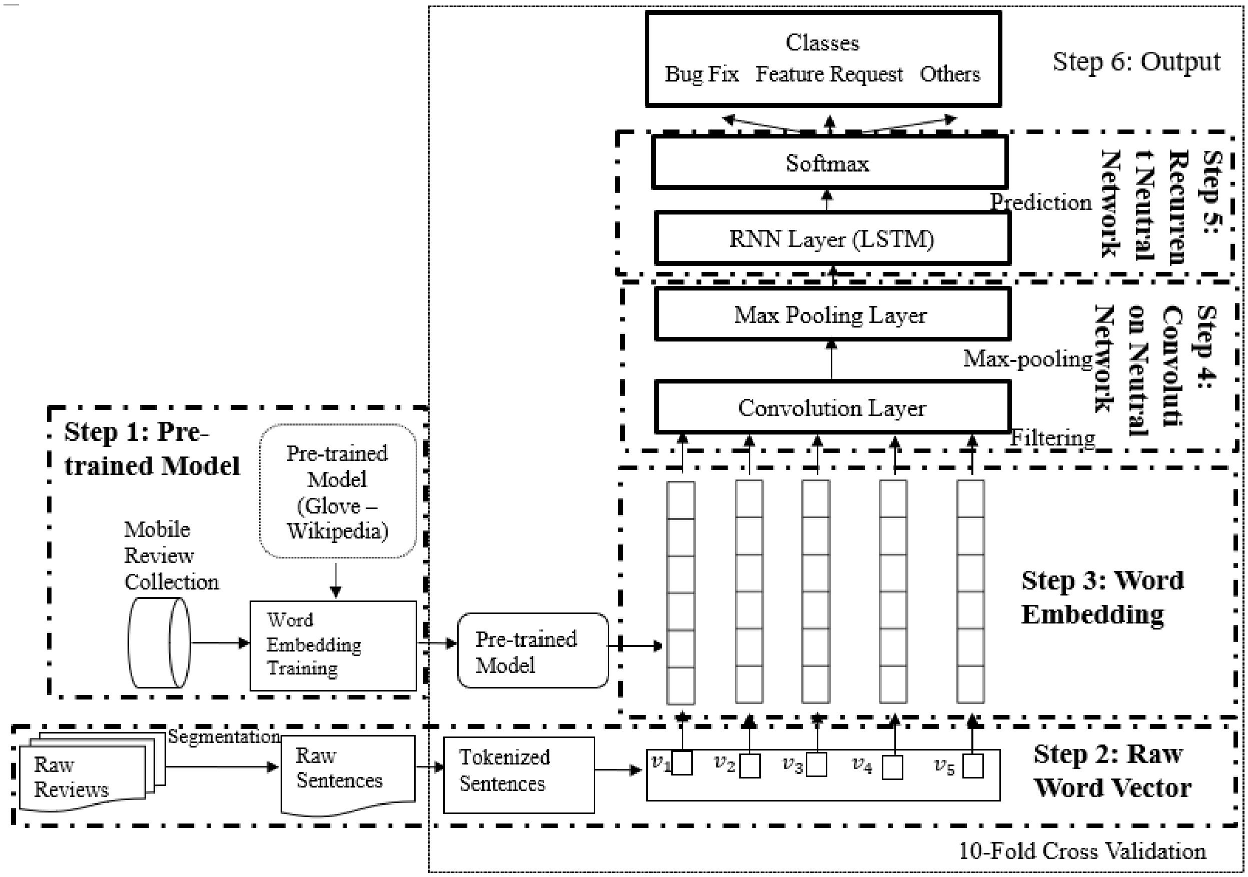

Figure 1 Deep Learning Based User Feedback Classification Framework

Performance Metrics

The performance of the proposed framework was measured using the following four measures: precision, recall, and the F-measure. These measures are broadly used in information retrieval and text mining evaluations (Powers 2011). Precision is defined as the percentage of correctly identified instances in all the instances identified by the framework. Recall is the percentage of the instances that the framework has correctly identified over the total number of relevant instances, and the F-measure is the weighted average of precision and recall.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{1}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{2}$$

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

For the performance evaluation of multi-class classification, the macro-average method was used to calculate the average precision, recall, and F-measure.

$$Recall_{avg} = \sum_n (Recall)\,/n \tag{4}$$

$$Precision_{avg} = \sum_n (Precision)\,/n \tag{5}$$

$$F - measure_{avg} = \sum_n (F - measure)\,/n \tag{6}$$

Experiment

Training Word-Embedding Models

Studies have confirmed that initializing word vectors with pre-trained word embedding can improve the performance of text classification in the absence of a large, supervised training set (Collobert et al., 2011; Socher et al., 2011; Iyyer et al., 2014). Word embedding can be trained using a domain-independent or a domain-dependent corpus, which were both tested in this experiment. First, the publicly available *word2vec* vectors that had been obtained from training using 100 billion tokens from Google News were used. Alternatively, training using word embedding with a domain-specific corpus, 1.8 billion tokens from the app reviews collected from the Apple App Store, was employed. To achieve the optimal training performance, I tested several parameter values used in the proposed framework. For example, window size parameter, which indicates the maximum number of words between the current and predicted word in a sentence, was tested. Its values include 3, 4, and 5 words. The results indicated that the window size was 4 when the training performance (measured by accuracy) was the best. Similarly, the vector size of each word, which means the dimensionality of the feature vectors, was tested. Its values include 100, 200, 300, 400, and 500. The classification performance shows that vector size 300 made the training classification performance the best. The word frequency is a minimum threshold to determine the word as features. The experiment shows when the word frequency was 5, the classification performance was the best. The vectors were trained via the continuous bag-of-words model (Mikolov et al. 2013). Words not occurring in the set of pre-trained words were initialized randomly. The *word2vec genism* library was utilized to train the model, tuning one parameter at a time with the other parameters held constant.

Experimental Results

Table 4-4 summarizes the performance of the benchmark methods and the deep learning-based user feedback classification method. The proposed deep learning-based method outperformed all the traditional text classification methods for identifying bug reports and new feature requests from user reviews. Among these classes, bug reports achieved the best performance, and the F-measure reached up to 0.83. By contrast, the F-measure of the feature request classfication performance was only 0.74. It is possible that the features of the bug fixes were more focused than those of the feature requests. Regarding word-embedding training, the word-embedding models trained by the Wikipedia corpus and app review corpus both achieved better performance than traditional text classification methods. However, the word-embedding model trained using apps reviews achieved better performance than that trained using the domain-independent Wikipedia corpus. This showed that the domain-specific corpus helped build a better word-embedding model than the domain-independent corpus.

Table 0-4 Performance of the Proposed Framework vs. Baseline Methods

Method Name	Information Type	Performance Metrics		
		F-Measure	**Precision**	**Recall**
SVM	Bug Report	0.762	0.726	0.802
	Feature Request	0.638	0.679	0.609
	Others	0.718	0.716	0.722
	Combined	0.706	0.711	0.708

KNN	Bug Report	0.645	0.632	0.664
	Feature Request	0.572	0.519	0.644
	Others	0.589	0.711	0.507
	Combined	0.604	0.626	0.603
RF	Bug Report	0.694	0.598	0.846
	Feature Request	0.517	0.763	0.394
	Others	0.690	0.673	0.717
	Combined	0.633	0.682	0.647
DL **(Wikipedia)**	**Bug Report**	0.805	0.824	0.789
	Feature Request	0.710	0.731	0.697
	Others	0.688	0.664	0.718
	Combined	0.738	0.742	0.737
DL **(Mobile app reviews)**	**Bug Report**	**0.828**	**0.815**	**0.846**
	Feature Request	**0.743**	**0.773**	**0.719**
	Others	**0.725**	**0.719**	**0.736**
	Combined	**0.766**	**0.769**	**0.768**

Conclusion and Discussion

In this paper, a new user feedback classification framework was proposed that automatically identified bug reports and feature requests from massive volumes of online app reviews. It was demonstrated that the proposed deep learning-based framework outperformed existing text classification baseline methods.

This study makes several methodological and theoretical contributions. First, the study proposes a novel deep learning framework that incorporates a CNN and an RNN to identify user needs and relevant details from unstructured textual data. The proposed model provides an effective framework to incorporate contextual information (syntactic features and domain background) to uncover sentences related to user needs. Second, the study confirms previous findings that word embedding can achieve better performance in terms of classification accuracy through different configurations of hyper-configurations. Experimental results on word embedding show that the proposed model outperforms the competing traditional classification methods and discovers more meaningful and accurate user needs. Third, this study explores the domain adaption problem, and the results show that the pre-trained word embedding model that is trained on the online app dataset outperforms all the benchmarks.

This research contributes to the rich body of research on customer value co-creation by providing an automated tool to classify user feedback in user reviews. Researchers can use the proposed methodology to identify and understand the user feedback embedded in the large volume of online, user-generated content. Firms facing hyper-competition can use the proposed method to automatically identify product issues from customer feedback and improve their product design by addressing those issues. Hence, managers are urged to see the benefits of quickly understanding customer feedback and transforming collaborative inputs from users into new business opportunities. The proposed method provides a feasible way for users to be involved in value co-creation (Ramaswamy and Prahalad 2004), which can create business value.

This research has significant managerial implications for app developers, users, and platform providers. For example, app developers can use information regarding user needs to improve product quality. By receiving attention from app developers, app customers will be more inclined to contribute valuable feedback regarding the developers' products. App platform providers can design new features to categorize user information and incorporate more innovative information based on app developers' needs and customers' feedback.

Despites its findings and implications, this study has several limitations. First, this research uses only one public data source and only one method of analysis (text analysis). Thus, an empirical study that incorporates other sources of data from manufacturers may yield more valuable and practical insights regarding quality improvement and product innovation opportunities. Second, this study only examines objective information classification problems in user reviews. Future studies should incorporate other perspectives, such as those regarding product advantages, which can also help managers understand customers' preferences and demands and thus allow managers to better position their products in the right customer segments. Third, although this study evaluates the results based on the classification performance of information types, it is still necessary to show qualitative measurements (e.g., key word lists) in the future for practical purposes.

References

Abrahams, A., Jiao, J., and Fan, W. 2013. "What's Buzzing in the Blizzard of Buzz? Automotive Component Isolation in Social Media Postings," *Decision Support Systems* (55:4), pp. 871–882.

Abrahams, A. S., Fan, W., Wang, G. A., Zhang, Z., and Jiao, J. 2015. "An Integrated Text Analytic Framework for Product Defect Discovery," *Production and Operations Management* (24:6), pp. 975–990.

Alves, H., Fernandes, C., and Raposo, M. 2016. "Value Co-Creation: Concept and Contexts of Application and Study," *Journal of Business Research* (69:5), Elsevier, pp. 1626–1633.

Aral, S., Dellarocas, C., and Godes, D. 2013. "Introduction to the Special Issue—social Media and Business Transformation: A Framework for Research," *Information Systems Research* (24:1), INFORMS, pp. 3–13.

Bengio, Y. 2009. "Learning Deep Architectures for AI," *Foundations and Trends® in Machine Learning* (2:1), Now Publishers, Inc., pp. 1–127.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. 2007. "Greedy Layer-Wise Training of Deep Networks," in *Advances in Neural Information Processing Systems*, pp. 153–160.

Brun, C., and Hagege, C. 2013. "Suggestion Mining: Detecting Suggestions for Improvement in Users' Comments," *Research in Computing Science* (70), pp. 171–181.

Büschken, J., and Allenby, G. M. 2016. "Sentence-Based Text Analysis for Customer Reviews," *Marketing Science* (35:6), INFORMS, pp. 953–975.

Chen, N., Lin, J., Hoi, S., Xiao, X., and Zhang, B. 2014. "AR-Miner: Mining Informative Reviews for Developers from Mobile App Marketplace," in *Proceedings of the 36th International Conference on Software Engineering*, ACM, pp. 767–778.

Collobert, R., and Weston, J. 2008. "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning," in *Proceedings of the 25th International Conference on Machine Learning*, ACM, pp. 160–167.

Dickson, P. R., and Ginter, J. L. 1987. "Market Segmentation , Product Differentiation , and Marketing Strategy," (51:April), pp. 1–10.

Galvis Carreño, L. V, and Winbladh, K. 2013. "Analysis of User Comments: An Approach for Software Requirements Evolution," in *Proceedings of the 2013 International Conference on Software Engineering*, pp. 582–591.

Hinton, G. E., and Salakhutdinov, R. R. 2006. "Reducing the Dimensionality of Data with Neural Networks," *Science* (313:5786), American Association for the Advancement of Science, pp. 504–507.

Kim, Y. 2014. "Convolutional Neural Networks for Sentence Classification," *arXiv Preprint arXiv:1408.5882*.

Lai, H., Hsu, J. S.-C., and Wu, M.-X. 2018. “The Impact S of Requested Permission on Mobile App Adoption: The Insights Based on an Experiment in Taiwan,” in *Proceedings of the 51st Hawaii International Conference on System Sciences*.

Landis, J. R., and Koch, G. G. 1977. “The Measurement of Observer Agreement for Categorical Data,” *Biometrics*, JSTOR, pp. 159–174.

Law, D., Gruss, R., and Abrahams, A. S. 2017. “Automated Defect Discovery for Dishwasher Appliances from Online Consumer Reviews,” *Expert Systems with Applications* (67), Elsevier Ltd, pp. 84–94.

Lee, G., and Raghu, T. S. 2014. “Determinants of Mobile Apps’ Success: Evidence from the App Store Market,” *Journal of Management Information Systems* (31:2), Taylor & Francis, pp. 133–170.

Liu, B. 2012. “Sentiment Analysis and Opinion Mining,” *Synthesis Lectures on Human Language Technologies* (5:1), Morgan & Claypool Publishers, pp. 1–167.

Liu, B., Hu, M., and Cheng, J. 2005. “Opinion Observer: Analyzing and Comparing Opinions on the Web,” in *Proceedings of the 14th International Conference on World Wide Web*, ACM, pp. 342–351.

Maalej, W., and Nabil, H. 2015. “Bug Report, Feature Request, or Simply Praise? On Automatically Classifying App Reviews,” in *Requirements Engineering Conference (RE), 2015 IEEE 23rd International*, IEEE, pp. 116–125.

Maalej, W., Nabil, H., and Stanik, C. 2017. “On the Automatic Classification of App Reviews,” *Software Engineering 2017,* Gesellschaft Für Informatik eV.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. 2013. “Efficient Estimation of Word Representations in Vector Space,” *arXiv Preprint arXiv:1301.3781*.

Moghaddam, S. 2015. “Beyond Sentiment Analysis: Mining Defects and Improvements from Customer Feedback,” in *Advances in Information Retrieval*, Springer, pp. 400–410.

Mudambi, S. M., and Schuff, D. 2010. “What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon. Com,” *MIS Quarterly* (34:1), pp. 185–200 (available at http://ssrn.com/abstract=2175066).

Mummalaneni, V., Gruss, R., Goldberg, D. M., Ehsani, J. P., and Abrahams, A. S. 2018. “Social Media Analytics for Quality Surveillance and Safety Hazard Detection in Baby Cribs,” *Safety Science* (104), Elsevier, pp. 260–268.

Nayebi, M., Adams, B., and Ruhe, G. 2016. “Release Practices for Mobile Apps -- What Do Users and Developers Think?,” *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, pp. 552–562.

Oh, J., Kim, D., Lee, U., Lee, J.-G., and Song, J. 2013. “Facilitating Developer-User Interactions with Mobile App Review Digests,” in *CHI’13 Extended Abstracts on Human Factors in Computing Systems*, ACM, pp. 1809–1814.

Pagano, D., and Maalej, W. 2013. “User Feedback in the Appstore: An Empirical Study,” in *Requirements Engineering Conference (RE), 2013 21st IEEE International*, IEEE, pp. 125–134.

Pang, B., Lee, L., and Vaithyanathan, S. 2002. “Thumbs up?: Sentiment Classification Using Machine Learning Techniques,” in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10*, Association for Computational Linguistics, pp. 79–86.

Powers, D. M. 2011. “Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness and Correlation,” Bioinfo Publications.

Prahalad, C. K., and Ramaswamy, V. 2000. “Co-Opting Customer Competence,” *Harvard Business Review* (78:1), pp. 79–90.

Prahalad, C. K., and Ramaswamy, V. 2004. “Co-creation Experiences: The next Practice in Value Creation,” *Journal of Interactive Marketing* (18:3), Wiley Online Library, pp. 5–14.

Qiao, Z., Wang, G. A., Zhou, M., and Fan, W. 2018. “The Impact of Customer Reviews on Product

Innovation: Empirical Evidence in Mobile Apps," in *Analytics and Data Science*, Springer, pp. 95–110.

Ragaglia, D., and Roma, P. 2014. "Understanding the Drivers of the Daily App Rank : The Role of Revenue Models," in *Proceedings of the 26th Annual Conference of the Productions and Operations Management Society*, pp. 1–10.

Ramanand, J., Bhavsar, K., and Pedanekar, N. 2010. "Wishful Thinking: Finding Suggestions and 'Buy' Wishes from Product Reviews," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Association for Computational Linguistics, pp. 54–61.

Ramaswamy, V., and Prahalad, C. K. 2004. "Co-Creation Experiences: The Next Ractice in Value Creation," *Journal of Interactive Marketing* (18:3), pp. 5–14.

Ramos, J. 2003. "Using Tf-Idf to Determine Word Relevance in Document Queries," in *Proceedings of the First Instructional Conference on Machine Learning*.

Society, T. E. 2011. "Product Differentiation, Multiproduct Firms, and Estimating the Impact of Trade Liberalization on Productivity," *Econometrica* (79:5), pp. 1407–1451.

Di Sorbo, A., Panichella, S., Alexandru, C. V, Shimagaki, J., Visaggio, C. A., Canfora, G., and Gall, H. C. 2016. "What Would Users Change in My App? Summarizing App Reviews for Recommending Software Changes," in *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, ACM, pp. 499–510.

Stavrianou, A., and Brun, C. 2012. "Opinion and Suggestion Analysis for Expert Recommendations," in *Proceedings of the Workshop on Semantic Analysis in Social Media*, Association for Computational Linguistics, pp. 61–69.

Stieglitz, S., and Dang-Xuan, L. 2013a. "Emotions and Information Diffusion in Social Media—Sentiment of Microblogs and Sharing Behavior," *Journal of Management Information Systems* (29:4), pp. 217–248.

Stieglitz, S., and Dang-Xuan, L. 2013b. "Social Media and Political Communication: A Social Media Analytics Framework," *Social Network Analysis and Mining* (3:4), Springer, pp. 1277–1291.

Täckström, O., and McDonald, R. 2011. "Semi-Supervised Latent Variable Models for Sentence-Level Sentiment Analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*, Association for Computational Linguistics, pp. 569–574.

Wang, C., Jiang, F., and Yang, H. 2017. "A Hybrid Framework for Text Modeling with Convolutional RNN," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 2061–2069.

Wang, H., Lu, Y., and Zhai, C. 2010a. "Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, pp. 783–792.

Wang, H., Lu, Y., and Zhai, C. 2010b. "Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach," *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 783–792.

Winkler, M., Abrahams, A. S., Gruss, R., and Ehsani, J. P. 2016. "Toy Safety Surveillance from Online Reviews," *Decision Support Systems* (90), Elsevier, pp. 23–32.

Zhou, S., Qiao, Z., Du, Q., Wang, A. G., Fan, W., and Yan, X. 2018. "Measuring Customer Agility from Online Reviews Using Big Data Text Analytics," *Journal of Management Information Systems*, p. Forthcoming.

Zhu, J., Zou, H., Rosset, S., and Hastie, T. 2009. "Multi-Class Adaboost," *Statistics and Its Interface* (2:3), pp. 349–360.