Reinforced Product Metadata Selection for Helpfulness Assessment of Customer Reviews

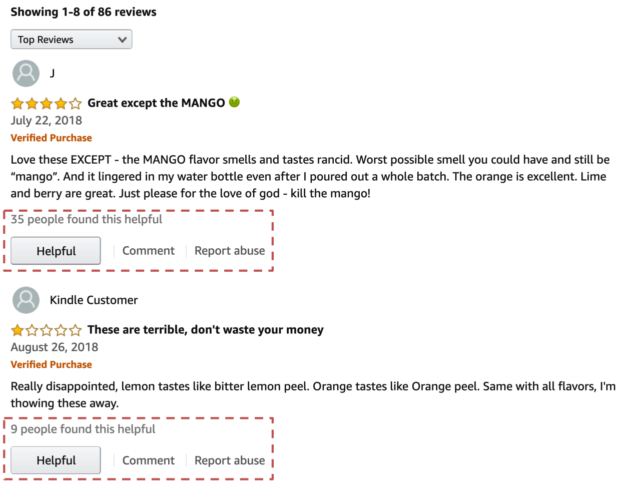
Miao Fan, Chao Feng, Mingming Sun, Ping Li
Cognitive Computing Lab
Baidu Research
No.10 Xibeiwang East Road, Beijing, 10085, China
10900 NE 8th St, Bellevue, WA 98004, USA
{fanmiao, v_fegchao, sunmingming01, liping11}@baidu.com

Abstract

To automatically assess the helpfulness of a customer review online, conventional approaches generally acquire various linguistic and neural embedding features solely from the textual content of the review itself as the evidence. We, however, find out that a helpful review is largely concerned with the metadata (such as the name, the brand, the category, etc.) of its target product. It leaves us with a challenge of how to choose the correct *key-value* product metadata to help appraise the helpfulness of *free-text* reviews more precisely. To address this problem, we propose a novel framework composed of two mutual-benefit modules. Given a product, a selector (agent) learns from both the *keys* in the product metadata and one of its reviews to take an action that selects the correct *value*, and a successive predictor (network) makes the *free-text* review attend to this *value* to obtain better neural representations for helpfulness assessment. The predictor is directly optimized by SGD with the loss of helpfulness prediction, and the selector could be updated via policy gradient rewarded with the performance of the predictor. We use two real-world datasets from Amazon.com and Yelp.com, respectively, to compare the performance of our framework with other mainstream methods under two application scenarios: helpfulness identification and regression of customer reviews. Extensive results demonstrate that our framework can achieve state-of-the-art performance with substantial improvements.

1 Introduction

The massive number of reviews left by many experienced consumers on online products are our priceless treasure in e-commerce. We believe that online customer reviews can provide more subjective and informative opinions from various perspectives on the products besides the objective descriptions given by their merchants. Hence, we prefer browsing the reviews online for the sake of finding our desirable products. However, it is quite time-consuming for potential buyers to sift through millions of online reviews with uneven qualities to make purchase decisions.



Showing 1-8 of 86 reviews

J

Great except the MANGO

July 22, 2018

Verified Purchase

Love these EXCEPT - the MANGO flavor smells and tastes rancid. Worst possible smell you could have and still be “mango”. And it lingered in my water bottle even after I poured out a whole batch. The orange is excellent. Lime and berry are great. Just please for the love of god - kill the mango!

35 people found this helpful

Kindle Customer

These are terrible, don't waste your money

August 26, 2018

Verified Purchase

Really disappointed, lemon tastes like bitter lemon peel. Orange tastes like Orange peel. Same with all flavors, I'm thowing these away.

9 people found this helpful

Figure 1: A screenshot of the top review on a product sold in Amazon.com: *SodaStream Fruit Drops Variety Pack, 1.67 Pound*. Online users can vote for the helpfulness of each customer review by clicking the “Helpful” button underneath.

To discover and surface helpful reviews for customers, some well-known websites (e.g., Amazon.com) have launched a module illustrated by Figure 1 which asks for feedback on the helpfulness of online reviews. And a recent study[1] reported that this featured module can increase the revenue of Amazon with an estimated 27 billion U.S. dollars annually. Although this crowdsourcing module is quite useful to find a fraction of helpful reviews, statistics (Fan et al., 2018) indicate that roughly **60%** online reviews still do not receive any vote of helpfulness or unhelpful-

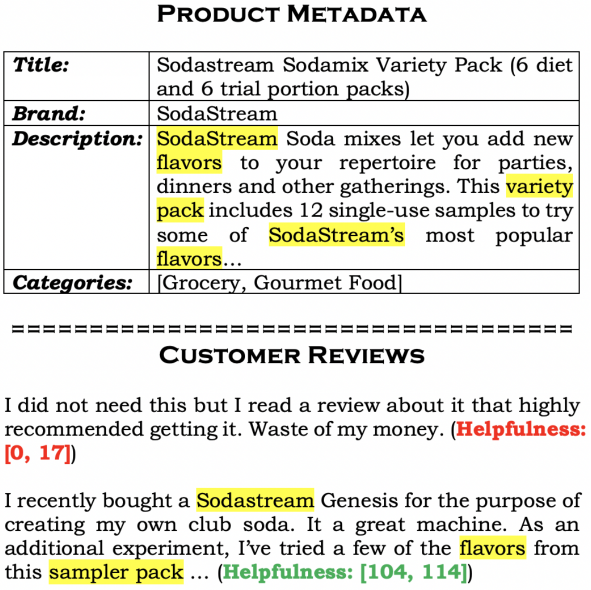
[1] articles.uie.com/magicbehindamazon/

ness. This phenomenon on unknown helpfulness is even more prominent in low-traffic items including those less popular and new arrival products.

We believe it is a promising study that establishes an automatic helpfulness assessment system for online reviews, which can be as useful as a product recommendation engine (Park et al., 2012) in e-commerce. Moreover, review helpfulness assessment (Diaz and Ng, 2018) and sentiment analysis (Liu and Zhang, 2012) are two different but related lines of work to study on the user-generated content (UGC). Sentiment analysis mainly focuses on identifying the opinion orientation of the reviewer himself/herself on the target product. A reviewer can express several emotional words in his/her comment. However, this comment may contain less information on the target product and may not be helpful to other potential consumers. Therefore, review helpfulness assessment concerns more about whether a comment is useful/helpful to other potential consumers.

So far as we know, a series of work on review helpfulness prediction has been proposed from two perspectives: 1) some work leveraged domain-specific knowledge to extract a wide range of hand-crafted features (including structural (Mudambi and Schuff, 2010; Xiong and Litman, 2014), lexical (Kim et al., 2006; Xiong and Litman, 2011), syntactic (Kim et al., 2006), emotional (Martin and Pu, 2014), semantic (Yang et al., 2015) and even argument features (Liu et al., 2017)) from the raw text of reviews as the evidence fed to off-the-shelf learning tools for helpfulness prediction; and 2) recent studies (Chen et al., 2018a,b; Fan et al., 2018) took advantages of deep neural nets by modifying the convolutional neural network (Kim, 2014) to acquire the low-dimensional representations of helpful reviews without the aid of feature engineering. Overall, these mainstream methods extract various linguistic and neural features solely from the raw text of a review as the evidence for helpfulness assessment.

We, on the other hand, consider that identifying the helpfulness of a review should be fully aware of the metadata (such as the name, the brand, the place of origin, the category, the description, etc.) of the target product besides the textual content of the review itself. To illustrate our idea, Figure 2 shows an example of two customer reviews in Amazon.com with diverse helpfulness scores on the same product online. We can easily figure out that the helpful review (104 of 114 people found it helpful) may concern the key of "description" in the metadata of the product, while the unhelpful review (0 of 17 people found it helpful) talks nothing about the product but expresses deep remorse. Without direct supervision from human beings, it is difficult for machines to infer the correct key/aspect in the product metadata that helpful reviews really concern.



PRODUCT METADATA

Title:	Sodastream Sodamix Variety Pack (6 diet and 6 trial portion packs)
Brand:	SodaStream
Description:	SodaStream Soda mixes let you add new flavors to your repertoire for parties, dinners and other gatherings. This variety pack includes 12 single-use samples to try some of SodaStream's most popular flavors...
Categories:	[Grocery, Gourmet Food]

====================================

CUSTOMER REVIEWS

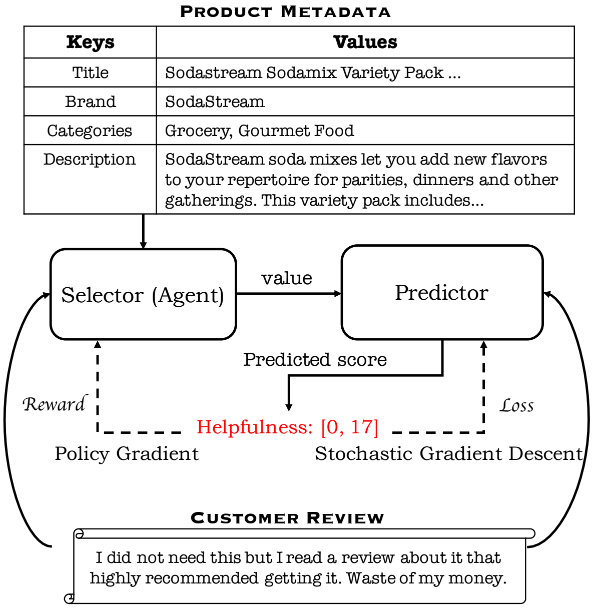
I did not need this but I read a review about it that highly recommended getting it. Waste of my money. (**Helpfulness: [0, 17]**)

I recently bought a Sodastream Genesis for the purpose of creating my own club soda. It a great machine. As an additional experiment, I've tried a few of the flavors from this sampler pack ... (**Helpfulness: [104, 114]**)

Figure 2: Two customer reviews with diverse helpfulness scores on the same product in Amazon.com.

It leaves us with a problem of how to teach machines learning to choose the correct *key-value* product metadata to help assess the helpfulness of *free-text* reviews more precisely. To address the issue, this paper introduces a novel framework composed of two mutual-benefit modules: a product metadata selector (agent) and a review helpfulness predictor (network). This work is proposed to be deployed in a real-world system. Therefore, we suggest to decouple the two modules (i.e., the selector and the predictor). Leveraging the attention mechanism is an alternative approach on an end-to-end framework. However, the decoupled modules are preferred in an industrial system.

Given a product, the selector **explores** the connection between the *keys* in the product metadata and one of its reviews to take an action that selects the correct *value*, and the successive predictor **exploits** this *value* attended by the *free-text* review to acquire better neural representations for helpfulness prediction. The predictor is directly



Keys	Values
Title	Sodastream Sodamix Variety Pack ...
Brand	SodaStream
Categories	Grocery, Gourmet Food
Description	SodaStream soda mixes let you add new flavors to your repertoire for parities, dinners and other gatherings. This variety pack includes...

I did not need this but I read a review about it that highly recommended getting it. Waste of my money.

Figure 3: Our reinforced framework for review helpfulness prediction: R^{2}HP.

optimized by the stochastic gradient descent algorithm (Ruder, 2016) with the loss of prediction, and the selector can be updated via the policy gradient method (Sutton et al., 2000) rewarded with the performance of the predictor.

We use two real-world datasets from Amazon.com and Yelp.com, to compare the performance with other mainstream approaches on two application tasks: helpfulness identification and regression of customer reviews. The experimental results reveal that our framework can reach state-of-the-art performance on both tasks with substantial improvements. In addition, our framework can help acquire the embeddings of the keys in product metadata, and visualization results illustrate that they can capture various aspects on customer reviews.

2 Framework

The intuition of our framework (named *reinforced review helpfulness prediction*, abbr. as R^{2}HP) is to predict the helpfulness of a customer review which is fully aware of the correct product metadata selected by a reinforced selector (agent). This work is proposed to be deployed in a real-world system which is responsible for recommending helpful reviews to millions of customers. Therefore, we suggest decoupling the two modules (selector and predictor). Leveraging the attention mechanism (Vaswani et al., 2017) is an alternative approach to establishing an end-to-end framework. However, the decoupled modules are preferred by the industry.

As shown by Figure 3, the selector (agent) learns from both the *keys* in the product metadata and one of its reviews to take an action that selects the correct *value*, and a successive predictor (network) makes the *free-text* review attend to this *value* to obtain better neural representations for helpfulness prediction. The predictor is directly optimized by SGD with the loss of prediction, and the selector could be updated via policy gradient rewarded with the performance of the predictor.

2.1 Reinforced Product Metadata Selection

Given a customer review and the key-value formed product metadata, our reinforced neural selector π takes them as input, to output a policy $\mathbf{p}$ which is the probability distribution over the keys. Suppose that the product metadata contains k keys each of which is represented by an l-dimensional vector. Then we can achieve the embeddings of the keys: $\mathbf{K} \in \mathsf{R}^{l \quad k}$.

Assume that there are n words/tokens in the customer review c. We align each word/token with the embedding dictionary acquired by the word embedding approaches such as Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014) or Elmo (Peters et al., 2018) to initialize the distributed representations of the customer review $\mathbf{C} \in \mathsf{R}^{l \quad n}$. To achieve the local contextual embeddings of the customer review c, we use a Bi-LSTM network (Schuster and Paliwal, 1997) which takes the word embeddings of the customer review $\mathbf{C}$ as input:

$$\mathbf{H}_c = \text{Bi-LSTM}(\mathbf{C}). \qquad (1)$$

$\mathbf{H}_c \in \mathsf{R}^{2l \quad n}$ stands for the contextual embeddings where each word can obtain two hidden units with the length of $2l$ encoding both the backward and the forward contextual information of the customer review locally.

We use $\mathbf{B} \in \mathsf{R}^{k \quad n}$ to obtain the bilinear relationship between the embeddings of the keys $\mathbf{K}$ and the local contextual embeddings of the customer review $\mathbf{H}_c$:

$$\mathbf{B} = \text{ReLU}(\mathbf{K}^{\mathrm{T}}\mathbf{W}\mathbf{H}_c), \qquad (2)$$

where $\mathbf{W} \in \mathsf{R}^{l \quad 2l}$ is the weight matrix for the Rectifier Linear Unit (ReLU). The i-th row of $\mathbf{B}$ contains the aspect/topic feature aligned by the local contextual embeddings of the customer review.

We apply the *reduce_max* strategy to each row of $\mathbf{B}$ to keep the most significant feature for each metadata key, and use the *softmax* function to gain the policy $\mathbf{p} \in \mathsf{R}^{k}$:

$$\mathbf{p} = \text{Softmax}(\text{Reduce\_max}(\mathbf{B}, axis = 1)). \quad (3)$$

Then our reinforced selector (agent) can select a metadata value $v \sim \pi(v|\mathbf{K}, \mathbf{C})$ in terms of $\mathbf{p}$.

2.2 Product-aware Review Helpfulness Prediction

In this part, we elaborate our neural predictor for review helpfulness prediction. It is devised based on the motivation that the helpfulness of an online review should be fully aware of the selected metadata of its target product besides the textual content of the customer review itself. The predictor is composed of two components: 1) the local contextual embeddings of a review and 2) the product-aware distributed representations of the review. We have explained how to obtain the local contextual embeddings of a review $\mathbf{H}_c \in \mathsf{R}^{2l \quad n}$ in the previous subsection. Here we will describe how to achieve the product-aware distributed representations of the review, denoted by $\overline{\mathbf{H}}_c$.

We use m to stand for the number of tokens/words in the selected metadata value v provided by our reinforced selector. Similarly, we can refine the word embeddings of the value $\mathbf{V}$ of the product metadata via another Bi-LSTM network:

$$\mathbf{H}_v = \text{Bi-LSTM}(\mathbf{V}), \quad (4)$$

and achieve the contextual embeddings of the selected product metadata $\mathbf{H}_v \in \mathsf{R}^{2l \quad m}$. To make the contextual embeddings of the customer review fully aware of the product metadata, we design a word-level matching mechanism as follows,

$$\mathbf{Q} = \text{ReLU}(\mathbf{W}^{0}\mathbf{H}_v + \mathbf{b}^{0} \otimes \mathbf{e})^{\text{T}}\mathbf{H}_c, \quad (5)$$

where $\mathbf{W}^{0} \in \mathsf{R}^{2l \quad 2l}$ is the weight matrix and $\mathbf{b}^{0} \in \mathsf{R}^{2l}$ is the bias vector. The outer product $\otimes$ copys the bias vector $\mathbf{b}^{0}$ for m times (i.e., $\mathbf{e} \in \mathsf{R}^{m}$) to generate a $2l \times m$ matrix. Then $\mathbf{Q} \in \mathsf{R}^{m \quad n}$ is the sparse matrix that holds the word-level matching information between the value v of the product metadata and the customer review c.

If we further apply the *softmax* function to each column of $\mathbf{Q}$, we will obtain $\mathbf{G} \in \mathsf{R}^{m \quad n}$, the i-th column of which represents the normalized attention weights over all the words in the metadata value v for the i-th word in the customer review c:

$$\mathbf{G} = \text{Softmax}(\mathbf{Q}). \quad (6)$$

Then we can use the attention matrix $\mathbf{G} \in \mathsf{R}^{m \quad n}$ and the contextual embeddings of the product metadata $\mathbf{H}_v \in \mathsf{R}^{2l \quad m}$ to re-form the product-aware review representation $\overline{\mathbf{H}}_c \in \mathsf{R}^{2l \quad n}$:

$$\overline{\mathbf{H}}_c = \mathbf{H}_v\mathbf{G}. \quad (7)$$

Driven by original motivation, we need to join the local contextual embeddings of the review ($\mathbf{H}_c$) and the product-aware distributed representations of the review ($\overline{\mathbf{H}}_c$) together for predicting its helpfulness with the feature matrix $\mathbf{H} \in \mathsf{R}^{2l \quad n}$:

$$\mathbf{H} = \mathbf{H}_c + \overline{\mathbf{H}}_c. \quad (8)$$

$\mathbf{H}$ can also benefit from the idea of ResNet (He et al., 2016) that efficiently acquires the residual between $\mathbf{H}_c$ and $\overline{\mathbf{H}}_c$, and provides a highway to update $\mathbf{H}_c$ if the residual is tiny.

Generally speaking, we define a loss function[2] $L(s_g|\mathbf{H}_c)$ which takes $\mathbf{H}_c$ as the feature to predict a helpfulness score s_p judged by the ground-truth score s_g. Given the value v is selected by π, our objective is to minimize the expectation:

$$J(\Theta) = \mathsf{E}_{v \quad \pi(vj\mathbf{K},\mathbf{C})}[L(s^g|v, \mathbf{H}^c)], \quad (9)$$

where Θ are parameters to be learned. The gradient of $J(\Theta)$ with respect to Θ is:

$$\begin{aligned} & r_{\Theta} J(\Theta) \quad (10) \\ & = r_{\Theta} \sum_{v} \pi(vj\mathbf{K}, \mathbf{C})[L(s^g jv, \mathbf{H}^c)] \\ & = \sum_{v} (L(s^g jv, \mathbf{H}^c)\, r_{\Theta}\pi(vj\mathbf{K}, \mathbf{C}) + \pi(vj\mathbf{K}, \mathbf{C})\, r_{\Theta} L(s^g jv, \mathbf{H}^c)) \\ & = \mathbb{E}_{v\sim\pi(v|\mathbf{K},\mathbf{C})}[L(s^g jv, \mathbf{H}^c)\, r_{\Theta} \log \pi(vj\mathbf{K}, \mathbf{C}) + \ r_{\Theta} L(s^g jv, \mathbf{H}^c)], \end{aligned}$$

where $\nabla \quad L(s^g|v, \mathbf{H}^c)$ refers to training the predictor by SGD and we use the REINFORCE algorithm (Williams, 1992) to update the selector with the gradient of $\log \pi(v|\mathbf{K}, \mathbf{C})$ and the reward $L(s^g|v, \mathbf{H}^c) \in (0.0, 1.0]$.

3 Experiments

3.1 Real-world Datasets

We look up two well-formatted JSON resources online which contain plenty of product metadata (including titles, brands, categories, and descriptions) and numerous customer reviews. One is the data collection[3] of Amazon.com crawled by

[2] We usually use the mean square error (MSE) as the loss function for helpfulness score regression and the cross-entropy as the loss function for helpfulness identification.

[3] The data collection is publicly available at http://jmcauley.ucsd.edu/data/amazon/links.html

Category (Amazon)	Training Set			Test Set		
	# (P.)	# (R. $\geq$ 1v.)	# (R. $\geq$ 0.75h.r.)	# (P.)	# (R. $\geq$ 1v.)	# (R. $\geq$ 0.75h.r.)
Clothing, Shoes & Jewelry	8,789	52,402	37,036	964	6,630	4,668
Electronics	16,456	215,512	136,798	1,858	21,706	13,845
Grocery & Gourmet Food	30,617	231,131	151,828	3,392	25,984	16,982
Health & Personal Care	19,817	216,839	135,501	2,198	23,517	14,565
Home & Kitchen	17,999	202,549	147,324	1,978	24,163	17,896
Movies & TV	7,518	316,235	143,790	847	35,880	15,973
Pet Supplies	18,189	167,775	123,287	2,031	21,170	15,650
Tools & Home Improvement	30,105	217,397	150,483	3,454	24,971	17,426
TOTAL	149,490	1,619,840	1,026,047	16,722	184,021	117,005

Table 1: The statistics of the Amazon dataset for helpfulness prediction of online reviews. # (P.): the number of products; # (R. $\geq$ 1v.): the number of the reviews, each receiving at least 1 vote regardless of helpfulness/unhelpfulness; # (R. $\geq$ 0.75h.r.): the number of the reviews, each regarded as helpful by at least 75% votes.

Category (Yelp)	Training Set			Test Set		
	# (P.)	# (R. $\geq$ 1v.)	# (R. $\geq$ 0.75h.r.)	# (P.)	# (R. $\geq$ 1v.)	# (R. $\geq$ 0.75h.r.)
Beauty & Spas	8,552	101,383	56,711	949	10,007	5,659
Health & Medical	10,079	90,045	57,967	1,132	10,434	6,758
Home Services	7,675	71,247	47,369	876	7,011	4,575
Restaurants	3,432	100,613	41,139	383	11,668	4,779
Shopping	11,706	104,555	49,833	1,295	12,892	5,969
TOTAL	41,444	467,843	253,019	4,635	52,012	27,740

Table 2: The statistics of the Yelp dataset for helpfulness prediction of online reviews. # (P.): the number of products; # (R. $\geq$ 1v.): the number of the reviews, each receiving at least 1 vote regardless of helpfulness/unhelpfulness; # (R. $\geq$ 0.75h.r.): the number of the reviews, each regarded as helpful by at least 75% votes.

(He and McAuley, 2016) up to July 2014. The other one is the dump file[4] directly provided by Yelp.com for academic purposes. We use the product ids (i.e., "asin" in Amazon and "business_id" in Yelp) as the foreign keys to align the metadata of products with customer reviews. 80% products with online reviews are randomly picked as the training set, leaving the rest as the test set. In this way, two real-world datasets, i.e., *Amazon* and *Yelp*, are built, and the statistics of them are shown by Table 1 and Table 2, respectively.

In this study, we regard the reviews which receive at least 1 vote for helpfulness/unhelpfulness, i.e., the column # (R.) $\geq$ 1v. in Table 1 and Table 2, as the experimental samples. In *Amazon*, the crowd-sourcing module for voting helpful reviews provides an "X of Y" helpfulness score, where "Y" stands for the total number of users who participate in voting, and "X" denotes the number of users who think the review is helpful. *Yelp* offers more options: *useful*: X, *cool*: Y, and *funny*: Z, to the users who are willing to give feedback. Regardless of the difference, we generally consider the reviews which receive at least 0.75 ratio of helpfulness/usefulness (helpfulness/usefulness score ≥ 0.75) as positive samples, leaving the others as the negative samples for classification.

3.2 Comparison Methods

We compare our framework (R^{2}HP) with a wide range of prior arts. Specifically, we re-implement the methods of learning from deep neural networks and with hand-crafted features. The up-to-date neural approaches involve the embedding-gated CNN (EG-CNN) (Chen et al., 2018a,b) and the multi-task neural learning (MTNL) architecture (Fan et al., 2018) for review helpfulness prediction. The hand-crafted features include the structural features (STR) (Mudambi and Schuff, 2010; Xiong and Litman, 2014), the lexical features (LEX) (Xiong and Litman, 2011), the emotional features (GALC) (Martin and Pu, 2014) and the semantic features (INQUIRER) (Yang et al., 2015). We also add two more experiments on integrating all the hand-crafted features via the Support Vector Machines (SVM) and the Random

[4] The dump file can be downloaded from `https://www.yelp.com/dataset`

Forest (R.F.) model for review helpfulness assessment.

3.3 Application Scenarios

Previous studies mostly reported their performance on either the application scenario of review helpfulness identification or regression. Therefore, we conduct extensive experiments comparing the performance of our framework with all the other approaches under both scenarios.

Identification of helpful reviews: As both the training and test sets are imbalanced, we adopt the Area under Receiver Operating Characteristic (AUROC) as the metric to evaluate the performance of all the approaches on helpful review identification. In line with Table 3 and Table 4, MTNL (Fan et al., 2018) achieves the best performance up-to-date on this classification task among the baseline approaches as it achieves the best performance on 12 of 14 categories in *Amazon* and *Yelp* datasets. R^{2}HP surpasses MTNL on both datasets and obtains state-of-the-art (micro-averaged) results of 67.5% AUROC (*Amazon*) and 75.1% AUROC (*Yelp*) with absolute improvements of **4.9%** AUROC and **4.7%** AUROC, respectively.

Regression of helpfulness score: In this task, all the approaches are required to predict the fraction of helpful votes that each review receives. We use the data in the column # (R. $\geq$ 1v.) in Table 1 and Table 2 as the training and test sets. The Squared Correlation Coefficient (R^2-score) is adopted as the metric to evaluate the performance of all the approaches on helpfulness score regression. Table 5 and Table 6 show that MTNL (Fan et al., 2018) achieves state-of-the-art performance on this regression task among the baseline approaches. Our framework outperforms MTNL on both datasets and obtains state-of-the-art (micro-averaged) results of 62.3% R^2-score (*Amazon*) and 74.0% R^2-score (*Yelp*) with absolute improvements of **5.4%** R^2-score and **5.8%** R^2-score, respectively.

4 Discussions

4.1 Ablation Study on Metadata Selector

In this part, we conduct the ablation study on different metadata selectors: i.e., random selector, heuristic selector, and our reinforced selector. The random selector requires no prior knowledge but randomly picks one pair of (key, value) from the product metadata with uniform distribution. The heuristic selector keeps choosing the pair of (key, value) in which the value contains the longest text[5]. Our reinforced selector learns from the reward given by the helpfulness predictor and makes a wise decision on the metadata selection. The values of product metadata selected by the three selectors are fed into the same predictor.

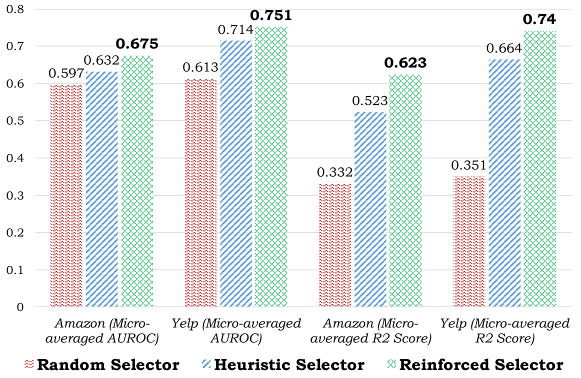


Figure 4: Performances of review helpfulness identification and regression supported by the random, the heuristic, and our reinforced selector on the *Amazon* and *Yelp* datasets.

Figure 4 shows the performance of both identification and regression of review helpfulness supported by the three different selectors, and the results demonstrate that our reinforced selector surpasses the other policies (the random selector and the heuristic selector) for metadata selection.

4.2 Case Study on Key Embeddings

Our framework also helps to acquire the distributed representations of the keys in product metadata. We believe these key embeddings can capture various aspects/topics on customer reviews and lead to the correct value for review helpfulness prediction. With the help of t-SNE (Maaten and Hinton, 2008), we can map the embeddings of the metadata keys from the *Amazon* and *Yelp* datasets into 2D vectors and illustrate them in Figure 5.

It shows that the key embeddings in *Amazon* locate at the bottom-right and the key embeddings in *Yelp* are generally at upper-left. The keys which express similar meanings within the same dataset are close to each other, such as the keys “city”, “state” and “address” in *Yelp*, and the keys “title”, “description” and “brand” in *Amazon*. Even across

[5] In most cases, the “name” or the “description” of a product is selected by the heuristic selector.

Category (Amazon)	**Area under Receiver Operating Characteristic (AUROC)**								
	STR	LEX	GALC	INQUIRER	FUSION (SVM)	FUSION (R.F.)	EG-CNN	MTNL	R^{2}HP
Clothing, Shoes & Jewelry	0.548	0.536	0.562	*0.601*	0.579	0.550	0.583	0.589	**0.623** (+0.022)
Electronics	0.583	0.549	0.588	*0.616*	0.577	0.580	0.611	0.613	**0.661** (+0.045)
Grocery & Gourmet Food	0.536	0.526	0.553	0.602	0.532	0.546	0.611	*0.626*	**0.657** (+0.031)
Health & Personal Care	0.558	0.523	0.559	0.610	0.591	0.562	0.613	*0.620*	**0.683** (+0.063)
Home & Kitchen	0.568	0.537	0.565	0.597	0.569	0.573	0.603	*0.610*	**0.646** (+0.036)
Movies & TV	0.603	0.558	0.621	0.634	0.603	0.607	0.648	*0.652*	**0.713** (+0.061)
Pet Supplies	0.560	0.542	0.585	0.603	0.548	0.558	0.580	*0.629*	**0.692** (+0.063)
Tools & Home Improvement	0.584	0.558	0.580	0.592	0.575	0.586	0.617	*0.624*	**0.672** (+0.048)
MACRO AVERAGE	0.568	0.541	0.577	0.607	0.572	0.570	0.608	*0.620*	**0.668** (+0.048)
MICRO AVERAGE (Primary)	0.571	0.543	0.580	0.609	0.573	0.574	0.614	*0.625*	**0.675** (+0.049)

Table 3: Results of performance (AUROC) of up-to-date methods on identifying helpful reviews evaluated by the test sets of Amazon. ($italic\ fonts$: the best performance among the baseline approaches; **bold fonts**: the state-of-the-art performance of all the approaches)

Category (Yelp)	**Area under Receiver Operating Characteristic (AUROC)**								
	STR	LEX	GALC	INQUIRER	FUSION (SVM)	FUSION (R.F.)	EG-CNN	MTNL	R^{2}HP
Beauty & Spas	0.573	0.610	0.627	0.674	0.651	0.662	0.693	*0.702*	**0.742** (+0.040)
Health & Medical	0.585	0.617	0.638	0.676	0.632	0.631	0.680	*0.697*	**0.725** (+0.028)
Home Services	0.582	0.596	0.625	0.684	0.635	0.638	0.663	*0.704*	**0.762** (+0.058)
Restaurants	0.595	0.616	0.652	0.682	0.669	0.654	0.681	*0.695*	**0.738** (+0.043)
Shopping	0.582	0.618	0.660	0.699	0.672	0.685	0.682	*0.719*	**0.784** (+0.065)
MACRO AVERAGE	0.583	0.611	0.640	0.683	0.652	0.654	0.680	*0.703*	**0.750** (+0.047)
MICRO AVERAGE (Primary)	0.584	0.613	0.643	0.684	0.654	0.656	0.681	*0.704*	**0.751** (+0.047)

Table 4: Results of performance (AUROC) on identifying helpful reviews evaluated by the test sets of Yelp.

Category (Amazon)	**Squared Correlation Coefficient (R^2-score)**								
	STR	LEX	GALC	INQUIRER	FUSION (SVM)	FUSION (R.F.)	EG-CNN	MTNL	R^{2}HP
Clothing, Shoes & Jewelry	0.265	0.337	0.398	0.588	0.490	0.506	0.558	*0.613*	**0.678** (+0.065)
Electronics	0.254	0.303	0.356	0.550	0.496	0.497	0.534	*0.593*	**0.636** (+0.043)
Grocery & Gourmet Food	0.242	0.319	0.408	0.450	0.426	0.425	0.469	*0.506*	**0.570** (+0.064)
Health & Personal Care	0.237	0.242	0.376	0.453	0.420	0.445	0.489	*0.519*	**0.583** (+0.064)
Home & Kitchen	0.236	0.298	0.330	0.464	0.439	0.461	0.502	*0.588*	**0.644** (+0.056)
Movies & TV	0.253	0.328	0.412	0.487	0.460	0.452	0.523	*0.594*	**0.636** (+0.042)
Pet Supplies	0.237	0.337	0.385	0.420	0.401	0.439	0.501	*0.573*	**0.657** (+0.084)
Tools & Home Improvement	0.234	0.301	0.387	0.437	0.447	0.502	0.532	*0.591*	**0.624** (+0.033)
MACRO AVERAGE	0.245	0.308	0.382	0.481	0.447	0.466	0.514	*0.572*	**0.629** (+0.057)
MICRO AVERAGE (Primary)	0.243	0.307	0.382	0.471	0.444	0.461	0.510	*0.569*	**0.623** (+0.054)

Table 5: Results of performance (R^2-score) on helpfulness voting regression evaluated by the test sets of Amazon.

Category (Yelp)	**Squared Correlation Coefficient (R^2-score)**								
	STR	LEX	GALC	INQUIRER	FUSION (SVM)	FUSION (R.F.)	EG-CNN	MTNL	R^{2}HP
Beauty & Spas	0.383	0.484	0.511	0.537	0.549	0.518	0.650	*0.662*	**0.727** (+0.065)
Health & Medical	0.353	0.454	0.533	0.557	0.598	0.579	0.672	*0.686*	**0.735** (+0.049)
Home Services	0.364	0.452	0.516	0.554	0.592	0.601	0.670	*0.695*	**0.741** (+0.046)
Restaurants	0.396	0.438	0.483	0.501	0.556	0.597	0.610	*0.623*	**0.682** (+0.059)
Shopping	0.406	0.447	0.537	0.623	0.634	0.709	0.737	*0.742*	**0.805** (+0.063)
MACRO AVERAGE	0.380	0.455	0.516	0.554	0.586	0.601	0.668	*0.682*	**0.738** (+0.056)
MICRO AVERAGE (Primary)	0.383	0.454	0.516	0.557	0.587	0.606	0.670	*0.682*	**0.740** (+0.058)

Table 6: Results of performance (R^2-score) on helpfulness voting regression evaluated by the test sets of Yelp. ($italic\ fonts$: the best performance among the baseline approaches; **bold fonts**: the state-of-the-art performance of all the approaches)

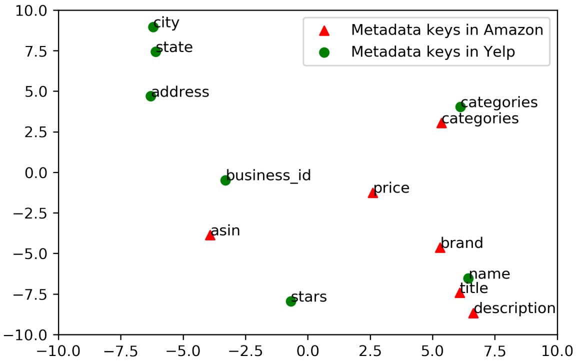


Figure 5: The 2D visualization of the embeddings of the metadata keys in *Amazon* and *Yelp*.

different datasets, the key embeddings draw near to each other if they are close in meaning, e.g., “title” (*Amazon*) and “name” (*Yelp*), or “asin” (*Amazon*) and “business_id” (*Yelp*).

5 Conclusion and Future Work

Driven by the intuition that the helpfulness of an online review should be fully aware of the metadata of its target product besides the textual content of the review itself, we take on the challenge of selecting the correct *key-value* product metadata to help predict the helpfulness of *free-text* reviews more precisely. To address the problem, we propose a novel framework in this paper, which is composed of two interdependent modules. Given a product, an agent (selector) learns from both the *keys* in the product metadata and one of its reviews to take an action that selects the correct *value*, and a successive network (predictor) makes the *free-text* review attend to this *value* to produce better neural representations for helpfulness prediction. We use two real-world datasets from Amazon and Yelp, respectively, to compare the performance of our framework with other mainstream methods on two tasks: helpfulness identification and regression of online reviews. Extensive results show that our framework can achieve state-of-the-art performance with substantial improvements. Further discussions demonstrate that it can not only provide better policies on selecting the correct *value* of product metadata but also acquire the embeddings of the *keys* in the product metadata.

We also believe the study on review helpfulness assessment could be as important as the topic of product recommendation, and several open problems are deserved to be explored in the future:

User-specific and explainable recommendation of helpful reviews: As different users may concern about various aspects of the products online, helpful review recommendation needs to be more user-specific and self-explainable.

Enhancing the prediction of helpful reviews with unlabeled data: As a small proportion of reviews could be heuristically regarded as helpful or unhelpful, it thus becomes a promising study to automatically predict the helpfulness of online reviews based on the small amount of labeled data and a vast amount of unlabeled data.

Cross-domain helpfulness prediction of online reviews (Chen et al., 2018b): Given that it costs a lot on manually annotating a sufficient number of helpful reviews in a new domain, we should explore effective approaches on transferring useful knowledge from limited labeled samples in another domain.

References

Cen Chen, Minghui Qiu, Yinfei Yang, Jun Zhou, Jun Huang, Xiaolong Li, and Forrest Sheng Bao. 2018a. Review helpfulness prediction with embedding-gated CNN. *CoRR*, abs/1808.09896.

Cen Chen, Yinfei Yang, Jun Zhou, Xiaolong Li, and Forrest Sheng Bao. 2018b. Cross-domain review helpfulness prediction based on convolutional neural networks with auxiliary domain discriminators. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL'18)*, pages 602–607. Association for Computational Linguistics.

Gerardo Ocampo Diaz and Vincent Ng. 2018. Modeling and prediction of online product review helpfulness: a survey. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL'18)*, pages 698–708.

Miao Fan, Yue Feng, Mingming Sun, Ping Li, Haifeng Wang, and Jianmin Wang. 2018. Multi-task neural learning architecture for end-to-end identification of helpful reviews. In *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM'18)*, pages 343–350.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, pages 770–778.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web (WWW'16)*, pages 507–517. International World Wide Web Conferences Steering Committee.

Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. 2006. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP'06)*, pages 423–430. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*, pages 1746–1751. Association for Computational Linguistics.

Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. In *Mining text data*, pages 415–463. Springer.

Haijing Liu, Yang Gao, Pin Lv, Mengxue Li, Shiqiang Geng, Minglan Li, and Hao Wang. 2017. Using argument-based features to predict and analyse review helpfulness. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP'17)*, pages 1369–1374. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research (JMLR)*, 9(Nov):2579–2605.

Lionel Martin and Pearl Pu. 2014. Prediction of helpful reviews using emotions extraction. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI'14)*, pages 1551–1557. AAAI Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS'13)*, pages 3111–3119.

Susan M. Mudambi and David Schuff. 2010. What makes a helpful online review? a study of customer reviews on amazon.com. *MIS Quarterly*, 34(1):185–200.

Deuk Hee Park, Hyea Kyeong Kim, Il Young Choi, and Jae Kyeong Kim. 2012. A literature review and classification of recommender systems research. *Expert Systems with Applications*, 39(11):10059–10072.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL'18)*, pages 2227–2237. Association for Computational Linguistics.

Sebastian Ruder. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NIPS'00)*, pages 1057–1063.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS'17)*, pages 5998–6008.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Wenting Xiong and Diane Litman. 2011. Automatically predicting peer-review helpfulness. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL'11)*, pages 502–507. Association for Computational Linguistics.

Wenting Xiong and Diane J Litman. 2014. Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING'14)*, pages 1985–1995. Association for Computational Linguistics.

Yinfei Yang, Yaowei Yan, Minghui Qiu, and Forrest Bao. 2015. Semantic analysis and helpfulness prediction of text for online product reviews. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL'15)*, pages 38–44. Association for Computational Linguistics.