ABSTRACT
Helpful reviews are essential for e-commerce and review websites, as they can help customers make quick purchase decisions and merchants to increase profits. Due to a great number of online reviews with unknown helpfulness, it recently leads to promising research on building automatic mechanisms to assess review helpfulness. The mainstream methods generally extract various linguistic and embedding features solely from the text of a review as the evidence for helpfulness prediction. We, however, consider that the helpfulness of a review should be fully aware of the metadata (such as the title, the brand, the category, and the description) of its target product, besides the textual content of the review itself. Hence, in this paper we propose an end-to-end deep neural architecture directly fed by both the metadata of a product and the raw text of its reviews to acquire product-aware review representations for helpfulness prediction. The learned representations do not require tedious labor on feature engineering and are expected to be more informative as the target-aware evidence to assess the helpfulness of online reviews. We also construct two large-scale datasets which are a portion of the real-world web data in Amazon and Yelp, respectively, to train and test our approach. Experiments are conducted on two different tasks: helpfulness identification and regression of online reviews, and results demonstrate that our approach can achieve state-of-the-art performance with substantial improvements.

KEYWORDS
E-commerce; online reviews; helpfulness prediction; neural networks; benchmark datasets

1 INTRODUCTION
The e-commerce and review websites dramatically facilitate our daily lives as they provide a massive number of products and businesses available online and allow us to access the comments made by experienced consumers. In order to find a desirable product, we prefer to browsing the online reviews of a product in addition to its descriptions, as we believe that the online reviews can provide more subjective and informative opinions from various perspectives on the product besides the objective descriptions given by its merchant. With the explosive growth of the transaction volume of the e-commerce market in recent years, it is common for an online product commented and rated by thousands of purchasers as illustrated by Figure 1. As a result, it is quite time-consuming for potential consumers to sift through all the online reviews with uneven qualities to make purchase decisions.

Therefore, helpful reviews are essential for e-commerce services, as they are able to bridge the gap between customers and merchants in a win-win manner, where customers tend to make quick purchase decisions once catching sight of helpful reviews and merchants can increase profits by surfacing the helpful reviews. To discover the helpful reviews, some platforms such as Amazon and Yelp have launched a module (see the "Helpful" buttons beneath each customer review shown by Figure 1) which allows users to...
give feedbacks on the helpfulness of online reviews. This featured module proved by a recent study\(^1\) increases the revenue of Amazon with an estimated 27 billion U.S. dollars annually. Although the crowd-sourcing module could help find a fraction of helpful reviews, roughly 60% online reviews in Amazon.com and Yelp.com did NOT receive any vote of helpfulness or unhelpfulness. This phenomenon on unknown helpfulness is even more common in low-traffic items including those less popular and new arrival products.

It leads to a promising research direction on building an automatic helpfulness prediction system for online reviews, which we believe could be as useful as a product recommendation engine in e-commerce. So far as we know, a series of work on review helpfulness prediction has been proposed from two perspectives: 1) some work leverages domain-specific knowledge to extract a wide range of hand-crafted features (including structural, lexical, syntactic, emotional, semantic and argument features) from the text of reviews as the evidence for off-the-shelf learning tools; and 2) recent studies modify the convolutional neural network [9, 10] to acquire low-dimensional features from the raw text of online reviews. Generally speaking, these mainstream approaches extract various linguistic and embedding features solely from the text of a review as the evidence for helpfulness prediction.

We, however, suggest that the helpfulness of a review should be fully aware of the meta-data (e.g., title, brand, category, description) of the target product besides the textual content of the review itself. Take the online customer review (circled by the orange dashed box) shown by Figure 1 as an example. The effective features to indicate that it is a helpful review on the product: *Beats Solo3* are probably the phrase "sound quality" and word "headphone". But the same textual features are hardly considered to be helpful when appearing in a comment on a *Nikon Digital Camera*.

In this paper, we propose an end-to-end deep neural architecture to capture the intrinsic relationship between the meta-data of a product and its numerous comments that could be beneficial to discover the helpful reviews. Our model is directly fed by both the title of a product and the raw text of its reviews, and then acquire product-aware review representations from the supervision of helpfulness scores given by the crowd-sourcing module. The learned neural representations do not require tedious labor on feature engineering and are expected to be more informative as the product-aware evidence to assess the helpfulness of online reviews. Given the drawbacks that prior systems have been evaluated by different datasets and not built on the successes of each other, we also construct two large-scale datasets, i.e. *Amazon-9* and *Yelp-5*, which are a portion of the real-world web data in Amazon.com and Yelp.com, respectively, for the successive assessment on the helpfulness prediction of online reviews. The mainstream approaches mentioned in this paper are re-implemented in addition to our model, for fair comparison. Extensive experiments are conducted on two different application scenarios: helpfulness identification and regression of online reviews using the two benchmark datasets. Experimental results demonstrate that our model can achieve state-of-the-art performance on the two tasks with significant absolute improvements (4.25% AUROC in the identification of helpful reviews and 4.40% \(R^2\)-score in the regression of helpfulness voting).

\(^1\)https://articles.uie.com/magicbehindamazon/

## 2 RELATED WORK

An up-to-date and comprehensive survey on various approaches on helpfulness prediction of online reviews was recently conducted by Ocampo Diaz and Ng [17]. According to their survey, these mainstream methods generally extract various linguistic and embedding features solely from the text of an online review as the evidence for helpfulness prediction. We go further and categorize those approaches into two classes in terms of the way of acquiring supportive features for helpfulness prediction, i.e., learning with hand-crafted features (see Section 2.1) and from deep neural networks (see Section 2.2), respectively.

### 2.1 Learning with Hand-crafted Features

So far as we know, a series of conventional approaches on review helpfulness prediction leverage domain-specific knowledge to extract a wide range of hand-crafted features from the text of customer reviews as the evidence fed into off-the-shelf classifiers or regressors such as SVM [1, 5] or Random Forest [12, 23]. According to [13] and [26], these hand-crafted features involve:

- **Structural features** (STR) [16, 25]: The structural evidence refers to the number of tokens, the number of sentences, the average length of sentences, and even the star rating of an online review. These important features indicate the attitude of the buyers when they write down their comments.
- **Lexical features** (LEX) [8, 24]: Inspired by the idea of text classification, the bag-of-words (BOW) features are essential to helpfulness prediction of online reviews. Explicitly, we usually remove the stop words and non-frequent words, extract unigrams (UGR) and bigrams (BGR) and weight these terms by the measurement of \(tf-idf\) as the lexical features.
- **Syntactic features** [8]: We can also obtain the part-of-speech (POS) tag of each token in a review. The syntactic features are composed of the percentages of tokens that are nouns, verbs, adjectives, and adverbs, respectively.
- **Emotional features**: Martin and Pu [14] used the Geneva Affect Label Coder (GALC) dictionary [20] to define emotional states of a review. The emotional features include the number of occurrences of each emotional state plus one additional dimension for the number of non-emotional words.
- **Semantic features**: Yang et al. [26] leveraged the General Inquirer (INQUIRER) dictionary [22] to map each word in a review into a semantic tag. This is similar to the way of obtaining emotional features. The semantic features are formulated by a vector where each entry records the number of the occurrences of each semantic tag.
- **Argument features**: Liu et al. [13] explored more intricate linguistic features such as evidence-conclusion discourse relations, also known as arguments, to study the helpfulness of an online review. To be exact, they adopted different granularities of argument features, e.g., the number of arguments, the number of words in arguments, etc.

### 2.2 Learning from Deep Neural Nets

The emergence of Deep Learning [11] brings in a good insight that we do not have to manually design heuristic rules to extract domain-specific features for learning tasks. To avoid the tedious
content of the review itself. As shown by Figure 2, it is composed of two components: 1) the local contextual embeddings of a review and 2) the product-aware distributed representations of the review. We will then explain how to model the two components.

Suppose that we have a product title \( P \) and one of its online reviews \( R \). We use \( m \) and \( n \) to denote the number of tokens/words in the product title \( P \) and the review \( R \), respectively. Firstly, we align each token with the embedding dictionary acquired by the word embedding approaches such as Word2Vec [15], Glove [18] or Elmo [19] to initialize the distributed representations of the product title \( P \in \mathbb{R}^{1 \times m} \) and the review \( R \in \mathbb{R}^{1 \times n} \).

To achieve the local contextual embeddings of the review \( R \), we use a Bi-LSTM network [21] which takes the word embeddings of the review \( R \) as input:

\[
H^R = \text{Bi-LSTM}(R). \tag{1}
\]

\( H^R \in \mathbb{R}^{2l \times n} \) 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 review locally.

Similarly, we can re-fine the word embeddings of the product title \( P \) via another Bi-LSTM network:

\[
H^P = \text{Bi-LSTM}(P), \tag{2}
\]

and achieve the contextual embeddings of the product title \( H^P \in \mathbb{R}^{2l \times m} \). To make the contextual embeddings of the review fully aware of the product title, we devise a word-level matching mechanism as follows,

\[
Q = \text{ReLU}(W^P H^P + b_R^P \otimes e_p^T H^R), \tag{3}
\]

where \( W^P \in \mathbb{R}^{2l \times 2l} \) is the weight matrix and \( b_R^P \in \mathbb{R}^{2l} \) is the bias vector for the Rectifier Linear Unit (ReLU). The outer product \( \otimes \) copies the bias vector \( b_R^P \) \( m \) times to generate a \( 2l \times m \) matrix. Then \( Q \in \mathbb{R}^{m \times n} \) is the sparse matrix that holds the word-level matching information between the product title \( P \) and the review \( R \). If we further apply the softmax function to each column of \( Q \), we will obtain \( G \in \mathbb{R}^{m \times n} \), the \( i \)-th column of which represents the normalized attention weights over all the words in product title \( P \) for the \( i \)-th word in the review \( R \):

\[
G = \text{softmax}(Q). \tag{4}
\]

Then we can use the attention matrix \( G \in \mathbb{R}^{m \times n} \) and the contextual embeddings of the product \( H^P \in \mathbb{R}^{2l \times m} \) to re-form the product-aware review representation \( \overline{H}^P \in \mathbb{R}^{2l \times n} \):

\[
\overline{H}^P = H^P G. \tag{5}
\]

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

\[
H = H^R + \overline{H}^R. \tag{6}
\]

H can also benefit from the idea of ResNet [6] that efficiently acquires the residual between \( H^R \) and \( \overline{H}^R \), and provides a highway to update \( H^R \) if the residual is tiny.
We find two well-formatted JSON resources online which contain plenty of meta-data (including titles, brands, categories, and descriptions) of products and numerous customer reviews. One is the data collection of Amazon.com crawled by He and McAuley [7] up to July 2014. The other one is the dump file directly provided by Yelp.com for academic purposes.

We use the product ids ("asin" in Amazon and "business_id" in Yelp) as the foreign keys to align the meta-data of products and numerous customer reviews. We randomly picked as the training set, leaving the rest as the test set. In this way, two benchmark datasets, i.e. Amazon-9 and Yelp-5, are prepared, and the statistics of the two datasets are shown by Table 1 and Table 2, respectively.

In line with Table 1, Amazon-9 covers more than 3 million products spreading over nine different categories in Amazon.com. About 50 million online reviews are included, but less than 48% (roughly 24 million reviews) of them receive at least 1 vote regardless of helpfulness/unhelpfulness by crowd-sourcing. As for Yelp-5 shown by Table 2, it contains about 130 thousand businesses which fall into five categories in Yelp.com. The proportion of voted reviews in Yelp-5 is also lower, i.e., roughly 52% (about 2.5 million reviews).

In this work, we regard the reviews which receive at least 1 vote, i.e. the column named after # (R. ≥ 1v.) in Table 1 and Table 2, as the experimental samples. In Amazon.com, the crowd-sourcing module for voting helpful reviews provides an "X of Y" score of helpfulness 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.com offers more options: useful, X, cool: Y, and funny: Z, to the users who are willing to give feedbacks. Regardless of the difference, we generally regard the reviews which receive at least 0.75 ratio of helpfulness/usefulness, i.e. # (R. ≥ 0.75h.r.), as positive samples, leaving the others as the negative samples.

### 4.1 Benchmark Datasets

We find two well-formatted JSON resources online which contain plenty of meta-data (including titles, brands, categories, and descriptions) of products and numerous customer reviews. One is the data collection of Amazon.com crawled by He and McAuley [7] up to July 2014. The other one is the dump file directly provided by Yelp.com for academic purposes.

We use the product ids ("asin" in Amazon and "business_id" in Yelp) as the foreign keys to align the meta-data of products and numerous customer reviews. We randomly picked as the training set, leaving the rest as the test set. In this way, two benchmark datasets, i.e. Amazon-9 and Yelp-5, are prepared, and the statistics of the two datasets are shown by Table 1 and Table 2, respectively.

In line with Table 1, Amazon-9 covers more than 3 million products spreading over nine different categories in Amazon.com. About 50 million online reviews are included, but less than 48% (roughly 24 million reviews) of them receive at least 1 vote regardless of helpfulness/unhelpfulness by crowd-sourcing. As for Yelp-5 shown by Table 2, it contains about 130 thousand businesses which fall into five categories in Yelp.com. The proportion of voted reviews in Yelp-5 is also lower, i.e., roughly 52% (about 2.5 million reviews).

In this work, we regard the reviews which receive at least 1 vote, i.e. the column named after # (R. ≥ 1v.) in Table 1 and Table 2, as the experimental samples. In Amazon.com, the crowd-sourcing module for voting helpful reviews provides an "X of Y" score of helpfulness 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.com offers more options: useful, X, cool: Y, and funny: Z, to the users who are willing to give feedbacks. Regardless of the difference, we generally regard the reviews which receive at least 0.75 ratio of helpfulness/usefulness, i.e. # (R. ≥ 0.75h.r.), as positive samples, leaving the others as the negative samples.

### 4.2 Comparison Methods

We compare our model (PRH-Net) with a wide range of prior arts 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.com offers more options: useful, X, cool: Y, and funny: Z, to the users who are willing to give feedbacks. Regardless of the difference, we generally regard the reviews which receive at least 0.75 ratio of helpfulness/usefulness, i.e. # (R. ≥ 0.75h.r.), as positive samples, leaving the others as the negative samples.
Table 3: Comparison of the performance (AUROC) of mainstream approaches on identifying helpful reviews evaluated by the test sets of Amazon-9. (italic fonts*: the best performance among the baseline approaches; bold fonts: the state-of-the-art performance of all the approaches)

<table>
<thead>
<tr>
<th>Category (Amazon-9)</th>
<th>STR</th>
<th>LEX</th>
<th>GALC</th>
<th>INQUIRER</th>
<th>FUSION (SVM)</th>
<th>FUSION (R.F.)</th>
<th>EG-CNN</th>
<th>MTNL</th>
<th>PRH-Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>0.595</td>
<td>0.572</td>
<td>0.610</td>
<td>0.620</td>
<td>0.594</td>
<td>0.601</td>
<td>0.625</td>
<td>0.629*</td>
<td>0.652 (+0.023)</td>
</tr>
<tr>
<td>Clothing, Shoes &amp; Jewelry</td>
<td>0.559</td>
<td>0.538</td>
<td>0.563</td>
<td>0.608*</td>
<td>0.587</td>
<td>0.557</td>
<td>0.590</td>
<td>0.592</td>
<td>0.614 (+0.006)</td>
</tr>
<tr>
<td>Electronics</td>
<td>0.590</td>
<td>0.555</td>
<td>0.593</td>
<td>0.627*</td>
<td>0.584</td>
<td>0.588</td>
<td>0.615</td>
<td>0.618</td>
<td>0.644 (+0.017)</td>
</tr>
<tr>
<td>Grocery &amp; Gourmet Food</td>
<td>0.540</td>
<td>0.526</td>
<td>0.566</td>
<td>0.618</td>
<td>0.537</td>
<td>0.556</td>
<td>0.613</td>
<td>0.638*</td>
<td>0.715 (+0.077)</td>
</tr>
<tr>
<td>Health &amp; Personal Care</td>
<td>0.560</td>
<td>0.533</td>
<td>0.569</td>
<td>0.617</td>
<td>0.599</td>
<td>0.565</td>
<td>0.617</td>
<td>0.624*</td>
<td>0.672 (+0.048)</td>
</tr>
<tr>
<td>Home &amp; Kitchen</td>
<td>0.572</td>
<td>0.545</td>
<td>0.576</td>
<td>0.609</td>
<td>0.579</td>
<td>0.573</td>
<td>0.605</td>
<td>0.611*</td>
<td>0.630 (+0.019)</td>
</tr>
<tr>
<td>Movies &amp; TV</td>
<td>0.613</td>
<td>0.562</td>
<td>0.624</td>
<td>0.637</td>
<td>0.605</td>
<td>0.617</td>
<td>0.648</td>
<td>0.652*</td>
<td>0.675 (+0.023)</td>
</tr>
<tr>
<td>Pet Supplies</td>
<td>0.560</td>
<td>0.542</td>
<td>0.585</td>
<td>0.603</td>
<td>0.548</td>
<td>0.558</td>
<td>0.580</td>
<td>0.619*</td>
<td>0.679 (+0.060)</td>
</tr>
<tr>
<td>Tools &amp; Home Improvement</td>
<td>0.584</td>
<td>0.548</td>
<td>0.580</td>
<td>0.592</td>
<td>0.565</td>
<td>0.586</td>
<td>0.607</td>
<td>0.621*</td>
<td>0.644 (+0.023)</td>
</tr>
<tr>
<td>MACRO AVERAGE</td>
<td>0.575</td>
<td>0.547</td>
<td>0.585</td>
<td>0.615</td>
<td>0.578</td>
<td>0.578</td>
<td>0.611</td>
<td>0.623*</td>
<td>0.658 (+0.035)</td>
</tr>
<tr>
<td>MICRO AVERAGE (Primary)</td>
<td>0.587</td>
<td>0.559</td>
<td>0.598</td>
<td>0.620</td>
<td>0.589</td>
<td>0.591</td>
<td>0.620</td>
<td>0.625*</td>
<td>0.651 (+0.026)</td>
</tr>
</tbody>
</table>

Table 4: Comparison of the performance (AUROC) of mainstream approaches on identifying helpful reviews evaluated by the test sets of Yelp-5. (italic fonts*: the best performance among the baseline approaches; bold fonts: the state-of-the-art performance of all the approaches)

<table>
<thead>
<tr>
<th>Category (Yelp-5)</th>
<th>STR</th>
<th>LEX</th>
<th>GALC</th>
<th>INQUIRER</th>
<th>FUSION (SVM)</th>
<th>FUSION (R.F.)</th>
<th>EG-CNN</th>
<th>MTNL</th>
<th>PRH-Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beauty &amp; Spas</td>
<td>0.512</td>
<td>0.500</td>
<td>0.527</td>
<td>0.570</td>
<td>0.521</td>
<td>0.541</td>
<td>0.571</td>
<td>0.587*</td>
<td>0.642 (+0.061)</td>
</tr>
<tr>
<td>Health &amp; Medical</td>
<td>0.525</td>
<td>0.517</td>
<td>0.538</td>
<td>0.576</td>
<td>0.539</td>
<td>0.531</td>
<td>0.580</td>
<td>0.596*</td>
<td>0.665 (+0.069)</td>
</tr>
<tr>
<td>Home Services</td>
<td>0.528</td>
<td>0.528</td>
<td>0.562</td>
<td>0.584</td>
<td>0.535</td>
<td>0.538</td>
<td>0.563</td>
<td>0.603*</td>
<td>0.732 (+0.129)</td>
</tr>
<tr>
<td>Restaurants</td>
<td>0.559</td>
<td>0.516</td>
<td>0.552</td>
<td>0.582</td>
<td>0.569</td>
<td>0.554</td>
<td>0.581</td>
<td>0.605*</td>
<td>0.658 (+0.053)</td>
</tr>
<tr>
<td>Shopping</td>
<td>0.528</td>
<td>0.518</td>
<td>0.560</td>
<td>0.609</td>
<td>0.542</td>
<td>0.555</td>
<td>0.572</td>
<td>0.619*</td>
<td>0.674 (+0.055)</td>
</tr>
<tr>
<td>MACRO AVERAGE</td>
<td>0.530</td>
<td>0.516</td>
<td>0.548</td>
<td>0.584</td>
<td>0.541</td>
<td>0.544</td>
<td>0.573</td>
<td>0.607*</td>
<td>0.674 (+0.073)</td>
</tr>
<tr>
<td>MICRO AVERAGE (Primary)</td>
<td>0.548</td>
<td>0.516</td>
<td>0.551</td>
<td>0.584</td>
<td>0.559</td>
<td>0.551</td>
<td>0.578</td>
<td>0.604*</td>
<td>0.663 (+0.059)</td>
</tr>
</tbody>
</table>

features include the structural features (STR) [16, 25], the lexical features (LEX) [8, 24], the emotional features (GALC) [14] and the semantic features (INQUIRER) [26]. We also add two more experiments on integrating all the hand-crafted features via the Support Vector Machines (SVM) and the Random Forest (R.F.) model for review helpfulness prediction.

4.3 Application Scenarios

Most previous studies just reported their performance on either the task of review helpfulness identification or regression. In this part, we assess the performance of PRH-Net by comparing our model with all the other approaches on both tasks.

4.3.1 Identification of Helpful Reviews. We use the data shown by the column named #(R. ≥ 1v.) in Table 1 and Table 2 to conduct this task. Within the data for binary classification, the reviews belonging to the column #(R. ≥ 0.75h.r.) are regarded as positive samples. 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. As shown by Table 3 and Table 4, MTNL [4] achieves the up-to-date performance on this classification task among the baseline approaches as it shows the best performance on 12 of 14 categories in Amazon-9 and Yelp-5 datasets. Our model (PRH-Net) surpasses MTNL on both datasets and obtains state-of-the-art (micro-averaged) results of 65.1% AUROC (Amazon-9) and 66.3% AUROC (Yelp-5) with absolute improvements of 2.6% AUROC and 5.9% AUROC, respectively.

4.3.2 Regression of Helpfulness Voting. In this task, all the approaches are required to predict the fraction of helpful votes that each review receives. We still use the data in the column named #R. ≥ 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 [4] achieves the up-to-date performance on this regression task among the baselines. Our model (PRH-Net) outperforms MTNL on both datasets and obtains state-of-the-art (micro-averaged) results of 55.2% $R^2$-score (Amazon-9) and 58.2% $R^2$-score (Yelp-5) with absolute improvements of 3.5% $R^2$-score and 5.3% $R^2$-score, respectively.
Table 5: Comparison of the performance ($R^2$-score) of mainstream approaches on helpfulness voting regression evaluated by the test sets of Amazon-9. (italic fonts*: the best performance among the baseline approaches; bold fonts: the state-of-the-art performance of all the approaches)

<table>
<thead>
<tr>
<th>Category (Amazon-9)</th>
<th>STR</th>
<th>LEX</th>
<th>GALC</th>
<th>INQUIRER</th>
<th>FUSION (SVM)</th>
<th>FUSION (R.F.)</th>
<th>EG-CNN</th>
<th>MTNL</th>
<th>PRH-Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>0.250</td>
<td>0.234</td>
<td>0.312</td>
<td>0.458</td>
<td>0.401</td>
<td>0.441</td>
<td>0.506</td>
<td>0.549*</td>
<td>0.586 (+0.037)</td>
</tr>
<tr>
<td>Clothing, Shoes &amp; Jewelry</td>
<td>0.245</td>
<td>0.307</td>
<td>0.368</td>
<td>0.428</td>
<td>0.360</td>
<td>0.376</td>
<td>0.491</td>
<td>0.510*</td>
<td>0.567 (+0.057)</td>
</tr>
<tr>
<td>Electronics</td>
<td>0.223</td>
<td>0.291</td>
<td>0.333</td>
<td>0.450</td>
<td>0.416</td>
<td>0.357</td>
<td>0.394</td>
<td>0.489*</td>
<td>0.516 (+0.027)</td>
</tr>
<tr>
<td>Grocery &amp; Gourmet Food</td>
<td>0.242</td>
<td>0.319</td>
<td>0.419</td>
<td>0.450</td>
<td>0.426</td>
<td>0.425</td>
<td>0.469</td>
<td>0.506*</td>
<td>0.569 (+0.063)</td>
</tr>
<tr>
<td>Health &amp; Personal Care</td>
<td>0.233</td>
<td>0.332</td>
<td>0.376</td>
<td>0.433</td>
<td>0.420</td>
<td>0.399</td>
<td>0.506</td>
<td>0.509*</td>
<td>0.537 (+0.028)</td>
</tr>
<tr>
<td>Home &amp; Kitchen</td>
<td>0.236</td>
<td>0.298</td>
<td>0.330</td>
<td>0.464</td>
<td>0.339</td>
<td>0.361</td>
<td>0.402</td>
<td>0.498*</td>
<td>0.513 (+0.015)</td>
</tr>
<tr>
<td>Movies &amp; TV</td>
<td>0.253</td>
<td>0.228</td>
<td>0.312</td>
<td>0.387</td>
<td>0.360</td>
<td>0.352</td>
<td>0.393</td>
<td>0.453*</td>
<td>0.495 (+0.042)</td>
</tr>
<tr>
<td>Pet Supplies</td>
<td>0.237</td>
<td>0.237</td>
<td>0.285</td>
<td>0.420</td>
<td>0.301</td>
<td>0.339</td>
<td>0.400</td>
<td>0.473*</td>
<td>0.523 (+0.050)</td>
</tr>
<tr>
<td>Tools &amp; Home Improvement</td>
<td>0.234</td>
<td>0.201</td>
<td>0.287</td>
<td>0.437</td>
<td>0.247</td>
<td>0.302</td>
<td>0.398</td>
<td>0.481*</td>
<td>0.503 (+0.022)</td>
</tr>
<tr>
<td><strong>MACRO AVERAGE</strong></td>
<td>0.239</td>
<td>0.272</td>
<td>0.336</td>
<td>0.436</td>
<td>0.363</td>
<td>0.372</td>
<td>0.440</td>
<td>0.496*</td>
<td>0.534 (+0.038)</td>
</tr>
<tr>
<td><strong>MICRO AVERAGE (Primary)</strong></td>
<td>0.243</td>
<td>0.258</td>
<td>0.325</td>
<td>0.445</td>
<td>0.385</td>
<td>0.399</td>
<td>0.463</td>
<td>0.517*</td>
<td>0.552 (+0.035)</td>
</tr>
</tbody>
</table>

Table 6: Comparison of the performance ($R^2$-score) of mainstream approaches on helpfulness voting regression evaluated by the test sets of Yelp-5. (italic fonts*: the best performance among the baseline approaches; bold fonts: the state-of-the-art performance of all the approaches)

<table>
<thead>
<tr>
<th>Category (Yelp-5)</th>
<th>STR</th>
<th>LEX</th>
<th>GALC</th>
<th>INQUIRER</th>
<th>FUSION (SVM)</th>
<th>FUSION (R.F.)</th>
<th>EG-CNN</th>
<th>MTNL</th>
<th>PRH-Net</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beauty &amp; Spas</td>
<td>0.283</td>
<td>0.384</td>
<td>0.511</td>
<td>0.537</td>
<td>0.349</td>
<td>0.418</td>
<td>0.550</td>
<td>0.552*</td>
<td>0.624 (+0.072)</td>
</tr>
<tr>
<td>Health &amp; Medical</td>
<td>0.253</td>
<td>0.454</td>
<td>0.433</td>
<td>0.557</td>
<td>0.478</td>
<td>0.459</td>
<td>0.572</td>
<td>0.573*</td>
<td>0.635 (+0.062)</td>
</tr>
<tr>
<td>Home Services</td>
<td>0.264</td>
<td>0.452</td>
<td>0.416</td>
<td>0.554</td>
<td>0.492</td>
<td>0.481</td>
<td>0.570</td>
<td>0.575*</td>
<td>0.645 (+0.070)</td>
</tr>
<tr>
<td>Restaurants</td>
<td>0.256</td>
<td>0.338</td>
<td>0.383</td>
<td>0.501</td>
<td>0.356</td>
<td>0.407</td>
<td>0.510</td>
<td>0.518*</td>
<td>0.564 (+0.046)</td>
</tr>
<tr>
<td>Shopping</td>
<td>0.306</td>
<td>0.347</td>
<td>0.417</td>
<td>0.523</td>
<td>0.334</td>
<td>0.400</td>
<td>0.537</td>
<td>0.540*</td>
<td>0.606 (+0.066)</td>
</tr>
<tr>
<td><strong>MACRO AVERAGE</strong></td>
<td>0.272</td>
<td>0.395</td>
<td>0.432</td>
<td>0.534</td>
<td>0.402</td>
<td>0.433</td>
<td>0.548</td>
<td>0.552*</td>
<td>0.615 (+0.063)</td>
</tr>
<tr>
<td><strong>MICRO AVERAGE (Primary)</strong></td>
<td>0.264</td>
<td>0.355</td>
<td>0.402</td>
<td>0.512</td>
<td>0.367</td>
<td>0.414</td>
<td>0.523</td>
<td>0.529*</td>
<td>0.582 (+0.053)</td>
</tr>
</tbody>
</table>

5 CONCLUSION AND FUTURE WORK

This paper engages in the emerging research on helpfulness prediction of online reviews. Our idea is driven by the motivations that 1) the helpfulness of an online review should be fully aware of the meta-data of its target product besides the textual content of the review itself; 2) the hand-crafted features requiring tedious labor are domain-specific and prone to high generation error; and 3) there is no widely-used benchmark dataset for constantly improving intelligent systems to precisely assess the helpfulness of online reviews.

To address the problems above, we contribute an end-to-end neural architecture which can automatically acquire product-aware review representations besides the textual embeddings of reviews as more informative evidence for review helpfulness prediction. We also construct two large-scale and real-world benchmark datasets, i.e. Amazon-9 and Yelp-5, for the sake of 1) fairly conducting the performance comparison of all the approaches on review helpfulness prediction, and 2) leaving the datasets available for successive studies. Specifically, we run extensive experiments on our newly constructed datasets under the application scenarios of helpfulness identification and regression. Experimental results demonstrate that our model surpasses all the mainstream approaches and achieves state-of-the-art performance with substantial improvements.

For future work, we might consider to study the following topics related to helpfulness prediction of online reviews:

- 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.
- Cross-domain helpfulness prediction of online reviews [3]: Given that it costs a lot on manually annotating plenty of helpful reviews in a new domain, we should explore effective approaches on transferring useful knowledge from limited labeled samples in another domain.
- Enhancing the prediction of helpful reviews with unlabeled data: As a small proportion of reviews could be heuristically regarded as helpful or unhelpful, it, therefore, becomes a promising study to automatically predict the helpfulness of online reviews based on the small amount of labeled data and a huge amount of unlabeled data.
REFERENCES


