A Content-Aware Trust Index for Online Review Spam Detection

Hao Xue, Fengjun Li

A Content-Aware Trust Index for Online Review Spam Detection

Hao Xue and Fengjun Li

The University of Kansas, Lawrence, KS, USA
{haoxue, fli}@ku.edu

Abstract. Online review helps reducing uncertainty in the pre-purchasing decision phase and thus becomes an important information source for consumers. With the increasing popularity of online review systems, a large volume of reviews of varying quality is generated. Meanwhile, individual and professional spamming activities have been observed in almost all online review platforms. Deceptive reviews with fake ratings or fake content are inserted into the system to influence people's perception from reading these reviews. The deceptive reviews and reviews of poor quality significantly affect the effectiveness of online review systems. In this work, we define novel aspect-specific indicators that measure the deviations of aspect-specific opinions of a review from the aggregated opinions. Then, we propose a three-layer trust framework that relies on aspect-specific indicators to ascertain veracity of reviews and compute trust scores of their reviewers. An iterative algorithm is developed for propagation of trust scores in the three-layer trust framework. The converged trust score of a reviewer is a credibility indictor that reflects the trustworthiness of the reviewer and the quality of his reviews, which becomes an effective trust index for online review spam detection.

Keywords: Trust, online review, opinion mining

1 Introduction

With the increasing popularity of online social-collaborative platforms, people get more connected and share various types of information to facilitate others' decision-making processes. A vast amount of user-generated content (UGC) has been made available online. For example, TripAdvisor.com, which specializes in travel-related services, has reached 315 million unique monthly visitors and over 200 million reviews. Yelp.com, which is known for restaurant reviews, has a total of 71 million reviews of businesses and a monthly average of 135 million unique visitors to the site. This plethora of data provides a unique opportunity for the formation of "aggregated opinions", from which people make reasonable judgments about the quality of a service or a product from an unknown provider.

However, the quality of UGC is problematic. For example, it has been observed that a non-negligible portion of online reviews is unfairly biased or misleading. To make things worse, deceptive reviews have been purposely planted

into online review systems by individual or professional spammers [12, 18, 39, 6]. For example, recent studies on Yelp show that about 16% restaurant reviews are considered suspicious and rejected by Yelp internally [20]. Opinions expressed in deceptive reviews deviate largely from the fact to mislead the consumers to make unwise decisions. This is known as *online review spamming*, which was first identified by Jindal et al. in [12].

Many deterrence-based and reputation-based approaches have been adopted to address the spamming problem in online review systems. For example, the "Verified Purchase" mechanism of Amazon.com labels reviews that are posted by consumers who actually have purchased the reviewed items. This label is often perceived by consumers as a positive indicator of the trustworthiness of the review. A more generally adopted approach is the "review of the review", which allows users to rate a review or vote for its "helpfulness". Readers then use the ratings and vote counts as a measure to assess the quality and the trustworthiness of the review. While these mechanisms provide additional information about how helpful or trustworthy a review is, their limitations are obvious. Similar to reviews, the "review of review" is a subjective judgment that can be easily gamed by purposeful spammers. Moreover, it suffers from inadequate user participation. Surveys show that only a small portion of users provides reviews online. Furthermore, among thousands of users who read a review, only a few provides feedback. In most online review systems, a large amount of reviews does not have any helpfulness or usefulness rating at all.

Detection-based mechanisms are considered more effective approaches to address the review spamming problem. Many learning-based schemes have been proposed to identify deceptive reviews and spamming reviewers from textual features [12, 25, 23], temporal features [6, 41], individual or group behavior patterns of spammers [39], and sentiment inconsistency [18, 24]. The rationale behind these approaches is two-fold. Along the first direction, the detection models rely on the deviation in rating behaviors. Since the objective of opinion spammers is to alter users' perception of the quality of a target, spammers often generate a large amount of reviews, seemingly from different users, with extreme ratings. In this way, spammers can significantly distort the mean rating. Such detection focuses on rating-based features that reflect the deviation from the aggregated rating (or the majority vote) [18, 1] and other rating behaviors (e.g., change of average rating over time, change of average ratings across groups of users, etc.). While these approaches have been used with success, they can be easily gamed by avoiding extreme rating behaviors. Along the second direction, detection schemes rely on text-based features, by identifying duplicated messages in multiple reviews [12, 23], or psycholinguistic deceptive characteristics [25]. These approaches involve training classifiers with manually labeled reviews, which is expensive and time-consuming. Combining review rating and textural features at the same time, some approaches detect spam reviews whose ratings are inconsistent with the opinions expressed in the review text [18, 24].

Inspired by these approaches, we propose a content-based trust index for review spam detection, which is based on a set of aspect-specific opinions extracted

from review content and iteratively computed in a three-layer trust propagation framework. First, we have observed two types of spams – (1) "Low-quality spams" that usually have short, poorly-written, sometimes irrelevant content. The low quality spams are generated at low cost so they often come in large quantity. These spams can be easily identified by existing detection mechanisms. (2) "High-quality spams" that are long, carefully composed, and well-written deceptive opinions. It is costly to generate such spams. However, it is also very difficult to detect them, especially using text features.

In this work, we target the second type of spamming reviews, whose content is carefully composed with bogus opinions. To engage the reader, these spams may include fake information cues or social motivational descriptions, which make them difficult to be detected by schemes using simple text-based features. To tackle this problem, we propose aspect-specific indicators that measure the deviation of an aspect-specific opinion of a review from the opinion aggregated across all reviews on that aspect of the target, based on majority vote. Although majority vote has some limitations in extreme cases, such as inertia against sudden change of quality, we assume in most cases the aspect-specific majority vote effectively reflects the fact. This is because in our approach, we first attempt to remove the low-quality reviews, regardless of benign or deceptive, as they do not contribute any meaningful *opinion*. With the remaining "meaningful" reviews, we can reasonably assume the benign reviews always outnumber the high-quality spams that are costly to generate. In extreme cases, where the number of high-quality spams is larger than or comparable to the number of truthful reviews, the review system is considered broken and no detection scheme could work. The rationale behind our approach is that if every review carries aspect-specific opinions, the majority vote on a common aspect should reflect the factual quality of the target, so that the agreement between the opinion of a review and the aggregated opinion reflects the quality of that review. Then, our scheme considers multiple aspect-specific indicators and integrates the deviations across all aspects.

To effectively integrate the aspect-specific indicators, we adopt a three-layer trust propagation framework, which was first described in [38]. It calculates trust scores for reviews, reviewers, and the aspect-specific opinions of the target (defined as "statement"). To do this, we first apply opinion mining techniques to extract aspect-specific opinions from the reviews, and then input them into the three-layer trust propagation model that iteratively computes the trust scores by propagating the scores between reviews, reviewers, and statements. As a result, the converged trust score of a reviewer reflects his overall deviation from the aggregated opinion across all aspects and all targets that he has reviewed. This is a strong indicator of trust to distinguish benign reviewers and high-quality spammers.

We summarize our contributions as: (i) We propose a novel aspect-specific opinion indicator as a content-based measure to quantify the quality and trustworthiness of review content. And (ii) We develop an iterative three-layer trust

propagation framework to compute trust scores for users, reviews, and statements as a measure of users' trustworthiness and stores' reputation.

2 Related Work

Opinion Mining. Opinion mining has been used to analyze the opinions, sentiments, and attitudes expressed in a textual content towards a target entity. It typically includes work from two related areas, opinion aspect extraction and sentiment analysis. Aspect extraction aims to extract product features from opinionated text. Many work considered it as a labeling task, thus rule based methods are used extensively [10, 19, 31–33]. To group the extracted aspect into categories, lexical tools like WordNet [22] is often used. Topic modeling-based approaches are also very popular[4, 17, 3, 13, 36], as they are able to extract and group aspects simultaneously. On the other hand, the goal of is to analyze the polarity orientation of the sentiment words towards a feature or a topic of the product. One of most common way is to use some sentiment lexicons directly, such as MPQA Subjectivity Lexicon [40] and SentiWordNet [2]. However, just like WordNet, these tools have their own limitations. Another common practice is to infer the polarity of target words using a small group of seed terms with known polarity [5, 37, 11]. In addition, supervised learning algorithms are often applied in previous work [28, 27, 34, 15]. In this work, we adopt the supervised learning method as our opinion mining technique. Note that opinion mining is not our focus here. The difference between our goal and typical opinion mining work is that we are not trying to improve the performance of extracting opinions. Instead, our purpose of applying opinion technique is to use the extracted aspects as deviation indicators for trustworthiness analysis.

Trust Propagation. Trust and trust propagation have been extensively studied in literature. The general idea of reinforcement based on graph link information has been proved effective. HITS [16] and PageRank [26] are successful examples in link-based ranking computation. [39] applied graph-based reinforcement model to compute trustworthiness scores for users, reviews, and stores. However, these approaches did not consider content information. [38] proposed a content-driven framework for computing trust of sources, evidence, and claims. The difference between this model and ours is that we extract more fine-grained information from content, while the model in [38] mainly used the similarities between content in general. In [38], the inter-evidence similarity plays an important role to make sure that similar evidences get similar scores. However, the consensuses of opinions used in our model already represent such similarity, so we did not add the inter-evidence similarity. Besides, we also redefined the computational rules in the context of our problem. In many work, trust is often generated and transmitted in a graph of trust, such in [14], [8], [21], [42]. Trust can also be inferred from rating deviations [35]. Different from previous approaches, our model derives trust from the consistence between an individual's opinion and the majority opinion.

3 Aspect-Specific Opinion Indicator

Existing content-based detection approaches take textual content of a review as input, which often use word-level features (e.g., n-grams) and known lexicons (e.g., WordNet[22] or psycholinguistic lexicon [18]) to learn classifiers that identify a review as spam or non-spam. To train the classifier, costly and time-consuming manually labeling of reviews is required. Due to subjectiveness of human judgment and personal preferences, there is no readily available ground truth of opinions. Therefore, a high-quality labeled dataset is difficult to obtain. Some existing work adopt crowdsourcing platforms such as Amazon Mechanical Turk to recruit human labeler, however, it is pointed out the quality of the labeled data is very poor. Different from these approaches, our opinion spam detection scheme focuses on deviation from the majority opinion. Although biased opinions always exist in UGC, we argue that a majority of users may be *biased but honest*, instead of maliciously deceptive. This is based on an overarching assumption regarding reviewer behaviors – that is the majority of reviews are posted by honest reviewers, as recognized by many existing work on opinion spam detection [18, 24, 1]. If this assumption does not hold, online peer review systems will be completely broken and useless. As a result, we propose to use the majority opinions as the "ground truth".

3.1 Aspect Extraction

Existing work on opinion mining studies opinions and sentiments expressed in review text at document, sentence or word/phrase levels. Typically, the overall sentiment or subjectiveness of a review (document-level) or a sentence of a review is classified and used as a text-based feature in spam detection. However, we consider these opinions are either too coarse or too fine-grained. For example, it is common that opposite opinions are expressed in an individual review – it may be positive about one aspect of the target entity but negative about another. This is difficult to capture using the document-level sentiment analysis. Therefore, the derived review-level majority opinion is inaccurate and problematic. Another direction of approaches proposes to use opinion features that associate opinions expressed in a review with specific aspects of the target entity [9, 24]. Intuitively, opinion features are nouns or noun phrases that typically are the subjects or objects of a review sentence. For example, in the below review, the underlined words/phrases can be extracted as opinion features.

"*This place is the bomb for milkshakes, ice cream sundaes, etc. Onion rings, fries, and all other "basics" are also fantastic. Tuna melt is great, so are the burgers. Classic old school diner ambiance. Service is friendly and fast. Definitely come here if you are in the area ...*"

Obviously, users may comment on a large number of very specific aspects about the target entity. The derived opinion features are thus too specific and too fine-grained to form a majority opinion on each feature, since other reviews

about the same target may not comment on these specific features. However, from the above example, we can see that opinion features such as "milkshakes", "fries", and "burgers" are all related to an abstract aspect "food". If we define a set of aspect categories, opinion features about a same or a similar high-level concept can be grouped together.

Consider a set of reviews ($\mathbf{R}$), which are written by a group of users ($\mathbf{U}$) about a set of entities ($\mathbf{E}$). Each review $r \in \mathbf{R}$ consists of a sequence of words $\{w_1, w_2, ..., w_{n_r}\}$. Then, we can define a set of m abstract aspects $a = \{a_1, a_2, ...a_m\}$, and sentiment polarity label $l = \{l_1, l_2, ...l_k\}$. As a result, for each review r, we can extract a set of aspect-sentiment tuple, denoted as $ao_i =< a_i, l_i >$, to represent the aspect-specific opinions of a user u towards a target entity e.

Typical sentiment polarity labels include "positive", "negative", "neutral", and "conflict" [7, 30]. Since "conflict" captures inconsistencies within a review but does not contribute to inter-review consistence, we do not include this label in our model. Abstract aspect categories are more difficult to define since they are domain-specific and thus need to be carefully tuned for a given domain. In this work, we use Yelp reviews as our dataset to study the credibility of users and their reviews. Therefore, we define a small set of aspect categories including four meaningful aspects *food, price, service*, and *ambience* for restaurant reviews. We consider the opinion extractions as a classification problem and adopt the support vector machine supervised learning model for opinion extraction. In this way, we classify each sentence to a specific aspect, and group sentences in a review about a same aspect as an aspect-specific "statement" (denoted as s_i). We use the SemEval dataset [30], which is a decent-sized set of labeled data for restaurant reviews, to train our classifier (see more details in Section 5). Our goal is to identify an adequate number of aspects that are commonly addressed by all reviews so that we can construct a credibility indicator from the aggregated opinions. In fact, too many over-specific aspects complicate the credibility computing model instead of improving it. Therefore, we combine all other aspect category labels in the SemEval dataset as a fifth category "miscellaneous". This is different from previous work that considered all aspects in the classification [7].

Next, we conduct the aspect-specific sentiment classification upon the classified aspect-specific statements to obtain aspect-specific sentiment polarities. For each category, the classification is conducted independently. For example, to determine the sentiment polarities of the "food" category, we conduct a sentiment classification upon all statements that have been classified into the category "food", and determine the aspect-sentiment tuples: "food-positive", "food-negative", and "food-neutral".

3.2 Opinion Vector and Quality Vector

To use the extracted opinions for further analysis, we define an opinion vector $\mathbf{o} = [o_1, ..., o_5]$ to capture aspect-specific opinions and their sentiment polarities. Each element of the opinion vector corresponds to an aspect of food, price, service, ambience, and miscellaneous, respectively. Sentiment polarities are represented by element values, where a positive sentiment is denoted by "+1", a

negative sentiment is denoted by "-1", and neutral is denoted by "0". Since a statement may not necessary to express an opinion about an aspect, we distinguish no opinion expressed from a neutral opinion by defining a corresponding opinion status vector **os**. For example, if a statement expresses three opinions, positive about food, neutral about price, and negative about service, its opinion vectors are $\mathbf{o} = [1, 0, -1, 0, 0]$ and $\mathbf{os} = [1, 1, 1, 0, 0]$.

With the opinion vectors, we can aggregate the opinions on multiple aspects from all reviewers of an entity to form four aspect-specific aggregated opinions. While aspect-specific opinions are subject judgements and thus can be biased, the aggregated aspect-specific sentiments are highly likely to reflect the true quality of the entity from a specific aspect. This is because individual biases are typically smaller aspect level than at document level, which is more affected by the weights subjectively assigned by individuals to multiple aspects. In this sense, aspect-level bias can be corrected by the majority view if the review amount is adequate. Furthermore, comparing with rating, aspect-specific opinions are more difficult to be tampered by opinion spammers, whose review text are likely to be pointless, wrong focused, or brief. Finally, the aggregated sentiments are robust to correct the inaccuracy introduced by opinion mining models. Opinion mining often suffers from precision problems, but our goal is to decide if the overall aspect-specific opinion is positive, neutral, or negative. Although each individual input incurs a small uncertainty, the chance to affect overall value is very small. Based on these considerations, we derive the aggregated aspect-specific opinion vectors as $\mathbf{o_{agg}} = [o_{agg_1}, ..., o_{agg_5}]$ and $\mathbf{os_{agg}} = [os_{agg_1}, ..., os_{agg_5}]$, where

$$o_{agg_i} = \begin{cases} 1, & avg_{i \in A_i}(o_i) \geq \theta_p \\ -1, & avg_{i \in A_i}(o_i) \leq \theta_n \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

In this way, the aggregated sentiment polarity of each aspect is mapped to the positive, neutral, and negative labels based on the averages. The aggregated aspect-specific opinion vector is considered a *quality vector*, which can be used to determine the credibility of a user statement. Intuitively, a statement is more credible and of higher quality, if it expresses a consistent opinion with the aggregated opinion about one aspect of the target entity, and thus the reviewer is considered more honest and trustworthy.

4 Content-based Trust Computation

We compute the aggregated aspect-specific opinion vector as a quality measure and use individual aspect-specific opinion vector as a credibility (or trust) measure. To integrate trust measures across multiple users and multiple entities, trust propagation models are commonly used [43, 38]. Therefore, in this work, we adopt a three-layer trust propagation model to compute iteratively the trust-related scores for users, reviews, and aspect-specific statements.

The three-layer propagation model was first introduced in [38] to compute trustworthiness of data sources of free-text claims online. Most of the previous

work is based on a bi-partite graph structure, which ignores the content and the context in which the content is expressed. The intermediate layer in the three-layer model can include the content context (i.e., the reviews) and capture the intertwined relationships between users, reviews, and opinions expressed in specific statement. Therefore, we define three types of nodes, *users*, *reviews*, and *statements*, and compute the trustworthiness scores from the obtained opinion vectors. Each user is connected to the reviews she posts. Each review is connected to the statements expressed in the review itself. The statement is defined as an opinion expressed on a target in the review system, e.g. restaurant1-food-positive. The structure is shown in Fig. 1. In the figure, u_i, r_i, and s_i represent a user node, a review node, and a statement node respectively. $h(u_i)$, $f(r_i)$, and $t(s_i)$ are defined as the score of a user, a review, and a statement respectively. The value $p(r_i, s_i)$ is a weight on the link from a review to a statement. These values will be described in rest of this subsection shortly.

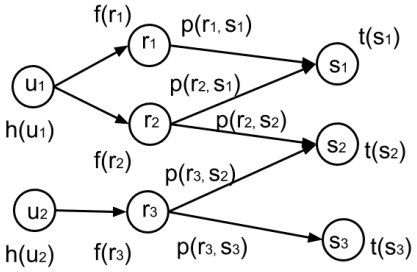


Fig. 1: Structure of the model

For each type of node, a score is defined, namely *honesty* for users, *faithfulness* for reviews, and *truthfulness* for statements. Different with the original model in [38], we defined a support weight on the links between reviews and statements, which measures the how supportive a review is for a statement. The support weight is defined as the sentiment consistency between the review and the statement it expressed. As mentioned above, there are three predefined sentiment polarities, positive, negative, and neutral. Here, if the sentiment polarity expressed in the review on a specific aspect category is the same as the sentiment polarity of the statement, then we say the review fully supports the statement. On the other hand, if the sentiment polarities between a review and a statement are totally opposite, i.e. positive and negative, or negative and positive, we say the review rejects the statement. For all other cases, we say the reviews partially support the statement. With these definitions, the values of support between review r_i and statement s_i are defined as:

$$p(r_i, s_i) = \begin{cases} 1, & r_i \text{ fully supports } s_i \\ 0, & r_i \text{ rejects } s_i \\ 0.5, & r_i \text{ partially supports } s_i \end{cases} \tag{2}$$

As mentioned before, each type of node has a type of score that measures the extent of trustworthiness. The honesty score for a user is defined in the following equation:

$$h^{n+1}(u_i) = \alpha h^n(u_i) + (1-\alpha)\frac{\sum_{r_j \in \mathcal{R}(u_i)} \sum_{s_k \in \mathcal{S}(r_j)} [p(r_j, s_k) \times t^n(s_k)]}{|\mathcal{R}(u_i)| \times |\mathcal{S}(r_j)|} \tag{3}$$

Here, different with the original model in [38], under our definition, the honesty score of a user consists of two parts. The first part is the honesty score from the last round. We added this part because a user's honesty does not entirely depend on the feedback from his/her statements. The second part is the feedback from the reviews and statements related to the user. $\mathcal{R}(u_i)$ is the collection of reviews that user u_i posts. $\mathcal{S}(r_j)$ is collection of statements review r_j expresses. $t^n(s_k)$ is the truthfulness score of statement s_k. $p(r_j, s_k)$ is a support value of review r_j to statement s_k. The second part is essentially a weighted average of truthfulness scores of all statements that reviews of user u_i expresses. The support value serves as a factor that controls the feedback. If a statement supports a statement with high truthfulness score, the contribution from this statement will be high. Otherwise, a user will be penalized for supporting a statement with low truthfulness score or rejecting a statement with high truthfulness score. The parameter α controls the ratio between the two parts. The faithfulness score for a review is defined as:

$$f^{n+1}(r_i) = \mu f^n(r_i) + (1-\mu)h^n(u(r_i)) \tag{4}$$

The faithfulness score of a review also comes from two parts, the faithfulness score from the previous round and the honesty score of the author. The parameter μ is also used to control the ratio between the two parts. Here $u(r_i)$ represents the user who writes review r_i. The truthfulness score for a statement is defined as:

$$t^{n+1}(s_i) = \frac{\sum_{r_j \in \mathcal{R}(s_i)} [f^n(r_j) \times h^n(u(r_j)) \times p(r_j, s_i)]}{|\mathcal{R}(s_i)|} \tag{5}$$

$\mathcal{R}(s_i)$ is the collection of reviews that express statement s_i. The truthfulness score of statement s_i is essentially a weighted average of honesty scores of the users whose reviews express this statement. The three types of scores are all in the range [0, 1]. From the formulas above, it is obvious that the three types of scores are defined in an intertwined relationship. The measurement of trust is propagated along the structural connections. For example, a user's honesty

score is dependent on the trustworthiness of the statements in his reviews, thus the trust is propagated from his statements to the user himself, and further propagates to his reviews and back to his statements. Each type of score gets feedbacks from the other two, which allows reinforcement based on the connections among the nodes. The scores of nodes are computed in an iterative computational framework, as shown in Algorithm 1. After the model converges, it outputs the final result.

Algorithm 1 Iterative framework to compute trust-related scores

Input:
Collections of users $\mathcal{U}$, reviews $\mathcal{R}$, and statements $\mathcal{S}$;
Initial sentiment polarities for all statements in $\mathcal{S}$;
Interpolation parameters α, μ;
Output:
honesty scores $h(u)$ for all users in $\mathcal{U}$, faithfulness scores $f(r)$ for all reviews in $\mathcal{R}$, and truthfulness scores $t(s)$ for all statements in $\mathcal{S}$;
repeat
Compute the honesty scores for all users using (3)
Compute the truthfulness scores for all statements using (5)
Compute the faithfulness scores for all reviews using (4)
Normalize each type of score so that the largest is 1
until converged

5 Experiments

5.1 Dataset

We used two datasets in the experiments. We first used the SemEval dataset [30], which contains 3,041 sentences from restaurant reviews, to train our classifier. In SemEval, each sentence is labeled with one or multiple aspect categories (i.e., food, service, price, ambience, and anecdotes/miscellaneous) and the corresponding sentiment polarities (i.e., positive, neutral, negative, and conflict). As discussed in Sect. 3, the "conflict" sentiment category is not considered in our model. We then split this dataset in 4:1 ratio with a training dataset and a testing dataset of 2,432 and 609 labeled sentences, respectively.

We tested our content-aware trust propagation scheme on a second dataset with restaurant reviews from Yelp.com, which is a subset of the dataset that we crawled from Yelp.com in 2013. The entire dataset contains 9,314,945 reviews about 125,815 restaurants in 12 U.S. cities from 1,246,453 reviewers between 2004 and 2013. In this experiment, we extracted a dataset for the city of Palo Alto, California. It contains 128,361 reviews about 1,144 restaurants from 45,180 reviewers. Although our dataset contains rich information about the reviewers, such as the total number of reviews, average ratings, social relationships, etc., we only used review content in this study.

5.2 Aspect Category and Sentiment Polarity Classifications

Aspect Category Classification. We use Support Vector Machine (SVM) in the Python machine learning library scikit-learn [29] as the classifier for opinion extraction. For feature extractions, we used the bag-of-words model and extracted the tf-idf weights as features. The classifiers for aspect categories and sentiment polarities are trained separately at sentence level.

Since SVM is a binary classifier, a trained SVM classifier can only classify whether a sentence contains a category or not. However, a single sentence may contain multiple aspect categories, which cannot be classified with a single SVM classifier. Therefore, we trained five separate binary one-vs-all SVM classifiers independently, one for each aspect category. For example, if a sentence contains opinions about "food", it is classified into the "food" category by the "food" classifier. If it contains opinions about both "food" and "price", the "food" classifier identifies the sentence as "food", and the "price" classifier also identities it as "price" at the same time. The results of aspect category classification are shown in Table 1.

Table 1: Classification performance of aspect category classifiers

Label	Precision	Recall	F1-score	Support	Accuracy
food	0.81	0.78	0.80	238	0.844
not_food	0.86	0.88	0.87	371	
avg / total	0.84	0.84	0.84	609	
price	0.91	0.62	0.73	65	0.952
not_price	0.96	0.99	0.97	544	
avg / total	0.95	0.95	0.95	609	
service	0.82	0.69	0.75	122	0.906
not_service	0.92	0.96	0.94	487	
avg / total	0.90	0.91	0.90	609	
ambience	0.83	0.52	0.64	84	0.920
not_ambience	0.93	0.98	0.95	525	
avg / total	0.91	0.92	0.91	609	
anecdotes/miscellaneous	0.77	0.70	0.73	243	0.796
not_anecdotes/miscellaneous	0.81	0.86	0.84	366	
avg / total	0.79	0.80	0.79	609	

For "avg/total" in the table, "avg" means the average of precision, recall, and f1-score, respectively, and "total" denotes the total support of each category. Among the five categories, the category anecdotes/miscellaneous has the worst performance (with the lowest precision of 0.77). This category contains the aspects that do not belong to any one of the other four categories. Since it is always easier to determine if a sentence does not belong to the anecdotes/miscellaneous category than it does, the precision, recall, and f1-score

of the "not_anecdotes/miscellaneous" category are higher than the "anecdotes / miscellaneous" category.

Interestingly, we find that the food category, which is a most popular aspect category for restaurant reviews, has the second-worst performance among the five categories. This may because many different terms and aspects representing the food category. Using tf-idf weights as the features, it is difficult to have a unified representation of the category. Therefore, it is relatively more difficult to train an effective classifier for the food category than for others such as price or service.

Sentiment Polarity Classification. After we obtain the classified results of aspect categories, we apply the sentiment classifier in each category to compute the category-based sentiment polarities. One review may contain multiple opinions about multiple categories. For example, after sentiment polarity classification, we can extract opinions as "food-positive", "price-neutral", and "service-negative" from a single review. We show the results of sentiment polarity classification in Table 2.

Table 2: Classification performance of category-based sentiment polarities

Label	Precision	Recall	F1-score	Support	Accuracy
food,negative	0.39	0.36	0.38	33	
food,neutral	0.50	0.04	0.07	25	
food,positive	0.80	0.92	0.85	169	0.740
avg / total	0.71	0.74	0.70	227	
price,negative	0.55	0.44	0.49	25	
price,neutral	0.00	0.00	0.00	2	
price,positive	0.67	0.81	0.73	36	0.635
avg / total	0.60	0.63	0.61	63	
service,negative	0.66	0.69	0.67	48	
service,neutral	0.00	0.00	0.00	7	
service,positive	0.73	0.79	0.76	61	0.698
avg / total	0.66	0.70	0.68	116	
ambience,negative	0.64	0.30	0.41	23	
ambience,neutral	0.00	0.00	0.00	5	
ambience,positive	0.68	0.92	0.78	49	0.675
avg / total	0.62	0.68	0.62	77	
anecdotes/miscellaneous,negative	0.11	0.10	0.10	31	
anecdotes/miscellaneous,neutral	0.60	0.49	0.54	96	
anecdotes/miscellaneous,positive	0.60	0.73	0.66	107	0.547
avg / total	0.54	0.55	0.54	234	

From the two tables, we can see that the performance of aspect category classification is much better than category-based sentiment polarity classification. This is because using bag-of-words model, it is easier to find representative

features for categories than for sentiment polarities. Sometimes, the sentiment polarities are implicit and context-dependent. In addition, the category-based sentiment analysis takes the classification results as aspect categories. Any inaccuracy from previous classification results affects the overall performance. It is worth noting that the classification performance of sentiment polarity in our scheme is comparable to the baseline (e.g., some submissions in the SemEval 14 contest [30]).

The classifications of aspect categories and category-based sentiment polarities are used as input to the trust propagation model. SVM does not yield the best results, but the current classification results do provide a good set of inputs to the trust propagation computation. Other supervise- and unsupervised-classification models may yield higher precision and thus improve the performance of our model.

5.3 Trustworthiness Scores Computation

We construct the proposed three-layer trust propagation model using the structural relationships among reviewers, reviews, and statements, where the statements are aspect-specific opinions about the restaurants. There are in total three different sentiment polarities, but for each restaurant there exists at most one statement for a specific aspect category. Since the score of statement nodes depends on the feedback from the other two types of nodes as well as the support value on the link, the sentiment of statement can be set to any arbitrary polarity.

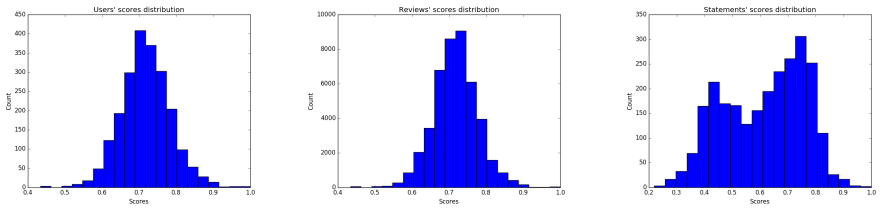
We conduct two sets of experiments which initialize the statement sentiments with two different settings. In the first set of experiments, we initialize the statement sentiments based on the aggregated opinions, and in the second set of experiments, we set all the statement sentiments as positive. In the experiments, we only use four aspect categories, i.e., *food*, *price*, *service*, and *ambience*. The category of miscellaneous is ignored since it is not an informative category related to trustworthiness analysis. Finally, in the experiments, we set the values of α and μ to 0.5.

Table 3: Average truthfulness scores of the statements of different categories under different initialization settings

Category	Initialized with majority opinions	Initialized to be all positive
Food	0.744	0.620
Price	0.554	0.196
Service	0.471	0.276
Ambience	0.676	0.102

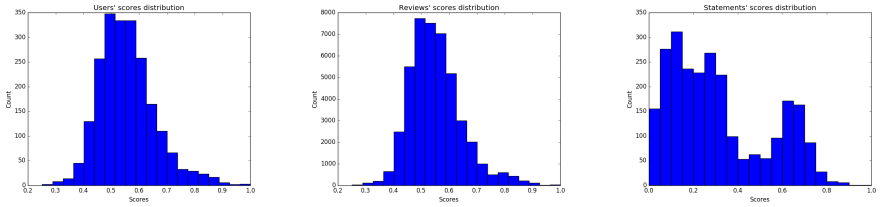
Results The average truthfulness scores of four aspect categories under two statement sentiment initialization settings are shown in Table 3. In both settings, food-related statements receive the highest scores.

Compared with scores using the first setting, the results from the second setting all decreased. Truthfulness scores about categories except food have obvious deduction, especially for the category ambience. This means that statements on categories like price and ambience are more controversial and subjective, since many "positive" statements about these categories are considered of low trustfulness.



(a) (b) (c)

Fig. 2: Distribution of scores when sentiments of statements are set based on majority opinions. (a) Distribution of honesty scores for users. (b) Distribution of faithfulness scores for reviews. (c) Distribution of truthfulness scores for statements



(a) (b) (c)

Fig. 3: Distribution of the scores when sentiments of statements are all set to be positive. (a) Distribution of honesty scores for users. (b) Distribution of faithfulness scores for reviews. (c) Distribution of truthfulness scores for statements

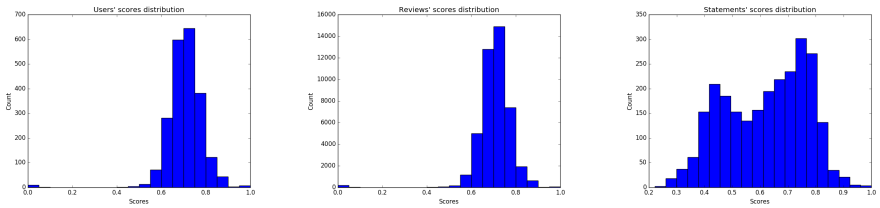
For score distributions, we first presents the results of the experiment where we set the sentiments based on majority opinions, as it is the most intuitive setting. The distributions of the honesty, faithfulness, and truthfulness scores are shown in Fig. 2. The results show that for users and reviews, the scores roughly follows normal distributions with mean both around 0.75, which indicate that most of the users and reviews are somehow with some biased opinions but still honest. The distribution of truthfulness scores are somehow skewed and pushed to the right side. The results indicate that most claims are of high truthfulness since they are initialized based on majority opinions.

We did the experiments under another setting to make sure our model works as we expected. In the second setting, we make initial sentiment polarities of the statements set to be positive. Under this setting, what can be expected is that some statements would be false since in reality, they do not have that kind of positive feedback, and thus these statements will receive much lower truthfulness scores. However, the scores of users and reviews will not be affected too much since our model will always award the users who express opinions that are consistent with the majorities and penalized those who do not. In the second set of experiments, when setting sentiment polarities of all statements to be positive, the distributions of scores are shown in Fig. 3. The most obvious change is that the distribution of truthfulness of statements is divided into two parts, which is as expected from the results in Table 3. A part of statements receive scores lower than 0.4, indicating these are the statements that becomes false because of the arbitrary initialization of positive sentiment. Another part of statements still have relatively high scores as they are still true under this setting. Note that the changes in the distribution of statements scores do not mean that our model is sensitive to initializations. The drop of truthfulness scores of some statements is caused by the unreliable initial values of statements (arbitrarily set to positive). During the trust propagation in our model, some of truthfulness of the statements are penalized since the majority do not agree that it should be positive. In fact, the changes of statements scores reflect how our model treats unreliable statements and it is exactly what we expect to see. For honesty and faithfulness, the distributions left shift a bit as some users are affected by the false statement. As expected, the honesty scores and faithfulness scores did not change much.

Evaluations In this work, we did two kinds of evaluations, add synthetic data and use human evaluators. The purpose of using synthetic data is to test whether our model works in the way as we expected. To achieve this goal, we modified the data of 20 users in the dataset and changed them to extreme cases. 10 users are changed to fully support all the statements their reviews expressed. The rest 10 users are changed to reject all the statements their reviews expressed. With the modified data, we conducted experiments under the setting that sentiments of statements are set based on majority opinions. The distributions of the scores are shown in Fig. 4. The scores of the users with synthetic data are shown in Table 4.

Table 4: Average honesty scores of users with synthetic data

Synthetic type	Min	Average	Median	Max
Support	0.784	0.865	0.863	0.942
Reject	6.842e-12	0.002	6.842e-12	0.013



(a) (b) (c)

Fig. 4: Distribution of the scores with synthetic data. (a) Distribution of honesty scores for users. (b) Distribution of faithfulness scores for reviews. (c) Distribution of truthfulness scores for statements

The results of the synthetic data show that the model works the way as expected, to award users who agree with majority opinions and penalize users who do not. As mentioned before, we argue that the majority opinions reflect the truth about the qualities of items in the review system, thus the honesty and faithfulness scores we defined reflect the trustworthiness of users and reviews respectively.

As for human evaluation, three human evaluators were involved. We randomly selected twenty users from our dataset as the subject of the evaluation. For each user, eight reviews were randomly picked. Since every two users could form a pair, there would be 190 pairs in total. We randomly picked 20 pairs to compare the users' extent of honesty by asking the three evaluators to read their reviews. For every pair of users, the evaluators were instructed to make a judgment of which user was more honest. For example, for two users u_1 and u_2, the judgment of honesty is either $u_1 > u_2$ or $u_1 < u_2$. We conducted two steps of evaluations. In the first step, the only information about a user we provided for the human evaluator was the eight randomly selected reviews. In the second step, along with the reviews, we provided the ratings about the reviewed restaurants on Yelp.com as facts of qualities.

Table 5: Agreement in first evaluation

	Our model	Evaluator 1	Evaluator 2	Evaluator 3
Our model		13	5	7
Evaluator 1	13		10	10
Evaluator 2	5	10		12
Evaluator 3	7	10	12	

After we got the judgment from the evaluators, we compared the judgments with the results of users' honesty scores from our proposed model. In the model, for each user, his/her honesty score can be computed. For each pair of users

u_1 and u_2, we can make a judgment according to the honesty scores $h(u_1)$ and $h(u_2)$. The agreements among human evaluators and our model of first level and second level evaluations are shown in Table. 5 and Table. 6 respectively. Here the agreement means that the judgment of whether a user is more honest than the other is consistent between two results. For example, in not meaningful. In the first evaluation, human evaluators just read the randomly selected reviews and have no reference for quality judgment. By barely reading reviews, the evaluators tend to make arbitrary judgments and the agreements between human and our model are relatively low. In the second evaluation, evaluators have ratings of the reviewed targets from Yelp.com as the group truth for qualities. Comparing the results of the two tables, we find that the agreements between human evaluators and our model in the second evaluation are higher than in the first one, which means that by providing the actual ratings of the restaurants as facts, the human evaluators were able to make more reasonable judgments and achieved better consistency with our model. Also, comparing to the intra-human agreements, the agreements between human evaluators and our model are pretty acceptable.

Table 6: Agreement in second evaluation

	Our model	Evaluator 1	Evaluator 2	Evaluator 3
Our model		13	12	13
Evaluator 1	13		7	12
Evaluator 2	12	7		13
Evaluator 3	13	12	13	

To further analyze the agreements between evaluators and our model, out of the agreements between each pair of evaluators, we computed the ratio of overlapping agreements of model and the pair of evaluators (e.g., the ratio of agreements of model, evaluator 1, and evaluator 2 over the agreements between evaluator 1 and evaluator 2). The computed ratios for the two evaluations are shown in Table 7. The increased ratio of overlapping agreements in the second evaluation indicates that with proper reference of quality, evaluators tend to make similar judgments with our model. The judgments that our model disagree with the evaluators, the evaluators themselves are also unlikely to agree with each other. The evaluations show that our model is able to achieve higher consistency with the human evaluators when they have fair reference of qualities. Thus, our model is able to evaluate the extent of honesty of users.

6 Conclusion

In this work, we study the problem of inferring trustworthiness from the content of online reviews. We first apply opinion-mining techniques using supervised learning algorithms to extract opinions that are expressed in the reviews. Then, we integrate the opinions to obtain opinion vectors for individual reviews and

Table 7: Ratio of overlapping agreements between model and each pair of evaluators in two evaluations

	Evaluator 1 & 2	Evaluator 1 & 3	Evaluator 2 & 3
First evaluation	0.400	0.500	0.167
Second evaluation	0.857	0.750	0.692

statements. Finally, we develop an iterative content-based computational model to compute honesty scores for users, reviews, and statements. According to the results, there exist differences of statement truthfulness across different categories. Our model shows that the trustworthiness of a user is closely related to the content of her reviews. The review dataset we used was collected in 2013. The structures and content in the dataset are static and there is no dynamic changes considered in our model. However, the reviews and qualities of restaurants tend to change with time. In order to take the dynamic changes into account, we plan to add a temporal dimension in our model in the future. For the opinion mining task, we applied a supervised learning model and used a labeled dataset. However, manually labeling dataset is usually both labor-intensive and time consuming. In our next step, we will apply unsupervised learning methods such as word2vec to group the aspect categories, and thus to automate the opinion mining process.

References

1. Akoglu, L., Chandy, R., Faloutsos, C.: Opinion fraud detection in online reviews by network effects. In: ICWSM (2013)
2. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: LREC. vol. 10, pp. 2200–2204 (2010)
3. Brody, S., Elhadad, N.: An unsupervised aspect-sentiment model for online reviews. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. pp. 804–812. Association for Computational Linguistics (2010)
4. Chen, Z., Mukherjee, A., Liu, B.: Aspect extraction with automated prior knowledge learning. In: ACL (1). pp. 347–358 (2014)
5. Fahrni, A., Klenner, M.: Old wine or warm beer: Target-specific sentiment analysis of adjectives. In: Proc. of the Symposium on Affective Language in Human and Machine, AISB. pp. 60–63 (2008)
6. Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R.: Exploiting burstiness in reviews for review spammer detection. In: ICWSM (2013)
7. Ganu, G., Elhadad, N., Marian, A.: Beyond the stars: Improving rating predictions using review text content. In: WebDB. vol. 9, pp. 1–6 (2009)
8. Guha, R., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proceedings of the 13th international conference on World Wide Web. pp. 403–412. ACM (2004)
9. Hai, Z., Chang, K., Kim, J.J., Yang, C.C.: Identifying features in opinion mining via intrinsic and extrinsic domain relevance. IEEE Transactions on Knowledge and Data Engineering 26(3), 623–634 (2014)

10. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 168–177. ACM (2004)
11. Jijkoun, V., Hofmann, K.: Generating a non-english subjectivity lexicon: relations that matter. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. pp. 398–405. Association for Computational Linguistics (2009)
12. Jindal, N., Liu, B.: Opinion spam and analysis. In: Proceedings of the 2008 International Conference on Web Search and Data Mining. pp. 219–230. WSDM '08, ACM, New York, NY, USA (2008)
13. Jo, Y., Oh, A.H.: Aspect and sentiment unification model for online review analysis. In: Proceedings of the fourth ACM international conference on Web search and data mining. pp. 815–824. ACM (2011)
14. Jøsang, A., Marsh, S., Pope, S.: Exploring different types of trust propagation. In: International Conference on Trust Management. pp. 179–192. Springer (2006)
15. Kennedy, A., Inkpen, D.: Sentiment classification of movie reviews using contextual valence shifters. Computational intelligence 22(2), 110–125 (2006)
16. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM) 46(5), 604–632 (1999)
17. Konishi, T., Tezuka, T., Kimura, F., Maeda, A.: Estimating aspects in online reviews using topic model with 2-level learning. In: Proceedings of the International MultiConference of Engineers and Computer Scientists. vol. 1, pp. 120–126 (2012)
18. Lim, E.P., Nguyen, V.A., Jindal, N., Liu, B., Lauw, H.W.: Detecting product review spammers using rating behaviors. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management. pp. 939–948. CIKM '10, ACM, New York, NY, USA (2010)
19. Liu, Q., Gao, Z., Liu, B., Zhang, Y.: A logic programming approach to aspect extraction in opinion mining. In: Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on. vol. 1, pp. 276–283. IEEE (2013)
20. Luca, M., Zervas, G.: Fake it till you make it: Reputation, competition, and yelp review fraud. Management Science 62(12), 3412–3427 (2016)
21. Massa, P., Avesani, P.: Trust-aware recommender systems. In: Proceedings of the 2007 ACM conference on Recommender systems. pp. 17–24. ACM (2007)
22. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM 38(11), 39–41 (1995)
23. Mukherjee, A., Liu, B., Wang, J., Glance, N., Jindal, N.: Detecting group review spam. In: Proceedings of the 20th International Conference Companion on World Wide Web. pp. 93–94. WWW '11 (2011)
24. Mukherjee, S., Dutta, S., Weikum, G.:
25. Ott, M., Choi, Y., Cardie, C., Hancock, J.T.: Finding deceptive opinion spam by any stretch of the imagination. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1
26. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: bringing order to the web. (1999)
27. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd annual meeting on Association for Computational Linguistics. p. 271. Association for Computational Linguistics (2004)

28. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. pp. 79–86. Association for Computational Linguistics (2002)
29. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
30. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: Semeval-2014 task 4: Aspect based sentiment analysis. In: Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014). pp. 27–35 (2014)
31. Popescu, A.M., Etzioni, O.: Extracting product features and opinions from reviews. In: Natural language processing and text mining, pp. 9–28. Springer (2007)
32. Poria, S., Cambria, E., Ku, L.W., Gui, C., Gelbukh, A.: A rule-based approach to aspect extraction from product reviews. In: Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP). pp. 28–37 (2014)
33. Qiu, G., Liu, B., Bu, J., Chen, C.: Opinion word expansion and target extraction through double propagation. Computational linguistics 37(1), 9–27 (2011)
34. Salvetti, F., Lewis, S., Reichenbach, C.: Automatic opinion polarity classification of movie. Colorado research in linguistics 17(1), 2 (2004)
35. Than, C., Han, S.: Improving recommender systems by incorporating similarity, trust and reputation. Journal of Internet Services and Information Security (JISIS) 4(1), 64–76 (2014)
36. Titov, I., McDonald, R.: Modeling online reviews with multi-grain topic models. In: Proceedings of the 17th international conference on World Wide Web. pp. 111–120. ACM (2008)
37. Turney, P.D.: Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 417–424. Association for Computational Linguistics (2002)
38. Vydiswaran, V., Zhai, C., Roth, D.: Content-driven trust propagation framework. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 974–982. ACM (2011)
39. Wang, G., Xie, S., Liu, B., Yu, P.S.: Identify online store review spammers via social review graph. ACM Trans. Intell. Syst. Technol. 3(4), 61:1–61:21 (Sep 2012)
40. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. Computational linguistics 35(3), 399–433 (2009)
41. Xie, S., Wang, G., Lin, S., Yu, P.S.: Review spam detection via temporal pattern discovery. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 823–831. KDD '12, ACM, New York, NY, USA (2012)
42. Xue, H., Li, F., Seo, H., Pluretti, R.: Trust-aware review spam detection. In: Trustcom/BigDataSE/ISPA, 2015 IEEE. vol. 1, pp. 726–733. IEEE (2015)
43. Yin, X., Han, J., Philip, S.Y.: Truth discovery with multiple conflicting information providers on the web. IEEE Transactions on Knowledge and Data Engineering 20(6), 796–808 (2008)