Text Mining and Probabilistic Language Modeling for Online Review Spam Detection

RAYMOND Y. K. LAU, S. Y. LIAO, and RON CHI-WAI KWOK, City University of Hong Kong
KAIQUAN XU, Nanjing University
YUNQING XIA, Tsinghua University
YUEFENG LI, Queensland University of Technology

In the era of Web 2.0, huge volumes of consumer reviews are posted to the Internet every day. Manual approaches to detecting and analyzing fake reviews (i.e., spam) are not practical due to the problem of information overload. However, the design and development of automated methods of detecting fake reviews is a challenging research problem. The main reason is that fake reviews are specifically composed to mislead readers, so they may appear the same as legitimate reviews (i.e., ham). As a result, discriminatory features that would enable individual reviews to be classified as spam or ham may not be available. Guided by the design science research methodology, the main contribution of this study is the design and instantiation of novel computational models for detecting fake reviews. In particular, a novel text mining model is developed and integrated into a semantic language model for the detection of untruthful reviews. The models are then evaluated based on a real-world dataset collected from amazon.com. The results of our experiments confirm that the proposed models outperform other well-known baseline models in detecting fake reviews. To the best of our knowledge, the work discussed in this article represents the first successful attempt to apply text mining methods and semantic language models to the detection of fake consumer reviews. A managerial implication of our research is that firms can apply our design artifacts to monitor online consumer reviews to develop effective marketing or product design strategies based on genuine consumer feedback posted to the Internet.

Categories and Subject Descriptors: I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language models; Text analysis*; H.2.8 [**Database Applications**]: Data mining; H.3.3 [**Information Search and Retrieval**]: Retrieval models

General Terms: Design, Algorithms, Experimentation

Additional Key Words and Phrases: Language models, text mining, review spam, spam detection, design science

ACM Reference Format:
Lau, R. Y. K., Liao, S. Y., Kwok, R. C. W., Xu, K., Xia, Y., and Li, Y. 2011. Text mining and probabilistic language modeling for online review spam detection. ACM Trans. Manag. Inform. Syst. 2, 4, Article 25 (December 2011), 30 pages.
DOI = 10.1145/2070710.2070716 http://doi.acm.org/10.1145/2070710.2070716

The work reported in this article has been funded in part by Hong Kong RGC's GRF research grant (Project no. 9041569) and the City University of Hong Kong's SRG grant (Project no. 7002426).
Author's addresses: R. Y. K. Lau (corresponding author), S. Y. Liao, and R. C. W. Kwok, Department of Information Systems, City University of Hong Kong, China; email: raylau@cityu.edu.hk; K. Xu, Department of Electronic Business, Nanjing University, China; Y. Xia, Department of Computer Science and Technology, Tsinghua University, China; Y. Li, School of IT, Queensland University of Technology, Australia.

DOI 10.1145/2070710.2070716 http://doi.acm.org/10.1145/2070710.2070716

1. INTRODUCTION

In recent years, there has been an explosive increase in the volumes of user-contributed consumer reviews posted to e-commerce Web sites (e.g., amazon.com) or opinion sharing Web sites (e.g., epinions.com). However, the widespread sharing and utilization of user-contributed reviews has also raised concerns about the trustworthiness of these items [Cheung et al. 2009; Dellarocas 2003, 2006; Lau et al. 2010; Lim et al. 2010]. For example, the BBC has reported that fake reviews are becoming a common problem on the Web, and a photography company was recently subjected to hundreds of defamatory consumer reviews.[1] The issue of fake reviews has also been widely discussed in the business press and has even been raised in legal circles.[2] With regard to email and Web spam [Cormack et al. 2007a; Gyöngyi and Garcia-Molina 2005], the primary objective of spammers is to utilize keywords to motivate readers to traverse to adversarial merchant sites or fool search engine indexing programs to generate higher page ranks for adversarial Web pages. Accordingly, certain discriminatory "features," such as an extraordinary number of URLs or particular advertising keywords, are attached to individual spam email messages or Web pages. These features can be utilized by spam detection programs to distinguish between spam and ham.

However, existing supervised machine learning techniques [Abbasi et al. 2010; Bratko et al. 2006; Cormack et al. 2007a, 2007b], which can successfully learn the prominent features from an individual email or Web page to identify spam, may not be effective in dealing with the review spam detection problem. The main reason is that fake reviews are mainly used by spammers to mislead readers into believing that the review contents are true. Therefore, the necessary distinguishing features may not be available to human readers or spam detection programs to separate the spam from ham. Figure 1 depicts an example of a fake review posted to amazon.com, which was detected by our prototype review spam detection system. The review concerns the Canon PowerShot SD600. As can be observed from Figure 1, even after examining all of the possible features (e.g., occurrence of special keywords, frequency of sentiment indicators, occurrence of URLs, and number of helpful votes) associated with the review, it may still be difficult for human readers or spam detection programs to determine whether the review is fake or legitimate. In fact, this fake review attracted 14 out of 15 helpful votes.

Nevertheless, if another review, that depicted in Figure 2, is presented, then a reader or spam detection program will be able to find the necessary feature to classify both reviews as suspicious fake reviews; the feature is the "semantic overlapping" of the contents of these reviews. The second review concerns a totally different product, the Kodak EasyShare C875 camera. However, the two reviews are almost the same, except that some of the wording has been deliberately changed. Both reviews refer to the same type of shopping experience, even though they are about two different products transacted on two different occasions. Obviously, the existing review moderation process (manual or automatic) adopted by amazon.com is not that effective in dealing with the review spam problem, as these fake reviews have been left online for many years! We believe that manual approaches of detecting fake reviews are not feasible because of the problem of information overload [Lau et. al. 2008; Yan et al. 2011]. The main intuition of our proposed review spam detection method is to use the semantic content overlapping among reviews as the basis to judge whether certain reviews are likely to be spam. In fact, content diversity (the opposite of content overlapping) has also been

[1] http://news.bbc.co.uk/2/hi/programmes/click_online/8826258.stm.
[2] http://www.nytimes.com/2009/07/15/technology/internet/15lift.html.

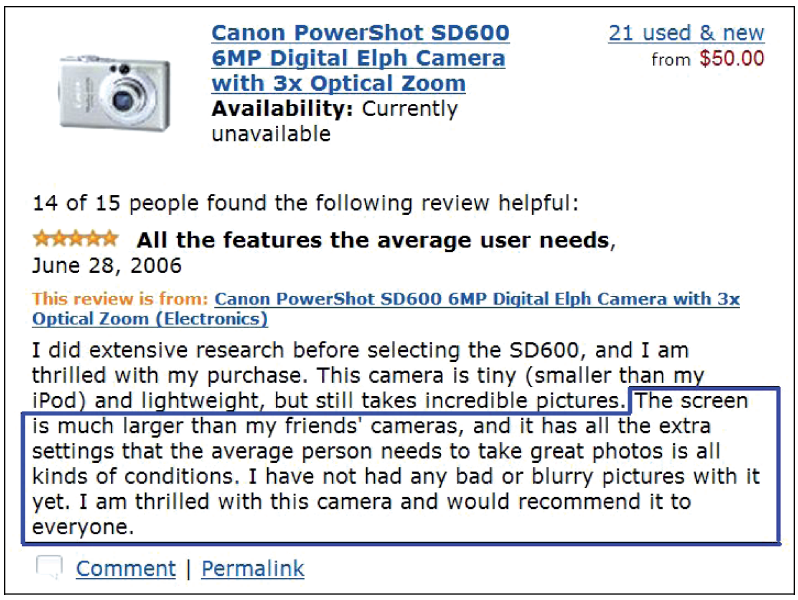

Fig. 1. The first example of a fake review.

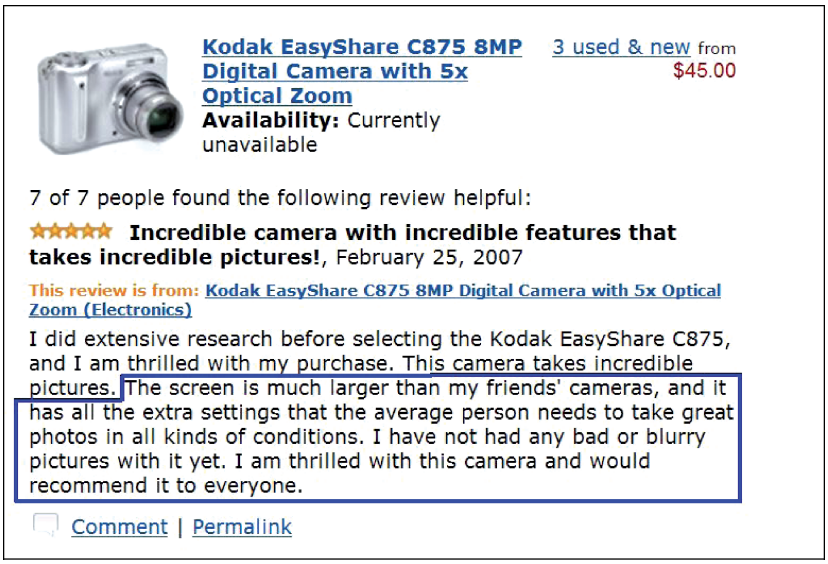

Fig. 2. The second example of a fake review.

successfully applied to the detection of blog spam [Mishne et al. 2005] and Web spam [Martinez-Romo and Araujo 2009].

Jindal and Liu [2007a, 2007b, 2008] define three types of review spam, namely untruthful reviews, brand-only reviews, and nonreviews (e.g., commercial advertisements). We believe that brand-only reviews are a kind of nonreview. Accordingly, this article discusses the computational models for detecting untruthful reviews and nonreviews. We pay particular attention to untruthful reviews, as they have been

recognized as presenting a more challenging research problem [Jindal and Liu 2008; Lim et al. 2010]. Untruthful reviews refer to reviews that pretend to reflect the writers' true opinions [Jindal and Liu 2008; Lau et al. 2010]. Figures 1 and 2 are examples of untruthful reviews. However, as spammers often adopt obfuscation strategies by deliberately modifying certain contents of a source to create spam [Abbasi et al. 2008; Fetterly et al. 2005], detecting untruthful reviews becomes an even more challenging task. Accordingly, an effective detection model should be able to detect this kind of content modification (e.g., word substitutions), perhaps by referring to the semantic term relationships or high-order concept association relationships discovered via text mining processes [Lau 2003; Lau et al. 2008]. For example, if the term "love" in one review is replaced by the term "like" to compose an untruthful review, it is possible for a detection system to identify the semantic content overlapping between these reviews based on the "love"→"like" relationship.

With respect to the fundamental issues involved in detecting fake reviews, an effective system of review spam detection should satisfy the following orthogonal requirements. First, the system should be accurate enough to reduce the chances of misclassifying ham as spam or vice versa. Second, the system should be efficient enough to process the large volumes of reviews currently being posted to the Internet. Third, the system should be equipped with a component capable of handling spammers' obfuscation strategies, which make fake reviews look like the legitimate ones. Fourth, the system should be able to detect different types of fake reviews. Finally, the system should be able to manage the uncertainty involved in review spam detection by providing probabilistic estimations of the detection results.

Guided by the design science research methodology [Hevner et al. 2004; March and Storey 2008; Peffers et al. 2008], one of the main contributions of our research is the design and development of a novel unsupervised detection model for combating untruthful reviews. Our unsupervised spam detection model is able to address the problem of "missing features in an individual review" related to untruthful review detection. In particular, a novel semantic language model is designed and applied to estimate the overlapping of semantic contents among reviews, and hence to identify untruthful reviews. Our proposed semantic language modeling approach for untruthful review detection is different from the traditional plagiarized content detection method [Chowdhury et al. 2002], in that it is able to take "substituted" terms into account when estimating the semantic content similarity among reviews. Moreover, we address the knowledge acquisition problem by developing a high-order concept association mining method to extract context-sensitive concept association knowledge. This knowledge can then be utilized by the proposed semantic language model to ascertain possible "concept substitutions" in untruthful reviews. Another main contribution of our work relates to the instantiation of our design such that rigorous experiments can be applied to evaluate the effectiveness of our proposed model. Above all, our design artifacts have allowed us to conduct an empirical study of the trustworthiness of online consumer reviews posted to the Internet. This study is one of the few large-scale empirical analyses of the trustworthiness of online consumer reviews conducted to date.

Though our current prototype system is not a fully-fledged commercial system for review spam detection, the proposed computational models are the critical modules for a fully functional review spam detection system. The managerial implications of our research are twofold. First, business managers and marketers will apply the design artifacts to identify and analyze fake reviews related to their products and services. Accordingly, more effective product design strategies and marketing plans are developed based on the volume of genuine consumer comments posted to the Internet. Second, online merchants will apply our design artifacts to continuously monitor and moderate user-contributed online reviews to maintain the quality of the reviews and the

reputation of the merchants' products. A societal implication of our research is that individual consumers will apply our design artifacts to identify genuine consumer reviews of the products they wish to purchase, thereby facilitating their comparative shopping processes on a daily basis.

The rest of the article is organized as follows. The next section highlights previous research related to the detection of review (opinion) spam, email spam, and Web spam in general. The system architecture of the review spam detection system is highlighted in Section 3. The computational models for the detection of fake reviews and high-order concept mining are described in Sections 4 and 5, respectively. Section 6 discusses the evaluation of our design artifacts, and Section 7 explains the application of our design artifacts to empirically assess the trustworthiness of online consumer reviews. Finally, we offer concluding remarks and describe future directions of our research.

2. RELATED RESEARCH

Lim et al. [2010] investigated the review spam detection problem by tracking spammers' rating behavior. In particular, they proposed several heuristics (e.g., reviewers who rated a large number of products with similar ratings, reviewers who rated a product in the earliest period and gave ratings deviating from the average rating for the entire period, etc.) to track spamming behavior and developed the corresponding quantitative measures to identify spam patterns. The top k-precision metric was used to assess the effectiveness of different heuristics and the corresponding quantitative measures. Even though content similarity was used as the basis to detect untruthful reviews, the obfuscation actions (e.g., deliberately substituting words in fake reviews) taken by spammers were not taken into account by the system [Lim et al. 2010].

Research in untruthful review detection also attempted to discover "unexpected rules" to pinpoint spammers' abnormal behavior [Jindal et al. 2010]. Instead of using predefined heuristics, the classical notions of rule support and rule confidence developed in the data mining community were used to construct novel measures to quantify confidence unexpectedness and support unexpectedness. These measures were then used to track the abnormal activities of spammers. A spammer-behavior-based detection method was developed to identify spamming activities and content promotion in online video social networks [Benevenuto et al. 2009]. More specifically, a supervised machine learning method, namely, the Support Vector Machine (SVM) [Joachim 1998], was applied to train classifiers to detect spammers and content promoters. For content promoters, who mainly post commercial advertisements, it may be feasible to use supervised machine learning techniques to identify the discriminatory features that would enable nonreviews to be detected. However, such discriminatory features may not be available in individual untruthful reviews. This article examines computational models for detecting nonreviews and untruthful reviews.

Jindal and Liu [2007a, 2007b, 2008] classified three types of review spam, namely, untruthful opinions, brand only spam, and nonreviews. A logistic regression model (a supervised classification method) was applied to detect three types of fake reviews. Duplicated reviews were first identified using the Jaccard ratio, and these reviews were then used as the training examples for the logistic regression classifier. In this article, we explore different computational methods (e.g., supervised and unsupervised classification methods) to tackle different types of fake reviews. Moreover, the uncertainty involved in review spam detection is managed by generating a probabilistic ranking of detected fake reviews.

Xiao and Benbasat [2011] have proposed a typology of product-related deceptive information practices to identify the various ways online traders deceive consumers at e-Commerce sites. The work reported in this article extends that of Xiao and Benbasat [2011] by developing a novel model to combat deception related to online reviews.

Abbasi et al. [2008] proposed a stylometric approach to identify online traders based on the writing style traces embedded in traders' online comments. In particular, the Karhunen-Loeve transform was applied to generate n-dimensional feature vectors, called a Writeprint, to represent a trader's writing pattern. Their method was evaluated using 600K online comments contributed by 200 eBay traders [Abbasi et al. 2008]. The primary objective of our research is to determine whether online comments are truthful or not and the efficiency of the proposed system is empirically tested using 2 million online reviews.

A taxonomy of Web spam was developed to analyze the common techniques applied in Web page spamming [Gyöngyi and Garcia-Molina 2005]. Web spam refers to Web pages deliberately created to trigger unjustifiably favorable relevance or importance. Based on random samples crawled from the Web, it was estimated that around 10–15% of the contents on the Web are spam [Gyöngyi and Garcia-Molina 2005]. Content-based [Ntoulas et al. 2006] and link-based [Zhou and Pei 2009] Web page spamming were both examined. These Web spam techniques are mainly used to fool search engines rather than humans to obtain a higher page rank for the target Web pages. It was pointed out that Web spam can be automatically generated by stitching together phrases drawn from a limited corpus [Gyöngyi and Garcia-Molina 2005]. Linguistic features were also examined for Web spam detection [Piskorski et al. 2008]. However, this article focuses on the more challenging problem of review spam detection.

Blog spam, a special case of Web page spam, was examined using probabilistic unigram language models and Kullback-Leibler (KL) divergence [Kullback and Leibler 1951] to distinguish spam blogs (splogs) from normal blog posts [Mishne et al. 2005]. Martinez-Romo and Araujo [2009] also employed unigram language models for Web spam detection. The work reported in this article deals with a much more challenging research problem, because the language usage in legitimate consumer reviews and fake reviews can be quite similar. Lin et al. [2008] employed self-similarity matrices and a histogram intersection similarity measure to analyze the regularities in blog posts over time. Macdonald et al. [2009] discussed the issue of spam in relation to opinionated blog posts although they did not propose a technique to automatically detect splogs.

Studies have examined the helpfulness of user-generated consumer reviews [Danescu-Niculescu-Mizil et al. 2009; Ghose and Ipeirotis 2007; Kim et al. 2006; Liu et al. 2008]. Although the prediction of review helpfulness is related to the identification of review spam, the goals of these tasks and the underlying techniques are quite different. Kim et al. [2006] applied an SVM regression model to examine the correlations between the structural, lexical, syntactic, semantic, and metafeatures of reviews and their helpfulness ratings. Unfortunately, these user-generated helpfulness votes can also be spam. Therefore, using helpfulness votes as the basis for labeling reviews for supervised classifier training may not be effective.

Recent research has applied the least-square SVM, which is a supervised machine learning method, to conduct coclassification based on bookmarks and user data to detect spam in social networks [Chen et al. 2009]. In the context of spam detection for short SMS messages, several supervised machine learning methods have been examined [Bratko et al. 2006; Chang et al. 2008; Cormack et al. 2007a]. Zheleva et al. [2008] examined a user-based reputation management system which makes use of the feedback from trustworthy users to identify spam emails. Given the fact that it is difficult for readers to distinguish between fake reviews and legitimate reviews [Jindal and Liu 2008], it may not be feasible to rely on user-based reputation management systems to identify fake reviews.

The Vector Space model (VS) has been widely used in information retrieval research [Salton and McGill 1983]. Each document (e.g., a review) is first characterized by a corresponding vector of term weights. The cosine angles of these vectors can then be

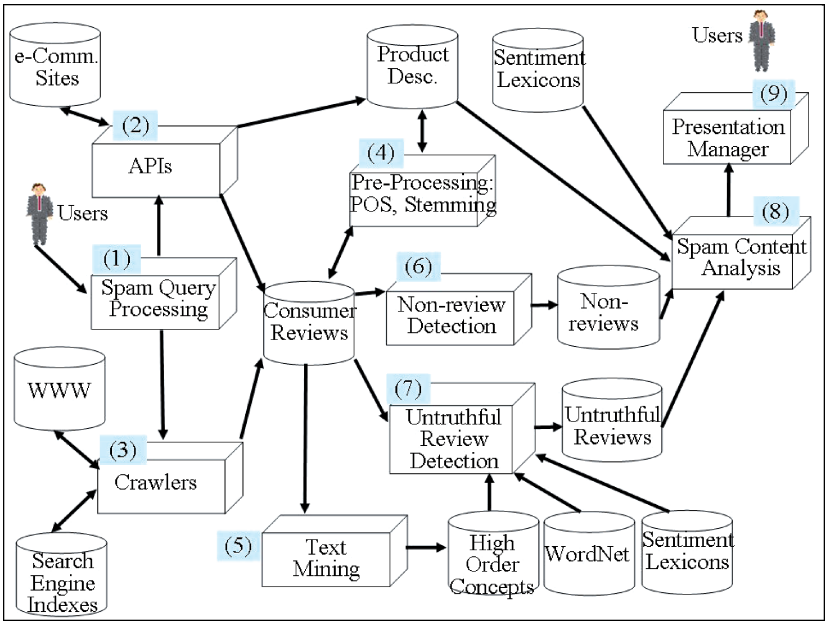

Fig. 3. The general system architecture of the review spam detection system.

computed to estimate the similarities of the documents [Salton and McGill 1983]. The I-Match system was developed to identify partially duplicated contents in terms of overlapping words or characters [Chowdhury et al. 2002]. The vector space model and the I-Match model are both unsupervised classification models and can be applied to the detection of untruthful reviews without requiring human-labeled training examples. Supervised machine learning techniques such as SVM [Joachim 1998] and the K-Nearest Neighbor (KNN) classifier [Mitchell 1997] can be applied to learn the discriminatory features that characterize each individual object (e.g., a review). These techniques can be applied to classify non-reviews by training the models using the non-reviews labeled by humans.

3. SYSTEM ARCHITECTURE

In this study, the design artifacts refer to the computational methods used for the detection of fake reviews and the instantiation of our design (i.e., a prototype system). The general system architecture of the prototype system of review spam detection and analysis is depicted in Figure 3. The arcs in Figure 3 represent the data flow among the various system modules. The specific functionality of each module is described as follows. Module 1: first, the user selects the detection scope (e.g., all the reviews of a product category) for the review spam detection. This user requirement is translated into a spam detection query to be processed by the system. Module 2: if the reviews of a product are not yet available in the system's local database or have not been updated for a predefined period of time, then our system will utilize the Web services or Application Programming Interfaces (APIs) provided by external e-commerce sites (e.g., amazon.com,[3] cnet.com,[4] etc.) to retrieve consumer reviews and related product information. Module 3: if APIs are not provided by an e-Commerce site or an Internet forum

[3]http://ecs.amazonaws.com/onca/xml?Service = AWSECommerceService.
[4]http://api.cnet.com/.

(e.g., epinions.com), then general search engines such as Google and dedicated crawler programs will be invoked to retrieve the online reviews. In our current implementation, the user can use either the Amazon Standard Identification Number (ASIN) or a product name to compose her query. An ASIN represents a unique product sold via amazon.com. Just like the invocation of Web services, our crawler programs are also sent to retrieve the latest information about products and reviews, and they periodically update this information in the local database. Module 4: traditional document preprocessing procedures [Salton and McGill 1983], such as stop-word removal, Part-of-Speech (POS) tagging, and stemming [Porter 1980], are then applied to process the product reviews and product descriptions retrieved from the Web. The POS tags are used when the syntactical feature analysis is invoked during the spam product feature analysis. Review preprocessing is invoked when the reviews are saved to the local database.

Module 5: after the reviews are preprocessed, the high-order concept association mining module is invoked to extract the prominent concepts and their high-order associations for each product category. These association relationships are used to bootstrap the performance of the semantic language model to detect untruthful reviews using the obfuscation strategies exercised by spammers. The text mining module is invoked periodically and executed as a background task. The details of the computational method for high-order concept associations mining are illustrated in Section 5. Module 6: non-review detection is performed by a supervised SVM classifier. Module 7: untruthful review detection is carried out independently by an unsupervised probabilistic language model. Both detection modules refer to the same information source (i.e., the downloaded consumer reviews), but they work independently (e.g., each module produces a probabilistic ranking of a specific type of suspicious fake reviews). In this article, we mainly focus on the detection of untruthful reviews because this is regarded as a more challenging research problem [Jindal and Liu 2008]. The computational details for untruthful review detection and non-review detection are illustrated in Section 4. The outputs generated by Modules 6 and 7 are consumed by the spam content analysis module (Module 8). More specifically, the contents of fake reviews (e.g., specific product features and their sentiments) are analyzed by the spam content analysis module. The spam content analysis module is constructed based on a previously developed opinion mining method [Lau et al. 2009b]. Due to space limitations, this module is not discussed in this article. Module 9: finally, the detected fake reviews and the results of a fine-grained spam content analysis are presented to the users via the presentation manager. Textual and graphical displays are used to highlight the fake reviews and the characteristics of the spam contents. Our instantiation[5] (i.e., a prototype system) was developed using Java (J2SE v 5.0), Java Server Pages (JSP) 2.1, and Servlet 2.5. The prototype system is hosted on a DELL 1950 III Server with Quad-Core Xeon 2.33 GHz processors, 16GB main memory, and 6TB hard disk. A screen shot of a typical spam detection interface of the prototype system is shown in Figure 4. The user first selects a product category (e.g., PC Hardware) and the list of products with fake reviews identified by the prototype system is then displayed. For untruthful review detection, the user can choose a pair of suspicious reviews from a probabilistic ranking of untruthful reviews to examine the contents of the original reviews.

4. THE COMPUTATIONAL MODELS FOR REVIEW SPAM DETECTION

4.1. The Computational Model for Untruthful Review Detection

In this article, untruthful reviews loosely refer to spammers' false comments (opinions) about particular products or services [Jindal and Liu 2008]. It is not practical

[5]http://quantum.is.cityu.edu.hk/om/.

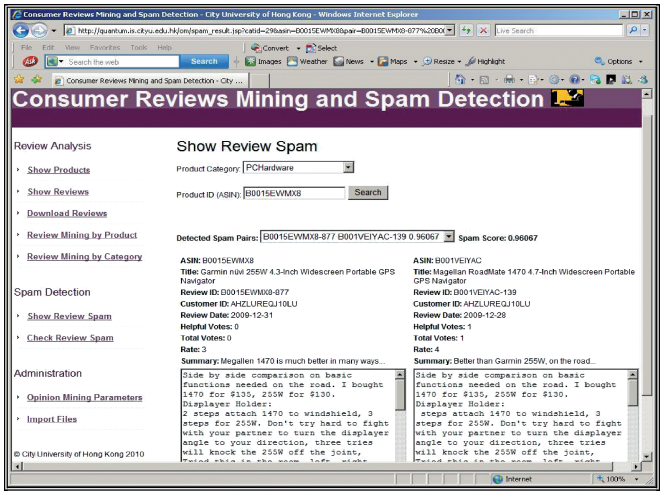

Fig. 4. A snapshot view of the prototype system.

to directly measure the concept of "untruthfulness" because a computer cannot read a review writer's mind. Therefore, we propose an approximation method which indirectly estimates the degree of "untruthfulness" based on the "overlapping" of the semantic contents of reviews. If the semantic contents of a review are mainly generated from another review, this suggests that both reviews may not sincerely reflect writers' true opinions. Figures 1 and 2 show examples of untruthful reviews that demonstrate a high degree of semantic content overlapping. However, we do not claim that all reviews with high semantic content overlapping must be fake reviews. Our aim is to develop a probabilistic ranking of suspicious fake reviews and, hence, to facilitate users' final decisions about the truthfulness of reviews. Such a scenario is similar to the functionality of popular Internet search engines. While a search engine cannot fully understand the information requirements of users, it can produce ranked lists of possibly interesting items and let the users decide the relevant items. Nonetheless, Internet search engines have been shown very useful in practice.

We extend the well-known Language Modeling (LM) framework [Lafferty and Zhai 2001; Liu and Croft 2004; Ponte and Croft 1998] to develop a novel semantic-based smoothing method to estimate the likelihood of semantic content generation among reviews. Moreover, we apply Kullback-Leibler (KL) divergence [Kullback and Leibler 1951] to measure the distance of the language models that represent individual reviews. Our proposed semantic language modeling approach is different from the traditional plagiarized content detection method [Chowdhury et al. 2002] in that "substituted" terms can be taken into account when estimating the semantic similarity of reviews. Figure 5 depicts two cases of obfuscation. Figures 5(a) and 5(b) show that the spammer has deliberately modified some words in the fake reviews. Figures 5(c) and 5(d) show that the spammer has deliberately modified the titles of the fake reviews. To effectively detect fake reviews, the proposed semantic language models should take into account relationships such as ("love"→"like") when estimating the semantic similarity of reviews. In fact, the term "like" is a synonym of the term "love" according to WordNet [Miller et al. 1990]. For term association relationships not defined in WordNet, such as ("fabulous"→"fantastic"), a text mining method is applied to dynamically discover the term association relationships to detect the spammers' obfuscation actions. The high-order concept association mining method is discussed in Section 5. The main

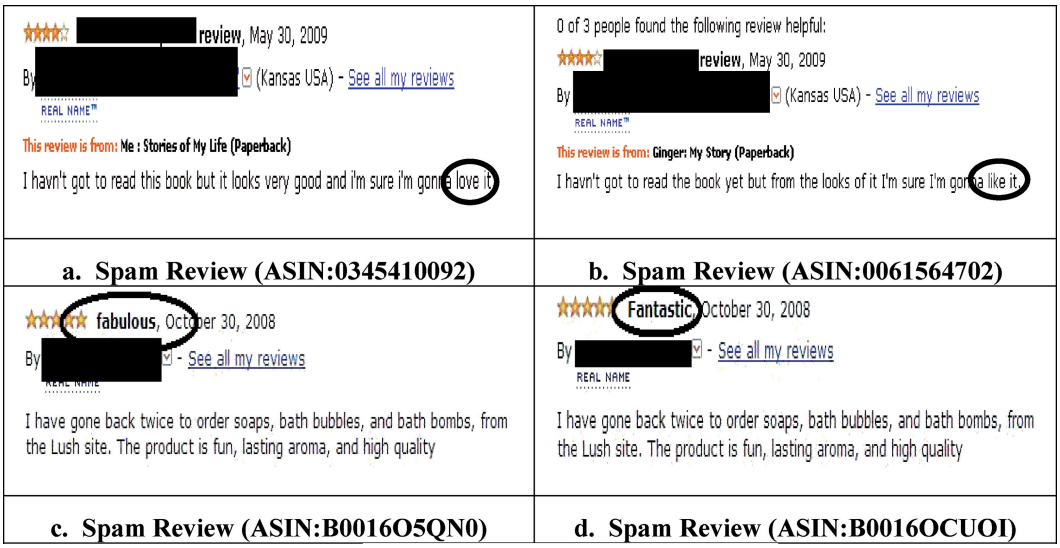

Fig. 5. Examples of obfuscation.

innovation of the proposed untruthful review detection method is that two apparently different untruthful reviews (e.g., many words are different) can be detected based on the semantic term relationships and the high-order concept association relationships.

The term "language model" is widely used among the speech recognition community to refer to a probability distribution M which represents the statistical regularities in the generation of a language [Nadas 1984]. In other words, a language model is a probabilistic function that assigns a probability to a string t drawn from some vocabulary T. Language modeling has been applied to estimate the relevance of a document d with respect to a query q in the field of Information Retrieval (IR) [Liu and Croft 2004; Ponte and Croft 1998]. Moreover, language modeling approaches have been successfully applied to Web spam detection [Martinez-Romo and Araujo 2009], blog spam detection [Mishne et al. 2005], and opinion mining [Lau et al. 2009c]. However, the aforementioned language modeling approaches do not take into account the term relationships when estimating the document generation probability. The basic unigram language model [Liu and Croft 2004; Ponte and Croft 1998] is defined by

$$P(q|d) \propto P(q|M_d) = \prod_{t_i \in q} P(t_i|M_d) \tag{1}$$

$$P(t_i|M_d) = (1-\lambda)P_{ML}(t_i|M_d) + \lambda P_{ML}(t_i|M_D) \tag{2}$$

$$P_{ML}(t_i|M_D) = \frac{tf(t_i, D)}{|D|}. \tag{3}$$

In Eq. (1), the term $P(q|d)$ represents the likelihood that a document d is relevant with respect to the query q, and the "relevance" is approximated by the probability that the document language model M_d "generates" the query q, that is, $P(q|M_d)$. This generation probability turns out to be the product of the probabilities of M_d generating the individual term t_i of the query q, that is, $P(t_i|M_d)$. One important element of the language modeling approach is the "smoothing" of the term probability [Zhai and Lafferty 2004]. The main intuition is that if a query term t_i is not found in document d, it does not necessarily mean that the document is not about t_i because semantically similar

terms such as synonyms could have been used to compose the document. Accordingly, the objective of smoothing the document model is to reduce the chance of overestimating the generation probability for terms observed in the document by applying a factor $(1-\lambda)$ to the maximum likelihood language model $P_{ML}(t_i|M_d)$. To alleviate the problem of underestimating the generation probability for terms not observed in the document, a factor λ is applied to the maximum likelihood estimation of t_i with respect to the entire collection D, that is, the collection language model $P_{ML}(t_i|M_D)$ defined in Eq. (2). The term λ is called the Jelinek-Mercer smoothing parameter, which usually assumes values in the range of [0.1, 0.7] [Nie et al. 2006; Zhai and Lafferty 2004]. Eq. (3) defines the Jelinek-Mercer smoothing process, where the probability of an unobserved term t_i is estimated according to its term frequency in the entire document collection D. In particular, $tf(t_i, D)$ represents the term frequency of t_i in the collection, and $|D|$ is the length (in words) of the entire document collection D. Similarly, the probability of $P_{ML}(t_i|M_d)$ is estimated according to $P_{ML}(t_i|M_d) = \frac{tf(t,d)}{|d|}$, where $|d|$ represents the document length.

To address spammers' obfuscation tactics [Abbasi et al. 2008], such as replacing the word "like" by "love," as shown in Figure 5, a better way to estimate the probability of an unseen term in a document (e.g., a review) is to take into account the relationship of ("love"→"like"), as the term "like" is a synonym of the term "love" according to WordNet [Miller et al. 1990]. The goal is to assign a more reasonable probability to an unseen term when evaluating two reviews such as those shown in Figures 5(a) and 5(b). This can be achieved using the novel semantic language model $P_{SEM}(t_i|M_d)$ defined according to Eq. (4). We have

$$\begin{aligned} P_{SEM}(t_i|M_d) &= \frac{\sum_{t_i,t_j \in R} P(t_i|t_j) P_{ML}(t_j|M_d)}{|R|} \\ &= \frac{\sum_{t_i,t_j \in R} P(t_j \rightarrow t_i) P_{ML}(t_j|M_d)}{|R|}, \end{aligned} \tag{4}$$

where $P(t_j \rightarrow t_i)$ is the certainty of the term association relationship between t_i and t_j, and is derived using our text mining method. The basic intuition of Eq. (4) is that if a term such as "like" (t_i) is not found in a document (review), but the term "love" (t_j) is found in the document, and a term association such as "love" → "like" is established (according to WordNet or high-order concept mining), then the generation probability of P_{ML}("love"$|M_d$) can be used to estimate P_{ML}("like"$|M_d$). In Eq. (4) the term R represents the set of term relationships in the form of $t_j \rightarrow t_i$, and $|R|$ is the cardinality of the set R. As a number of term association relationships may be discovered via text mining, only the top n associations ranked by $P(t_i|t_j)$ for each term t_i are considered. For the synonyms extracted from WordNet, $P(t_i|t_j) = 1.0$ is assumed because these relations are defined by human experts. Using $P_{SEM}(t_i|M_d)$ to smooth the language model further alleviates the problem of underestimating the unseen terms in a document. The translation language model [Berger and Lafferty 1999] and the inferential language model [Nie et al. 2006] have also been proposed to utilize term relationships to smooth language models. However, our proposed computational method is different from the aforementioned language models. In particular, our semantic language model can take into account the dynamic concept associations discovered via the text mining method.

As there may be unseen terms not captured in the term relationship set R, the collection language model is still required to smooth the overall language model. Therefore, our semantic language model for review spam detection is defined according to Eq. (5). First, a Jelinek-Mercer smoothing parameter ν is applied to smooth the unseen terms

in a document, as these unseen terms may be related to some other terms captured in the term relationship set R. Then, a second Jelinek-Mercer smoothing parameter μ is applied to conduct further smoothing for the unseen terms not captured by any term relationships.

$$\begin{aligned} P(t_i|M_d) \;=\; & (1-\mu)((1-\nu)P_{ML}(t_i|M_d) + \nu P_{SEM}(t_i|M_d)) \\ & + \mu P_{ML}(t_i|M_D) \end{aligned} \tag{5}$$

KL divergence [Kullback and Leibler 1951] is a well-known measure commonly used to estimate the distance between two probability distributions, and has been successfully applied to Web spam detection [Martinez-Romo and Araujo 2009; Mishne et al. 2005]. Accordingly, we apply a negative KL divergence to measure the similarity between pairs of language models, such as M_{d_1} and M_{d_2}, representing two reviews. If the negative KL divergence of the two language models is large, then the distance of the corresponding probability distributions is considered small. In other words, the semantic contents of the pair of reviews are quite similar, and they are likely to be spam. In fact, the negative KL divergence measure has also been applied to estimate the similarity between a query q and documents d in IR [Lafferty and Zhai 2001]. The negative KL divergence measure can be seen as a kind of normalization applied to the term generation probabilities derived by our semantic language models. The final formulation of the untruthful review detection method underpinned by LM and KL divergence is defined by

$$\begin{aligned} Score_{KL}(d_1, d_2) \;=\; & -KL(M_{d_1}||M_{d_2}) \\ =\; & -\sum_{t_i \in \{d_1 \cup d_2\}} P(t_i|M_{d_1}) \times \log_2 \frac{P(t_i|M_{d_1})}{P(t_i|M_{d_2})}, \end{aligned} \tag{6}$$

where t_i is a term appearing in either d_1 or d_2. The proposed method requires only one KL computation, and d_1 is assumed to be the longer review for each pair. The computational complexity of the proposed untruthful review detection method is characterized by $O(|N|^2)$, where N is the set of reviews for detection. In other words, the method has the quadratic time complexity, which belongs to the broader class of polynomial time complexity. This class of computational problems is generally considered tractable and efficient [Papadimitriou 1994].

4.2. The Computational Model for Non-Review Detection

Support vector machines have been successfully applied to text categorization tasks and have been shown to perform better than probabilistic classifiers [Joachim 1998]. For the two-class classification problem (e.g., spam versus ham), the basic principle is to find a hyperplane represented by the weight vector $\vec{\omega}$ (a normal vector perpendicular to the hyperplane), which not only separates the review vectors into two classes, but also guarantees that the separation, or margin, is as large as possible. This search corresponds to a constrained optimization problem with the form: $\vec{\omega} = \sum \alpha_i c_i \vec{d}_i, \forall \alpha_i \geq 0$ and $b = c_i - \vec{\omega}^{\mathrm{T}} \vec{d}_i, \forall \vec{d}_i : \alpha_i \neq 0$, where each $\vec{d}_i$ represents a labeled review, and $c_i \in \{1, -1\}$ denotes a class of this binary classification problem. The term α_i is the Lagrange multiplier, with each nonzero α_i indicating that the corresponding $\vec{d}_i$ is a support vector; and b is the intercept of the binary classifier. Based on the parameters, such as $\vec{\omega}$, α_i, and b, learned from the training data, the large margin classification function $f(\vec{d})$ can then be defined, where the signum function (sgn) is used to determine the class of a given test object $\vec{d}$ (i.e., an unlabeled review). For instance, a positive margin

value suggests a spam class, and a negative margin value indicates a ham class.

$$f(\vec{d}) = \text{sgn}\left(\sum_i \alpha_i c_i \vec{d}_i^{\mathrm{T}} \vec{d} + b\right) \tag{7}$$

We use Joachim's [1999] SVM package[6] for the classifier training and testing. All of the default parameter values (e.g., a linear kernel function for the reason of computational efficiency) of the SVM package are adopted. Syntactical, lexical, and stylistic features are identified and applied to detect non-reviews based on the SVM-based classifier. These features have been shown effective in detecting Web spam in general [Piskorski et al. 2008]. For syntactical features, we compute the percentage of nouns (f1), verbs (f2), and pronouns (f3) presented in a review based on our WordNet-based POS tagger. In addition, the ratio of modal verbs to the total number of verbs (f4) in a review is used. Lexical features include lexical validity (f5), text orientation (f6), sentiment orientation (f7), lexical diversity (f8), content diversity (f9), POS n-grams diversity (f10), and lexical entropy (f11). Lexical validity is the ratio of the number of valid words to the number of all words in a review, with valid words defined with respect to WordNet. Text orientation is the ratio of the number of potential words to the number of all words in a review, where a potential word is a token that does not contain numbers, special characters, or non-letter symbols. Lexical diversity is the ratio of the number of unique words to the number of all words. Content diversity is the ratio of the number of unique nouns and unique verbs to the number of all nouns and verbs presented in a review. Sentiment orientation is the ratio of the number of sentiment indicators to the number of all tokens in a review, where sentiment is indicated according to the opinion lexicon OpinionFinder [Riloff et al. 2005]. To make a trade-off between effectiveness and computational efficiency, we only consider word bigrams for computing f10 and f11. POS bigrams have the form (noun, verb), (adjective, noun), (adverb, adjective), etc. POS bigram diversity (f10) is the ratio of the number of different POS bigrams to the total number of POS bigrams. Let $B = \{b_1, \ldots, b_n\}$ be the set of all POS bigrams in a review, and let p_{b_i} be the distribution of a POS bigram $b_i \in B$. Lexical entropy (f11) is defined by $\text{lexicalentropy} = -\sum_{i=1}^{n} p_{b_i} \times \log_2 p_{b_i}$.

Stylistic features include capitalized diversity (f12), repetition diversity (f13), emotiveness diversity (f14), passive voice frequency (f15), and self-reference diversity (f16). Capitalized diversity is the ratio of the number of tokens starting with capital letters to the total number of tokens in a review. Repetition diversity is the ratio of repeated tokens to the total number of tokens. Emotiveness diversity is the ratio of the number of adjectives and adverbs to the number of all nouns and verbs in a review. Passive voice frequency is the ratio of the number of passive verb constructions to the number of all verbs. Self-reference diversity is the ratio of the number of first person pronouns to the number of all pronouns. A total of sixteen features are used and all the feature values are subject to linear normalization. Figure 6 depicts an example of a non-review (e.g., a commercial advertisement for an online gambling site) detected by our prototype system. This review mainly tries to lure the readers to visit the particular gambling site. This non-review was left undetected at Amazon for years!

5. MINING HIGH-ORDER CONCEPT ASSOCIATIONS

In this article, we refer to terms with general meanings (usually with multiple senses) as high-order concepts. A concept is a well-defined linguistic unit with well-defined semantics [Lau et al. 2009a, 2008]. Using the classical apriori-based association rule mining approach [Agrawal and Srikant 1994], the association between "fabulous" and

[6]http://svmlight.joachims.org/.


1 of 1 people found the following review helpful:

★★★★★ **http://www.gonegambling.com?referee=daju**, September 6, 2004

By **daju "daju"**

This review is from: Casino Gambling Free Casino Bonus Game Portal (Website)

Your one stop shop for the ultimate in entertainment.

http://www.gonegambling.com?referee=daju

There is all those quick ways to make money, "hogwash" !!! Try to gamble, I did and won a progressive $12,000.oo!!!
My goodness it is the most crazy scheme this gambling thing. If you can, control yourself in the heat of the moment. This is gonna be an addictive challange.
You can find advice, knowledge and my gosh, "tons of free money" , I promise you will be surprised as an ox getting pissed on by the lead ox. As the following is the ultimate gambling protal:

http://www.gonegambling.com?referee=daju

Fig. 6. An example of nonreview from amazon.com.

"fantastic" as shown in Figure 5(c) and 5(d) may never be discovered because the two terms do not appear in the same transaction (i.e., document). One of the main contributions of this article is the development of a text mining method for discovering high-order concept associations like "fabulous"→"fantastic" for review spam detection. The proposed high-order concept discovery method is underpinned by the context-sensitive text mining approach [Lau 2003; Lau et al. 2008]. The method consists of two main phases: concept extraction and association extraction.

5.1. Concept Extraction

After standard document preprocessing, such as stop-word removal, POS tagging, and word stemming [Salton and McGill 1983], a windowing process is conducted over the collection of reviews. The proximity factor imposed by the windowing process is the key to reduce the number of noisy term relationships. For each review, a virtual window of δ words is moved from left to right one word at a time until the end of a sentence is reached. Within each window, the statistical information among tokens (words) is collected to develop collocational expressions [Perrin and Petry 2003]. The windowing process is repeated for each document until the entire collection has been processed. According to previous studies, a text window of between 5 and 10 terms is effective [Lau 2003; Lau et al. 2008; Perrin and Petry 2003]. Therefore, we adopt this window size as the basis for the windowing process. To improve computational efficiency and filter noisy relations, only the concepts with certain linguistic patterns (e.g., "Noun," "Adjective," "Noun Noun," "Adjective Noun," etc.) are extracted.

Mutual Information (MI), which has been used in collocational analysis [Perrin and Petry 2003] in previous research, is adopted as the basic computational method for the statistical token analysis. Mutual Information is an information-theoretic method for computing the dependency between two entities and is defined by [Shannon 1948]

$$MI(t_i, t_j) = \log_2 \frac{\Pr(t_i, t_j)}{\Pr(t_i \Pr(t_j)}, \tag{8}$$

where $MI(t_i, t_j)$ is the mutual information between term t_i and term t_j. $\Pr(t_i, t_j)$ is the joint probability that both terms appear in a text window, and $\Pr(t_i)$ is the probability that the term t_i appears in a text window. The probability $\Pr(t_i)$ is estimated based on $\frac{w_t}{w}$, where w_t is the number of windows containing the term t, and $|w|$ is the total number of windows constructed from the collection. Similarly, $\Pr(t_i, t_j)$ is the fraction of the number of windows containing both terms out of the total number of windows.

To consider counter-evidence, such as term t_i and term t_j do not appear in the same text window, a Balanced Mutual Information (BMI) measure is developed [Lau 2003; Lau et al. 2009a]. The BMI measure considers both term presence and term absence as evidence for estimating the strength of the association between a concept and its underlying descriptive terms. We have

$$\begin{aligned} Ass(t_i, t_j) &= BMI(t_i, t_j) \\ &= \alpha \times \Pr(t_i, t_j) \log_2 \left(\frac{\Pr(t_i, t_j)}{\Pr(t_i)\Pr(t_j)} + 1 \right) \\ &\quad + (1 - \alpha) \times \Pr(\neg t_i, \neg t_j) \log_2 \left(\frac{\Pr(\neg t_i, \neg t_j)}{\Pr(\neg t_i)\Pr(\neg t_j)} + 1 \right), \end{aligned} \tag{9}$$

where $Ass(t_i, t_j)$ is the association weight between two tokens (e.g., one concept and one descriptive term). $\Pr(t_i, t_j)$ is the joint probability that both terms appear in a text window, and $\Pr(\neg t_i, \neg t_j)$ is the joint probability that both terms are absent in a text window. The factor $\alpha > 0.5$ is used to tune the relative weight of the positive and negative evidence, respectively. As it is counterintuitive to have a zero BMI value if two terms always appear together in every text window, the faction $\frac{\Pr(t_i, t_j)}{\Pr(t_i)\Pr(t_j)}$ is adjusted by adding the constant 1 before applying the logarithm. As concept extraction is applied after removing stop-words, only significant token associations are considered. In Eq. (9), each BMI value is then normalized by the corresponding joint probabilities. Only terms with an association weight greater than a threshold η (i.e., $Ass(t_i, t_j) > \eta$) are considered significant terms and included in a concept vector. Once all the BMI values in a collection are computed, the values are then subject to linear scaling such that each term association weight falls within the unit interval. It should be noted that the constituent terms of a concept are always implicitly included in the underlying concept vector with a default association weight of 1. By applying the concept extraction procedure to a review collection, we can discover high-order concepts and represent them as concept vectors. The following are examples of concept vectors for "fabulous" and "fantastic" (for readability, the original stemmed version is converted to a nonstemmed format). These concept vectors are discovered based on our collection of reviews downloaded from amazon.com.

fabulous = <(nice 0.91783), (work 0.97112), (amazing 0.96894), (always 0.96722), (computer 0.96525), (incredible 0.96489), (ummhow 0.96113), ... >

fantastic = <(amazing 0.98312), (nice 0.98217), (work 0.97113), (fun 0.97005), (router 0.96726), (ok 0.96523), (always 0.95198), ... >

5.2. High-Order Concept Association Extraction

The final stage of the high-order concept association mining method is the extraction of the association relations among concepts based on the notion of "subsumption" [Sanderson and Croft 1999]. Let $Spec(c_x, c_y)$ denote that concept c_x is a specialization [subclass) of another concept c_y. The degree of this subsumption relation is derived by

$$Spec(c_x, c_y) = \frac{\sum_{tx \in cx, ty \in cy, tx = ty} Ass(t_x, c_x) \otimes Ass(t_y, c_y)}{\sum_{tx \in cx} Ass(t_x, c_x)} \tag{10}$$

where $\otimes$ is a standard fuzzy conjunction operator which is equivalent to a minimum function. The preceding formula states that the degree of subsumption (specificity) of c_x to c_y is based on the ratio of the sum of the minimal association weights of

Table I. The Amazon Evaluation Dataset

Product Category	Amazon Browse Node#	# of Reviews Downloaded	# of Untruthful Reviews	# of Artificial Non-Reviews	# of Legitimate Reviews	Size of Evaluation Dataset
Automotive	15690151	108,423	204	102	5,100	5,406
Beauty	11055981	97,663	187	94	4,675	4,956
Grocery	3760931	122,474	220	110	5,500	5,830
Cameras	281052	126,238	216	108	5,400	5,724
Computers	541966	515,182	220	110	5,500	5,830
Books	9 and 86	481,291	201	101	5,025	5,327
DVD	163416	173,674	204	102	5,100	5,406
Music	34	229,154	212	106	5,300	5,618
Software	409488	58,599	192	96	4,800	5,088
Video Games	493964	406,291	205	103	5,125	5,433
Total		2,318,989	2,061	1,032	51,525	54,618

the common terms of the two concepts to the sum of the term weights of concept c_x. According to Eq. (10) and the concept vectors depicted in Section 5.1, high-order concept associations such as ("fabulous"→"fantastic") can be discovered even if these concepts do not co-occur in the same transaction (review).

6. EXPERIMENTS AND RESULTS

We adopted the evaluation approach and the common effectiveness measures, such as *lam%* and (*1-AUC*) used by the TREC Spam Track [Cormack 2007], to evaluate the effectiveness of our review spam detection models. As the TREC Spam Track evaluation dataset only contained email messages, we built our evaluation dataset based on reviews downloaded from amazon.com in January 2010. More specifically, we utilized the Amazon Web services (October-2009 version) to extract reviews from ten Amazon product categories. The details of this review collection are shown in Table I. A subset of this collection (3,093 spam and 51,525 ham) was then used to evaluate our proposed models. It should be noted that some reviews were duplicated by Amazon's internal categorization procedure. More specifically, for different versions of the same or similar products (hardcopy versus paperback books), the Amazon review catalog system will copy all of the reviews of one version of a product (with a unique ASIN) to the second version (with another unique ASIN) of the same product. We applied a heuristic to remove the duplicated reviews created by Amazon's internal cataloging service before adding the reviews to our evaluation dataset. For example, if the difference in the total length of the reviews (in characters) of two products is less than a tolerance percentage (e.g., 1% of the review length), the two products will be considered to be two versions of the same product. Accordingly, only one of the two products and its reviews will be included in the evaluation dataset.

6.1. The Evaluation Dataset

Building an evaluation dataset to assess the performance of a review spam detection system is a challenging task, given the huge volume of online consumer reviews and the lack of explicit features for human readers to identify fake reviews [Lim et al. 2010; Jindal and Liu 2008]. Accordingly, we used a semiautomatic method to build our evaluation dataset [Jindal and Liu 2008]. Figure 7 highlights the procedure used to create our evaluation dataset.

After downloading reviews from Amazon, we first removed the duplicated reviews created by Amazon's cataloging procedure. Each review of a product from our selection of Amazon product categories was represented by a vector of term weights. In particular, we used the TFIDF vector representation [Salton and McGill 1983]. The cosine scores of each review vector against all the other review vectors of the same

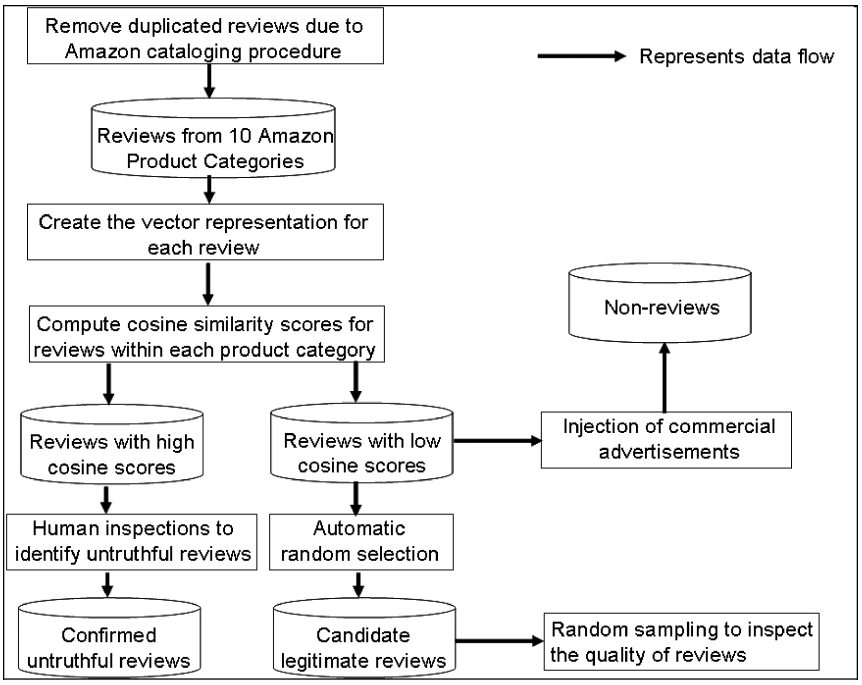

Fig. 7. The procedure of test data creation.

product categories were computed using the cosine similarity measure [Salton and McGill 1983]. If the cosine score of a pair of reviews was greater than or equal to a predefined threshold value (e.g., 0.83), the reviews were added to a candidate spam set for the corresponding product category. The suspicious pairs of reviews from the candidate spam set were then randomly selected for manual inspection. For each product category, two human annotators (two undergraduate students with over three years' review posting/browsing experience) were appointed to review the candidate spam set. If both human annotators confirmed a case as spam, the pair of reviews were added to the confirmed set of untruthful reviews. For the experiment reported in this article, only untruthful reviews with at least one or more word substitution were included in our untruthful review evaluation dataset. Moreover, our annotators verified that these untruthful reviews could not also be classed as non-reviews. Overall, the two annotators demonstrated high inter-rater agreement and achieved an average kappa value of $\kappa = 0.81$. For the extraction of legitimate reviews, the cosine similarity measure was again used. If the cosine scores of a review were all below a threshold value (e.g., 0.25), the review was considered a candidate of ham. Similar to previous studies [Lim et al. 2010; Jindal and Liu 2008], our annotators did not attempt to verify every review that was considered a candidate of ham given the huge number found in a collection. On the other hand, to build the evaluation dataset for non-review detection, we extracted a subset of reviews with low similarity scores and then artificially injected typical commercial advertisements in their place. These reviews became the test cases for the positive class. The artificial non-reviews were created to ensure that there was a certain percentage of non-reviews in the test dataset to facilitate the subsequent experimentation. For the test cases for detecting the negative class of non-reviews, we simply used the set of candidate legitimate reviews already extracted. Although the set of candidate legitimate reviews may have contained non-reviews or untruthful

Table II. A Confusion Matrix for the Evaluation of Spam Detection Methods

		Gold Standard – Human Classification	
		Spam	Ham
System's Classification	Spam	a	b
	Ham	c	d

reviews, our random sampling of 1% of these reviews showed that none of the sampled review was a non-review or untruthful review. This result from the random inspection suggests that our legitimate set has a relatively high quality. The details of our evaluation dataset are shown in Table I. For our evaluation dataset, the spam-ham ratios for untruthful reviews and non-reviews were set to 4% and 2%, respectively, to simulate a real-world skewed spam-ham distribution [Jinal and Liu 2008].

The cosine score thresholds for extracting the candidate spam and ham sets were set using an empirical testing approach. In particular, different thresholds were trialed, and the quality of the resulting sets was inspected based on random sampling. If the result of an inspection was good (e.g., none of the inspected reviews from the candidate ham set was untruthful), and yet a reasonable number of candidates were identified, then the corresponding cosine score threshold was selected. We acknowledge that the thresholds adopted may not have been optimal. However, the search for better cosine score thresholds and the sensitivity analysis of the spam detection results of the various thresholds will be left as part of our future research. As we used the Vector Space model (VS) and the cosine similarity threshold [Salton and McGill 1983] to extract the candidate spam set, one may question the feasibility of using the VS model for untruthful review detection. Indeed, the VS model is one of the candidate models examined for untruthful review detection. The evaluation of the various methods of untruthful review detection is discussed in Section 6.3.

6.2. The Performance Measures

We employed the measures adopted in the TREC Spam Track [Cormack and Lynam 2005; Cormack 2007] to evaluate the performance of various review spam detection methods. Although these measures were originally used to assess the effectiveness of the spam email filters used in the TREC Spam Track, they have been widely used to evaluate other kinds of spam, such as Web spam [Martinez-Romo and Araujo 2009].

With reference to the confusion matrix depicted in Table II, the various effectiveness measures can be defined by

$$hm = \frac{b}{b+d}, \tag{11}$$

$$sm = \frac{c}{a+c}, \tag{12}$$

$$lam = \text{logit}^{-1}\left(\frac{\text{logit}(hm) + \text{logit}(sm)}{2}\right), \tag{13}$$

where a, b, c, and d refer to the number of reviews falling into each corresponding classification category. The ham misclassification rate (hm) is the fraction of all ham misclassified as spam, and the spam misclassification rate (sm) is the fraction of all spam misclassified as ham. There is a natural tension between the ham and spam misclassification rates, as a spam detection system will always improve the hm rate at the expense of the sm (e.g., by increasing the spam classification threshold t) and vice versa. It is therefore desirable to have a single measure that combines both of the

aforementioned measures. Accordingly, the TREC Spam Track employed the *logistic average misclassification rate* (*lam*) to measure the effectiveness of the spam detection systems. With reference to Eq. (13), the logit functions are defined by: $\text{logit}^{-1}(x) = \frac{e^x}{1+e^x}$ and $\text{logit}(x) = \ln\left(\frac{x}{1-x}\right)$. As *hm*, *sm*, and *lam* are measures of failure rather than effectiveness, a small ratio achieved by a detection system implies good performance. The ratio of true positive, $tp = \frac{a}{a+c}$, in contrast is the fraction of all spam identified by a detection system and is similar to the notion of recall in IR [Salton and McGill 1983]. The common effectiveness measure, accuracy $= \frac{a+d}{a+b+c+d}$, may not be a good instrument to assess the performance of spam detection systems. Given the skewed distribution of spam and ham (e.g., there is usually a large amount of ham and only a small amount of spam), a spam detection system may trivially classify all reviews as ham (i.e., category *d*) and yet achieve a relatively high degree of accuracy.

The Receiver Operating Characteristic (ROC) curve [Hand and Till 2001], which is the graphical representation of *tp* (i.e., $1 - sm$) as a function of *hm*, has also been used to evaluate spam detection systems [Cormack 2007; Cormack et al. 2007b]. The advantage of the ROC curve is that the evaluation (or comparison) of spam detection systems is independent of the choice of a particular threshold value [Hand and Till 2001]. Given the spamminess scores (i.e., the classification scores computed by a spam detection system) assigned to some test reviews, a detection system can rank all the reviews in descending order of spamminess scores. Based on such a ranking, it is possible to compute the corresponding *tp* and *hm* rates by hypothetically adjusting the classification threshold *t* along the ranking [Cormack 2007; Cormack et al. 2007b]. To be consistent with the *hm* and *sm* rates, which measure failure rather than effectiveness, the TREC Spam Track also employed the measure "Area Above the ROC Curve," that is, $(1 - AUC)$ to evaluate the spam detection systems [Cormack and Lynam 2005; Cormack 2007].

6.3. Evaluation of the Untruthful Review Detection Methods

For the evaluation of the proposed untruthful review detection method, we implemented a variety of detection models in our prototype system. More specifically, we compared the performance of the proposed Semantic Language Model (SLM) (i.e., Eqs. 1-6) with that of four baseline models, namely, the unigram language model LM (i.e., Eqs.(1)–(3)), the I-Match plagiarized document detection model [Chowdhury et al. 2002], the Vector Space Model (VS) [Salton and McGill 1983], and the Support Vector Machine (SVM) [Joachim 1998]. Except for the SVM method, all the baseline models can be regarded as unsupervised classification models. In the LM method, the shorter review is treated as a query whereas the longer review is treated as the document that generates the query. If the generation probability is high, then the pair of reviews is considered untruthful. The LM baseline method cannot take into account term substitutions resulting from spammers' obfuscation strategies. Our SVM method made use of the same set of features used in a previous untruthful review detection study [Jindal and Liu 2008], which comprised 21 content-based features, 11 reviewer-based features, and 4 product-based features. To count the percentages of positive and negative sentiment indicators, we utilized the OpinionFinder sentiment lexicon [Riloff et al. 2005]. In this experiment, the SVM method used a variant of the decision function defined in Eq. (7). In particular, the margin scores (i.e., spamminess scores) were directly used to rank the test cases. A video game subset of the evaluation data was used to train the SVM, and the values $\mu = 0.58$ and $\nu = 0.64$ were empirically established for the SLM model. According to the cross-validation strategy, the SLM parameters tuned based on the video games subset were then applied to the spam detection for the rest of the evaluation dataset [Mitchell 1997]. Applying heuristic search methods, such as genetic

Table III. Comparative Performance of Various Untruthful Review Detection Methods

Method	*tp*%	*hm*%	*sm*%	*lam*%	(1-*AUC*)%
SLM	**97.77%**	**1.30%**	**2.23%**	**1.70%**	**0.1346%**
LM	95.88%	2.80%	4.12%	3.40%	1.5921%
I-Match	95.92%	2.81%	4.08%	3.38%	1.5919%
VS	94.52%	6.65%	5.48%	6.04%	3.7965%
SVM	56.53%	40.00%	43.47%	41.73%	44.2875%

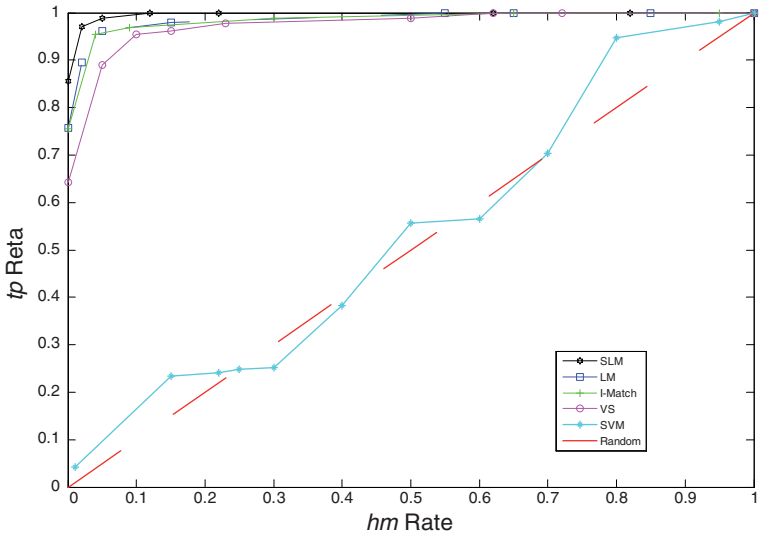

Fig. 8. The ROC curves of various untruthful review detection methods.

algorithms [Goldberg 1989; Lau et al. 2006], to the parameter tuning may bootstrap the performance of the SLM method. However, we will leave these more sophisticated forms of parameter tuning as part of our future work.

A summary of the comparative performance of the various untruthful review detection methods is depicted in Table III. The SLM method clearly has the lowest error rate in terms of *lam*% (1.70%) and (1-*AUC*)% (0.1346%). The SLM method also has the highest true positive percentage (e.g., 97.77%). According to the ROC curve analysis shown in Figure 8, the SLM method consistently performs better than all the other baseline methods at all possible threshold levels (e.g., the SLM ROC curve is above all the other ROC curves). The LM method is not as effective as the SLM method, even though both approaches are based on a unigram language modeling framework. The main reason for this is that the SLM method is able to take into account the substituted terms (resulting from spammers' obfuscation strategies) in untruthful reviews. By using the semantic relationships captured from WordNet and the high-order concept associations discovered via the proposed text mining method, the SLM method can detect a pair of untruthful reviews even if their wordings are not the same. The I-Match method achieves more or less the same performance as the LM method, as both methods lack the computational mechanisms needed to detect substituted words in untruthful reviews. The VS method is not as effective as the language modeling-based methods or the plagiarized content detection method, although all of these methods can be regarded as types of unsupervised classification methods. After completing our in-depth analysis of the detection results, it was found that the VS method performed very poorly for short reviews. For the short reviews, even if there was only one overlapping term between

two reviews, the cosine similarity score could be very high due to the high term weight assigned to the rare overlapping term. As a result, the VS detection method mistakenly classified a relatively large number of ham as spam. Finally, the results of the SVM method confirm our observation that it is difficult for supervised classifiers to correctly identify untruthful reviews, as the necessary discriminatory features may not be found inside the individual untruthful reviews and large numbers of training examples are not available to train the classifiers. As a result, the SVM method mistakenly classified many ham as spam and vice versa in our experiment. The performance of the SVM method is only slightly better than that of a random classifier.

For the second experiment, we used a number of untruthful reviews from each product category as seeding reviews to artificially construct duplicated reviews to test the performance of various methods under different levels of obfuscation. We used a computer program to automatically modify some of the words in the seeding reviews to generate the corresponding untruthful reviews. When word substitutions were performed to simulate spammers' obfuscation strategies, the set R of high-order concept associations discovered via text mining were explored first. If none of the words in a seeding review was found in the set R, the synonyms defined in WordNet would then be used to find replacement words. The spam-ham ratio was controlled so that it was more or less the same as in the first experiment. Various percentages of word substitutions (e.g., replacing 10%, 30%, 50%, or 70% of the words in a seeding review) were initiated to mimic different obfuscation scenarios. The performance of the various untruthful review detection methods under different obfuscation scenarios is highlighted by the ROC curves plotted in Figures 9(a), 9(b), 9(c), and (9d). When only a small number of words were replaced to produce untruthful reviews, the performance of the various detection methods was similar to that reported in the first experiment. However, when more words were replaced in the untruthful reviews, the difference in the performance of the SLM method and the other baseline methods (reflected in the distances between the ROC curves) became more obvious. The proposed SLM method always performs the best, as it can take into account the possible obfuscation measures taken by the spammers. The performance of the SLM method is outstanding across the various degrees of obfuscation. We believe that this characteristic of the SLM method makes it a good candidate for application in real-world review spam detection environments.

6.4. Evaluation of the Non-review Detection Methods

For the third experiment, we evaluated the performance of three supervised machine learning models, namely, SVM, K-Nearest Neighbor (KNN) [Mitchell 1997], and Logistic Regression (LR) [Jindal and Liu 2008] in detecting non-reviews. For the SVM-based method, we applied Joachim's SVM package and adopted all the default parameter values (e.g., a linear kernel function) [Joachim 1999]. The details are illustrated in Section 4.2. For the KNN method, we applied the nearest $K = 5$ neighbors of a test object. The weighted distance of the test review to the five nearest training reviews was used to generate the spamminess score. A spam was assigned the weight of 1 and a ham was given the weight of -1. For the LR-based method, we adopted a similar approach as that used by Jindal and Liu [2008]. All of the supervised machine learning methods utilized the same feature set, that is, the syntactical, lexical, and stylistic features illustrated in Section 4.2. The training set contained 70% of the data of the evaluation dataset, and the test set contained the remaining 30%. The same spam-ham ratio was applied to both the training set and the test set. The details of this non-review dataset are depicted in columns 5 and 6 of Table I. The comparative performance of the three methods is depicted in Table IV. As can be observed, both the lam percentage (5.04%) and the area above ROC curve percentage ($1- AUC\% = 2.5112\%$) of the SVM method are the lowest among all the three methods. The results of this experiment confirm that

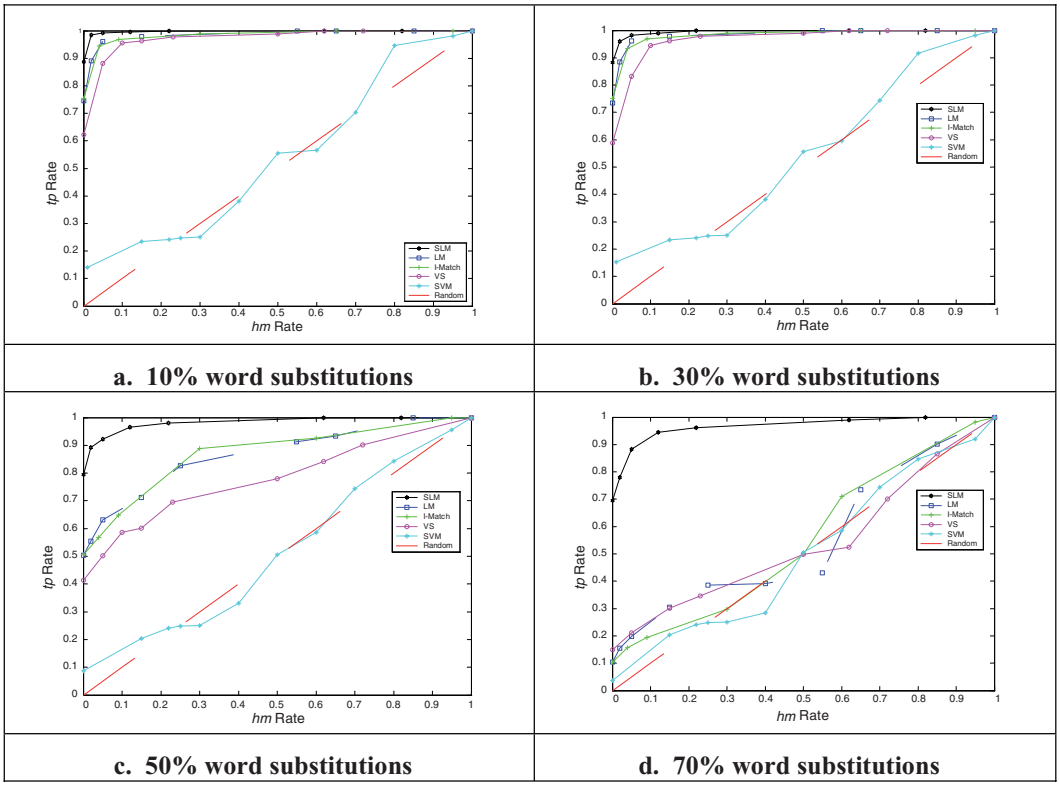

Fig. 9. Performance of the untruthful detection methods under varying degrees of obfuscation.

Table IV. Comparative Performance of Various Non-Review Detection Methods

Method	tp%	hm%	sm%	lam%	(1-AUC)%
SVM	95.06%	5.14%	4.94%	5.04%	2.5112%
LR	94.19%	6.38%	5.81%	6.09%	3.4176%
KNN	92.05%	8.35%	7.95%	8.15%	4.5953%

the proposed SVM-based model is an effective non-review detection method. The reason the SVM performs better than the other methods of non-review detection may be due to the fact the SVM is a large margin classifier, which enables it to effectively learn the set of feature weights that maximally separate the spam and the ham classes. As a result, fewer mistakes were produced by the SVM method. In contrast, although the KNN and LR methods can also learn sets of feature weights to identify the spam cases, these sets of weights may not maximally separate the two classes. As a result, in confusing cases, the KNN and LR methods are both likely to predict the wrong class. The SVM method performed much better in non-review detection because discriminatory features were available inside individual non-reviews to train the SVM classifier.

6.5. Evaluation of the Detection Methods Based on a Mixed Dataset

To simulate a realistic setting where untruthful reviews and non-reviews are mixed together in a single data source, we also conducted additional experiments which were essentially reruns of the first and third experiments. In these experiments, the experimental conditions were exactly the same as before except for the input data.

The input data are a combination of untruthful reviews, non-reviews, and legitimate reviews, as depicted in columns 4, 5, and 6 of Table I. Our experimental results showed that the SVM and SLM modules achieved the same classification results as reported in Tables III and IV (i.e., the best performance). For brevity, we do not reproduce the experimental results here. These two modules achieved the same detection performance as reported earlier because they were configured to detect a specific type of spam only. The modules simply ignored the other kind of spam included in the test dataset. For instance, the SVM module was trained to only classify non-reviews based on a training set containing examples of non-reviews and simply treated untruthful reviews as legitimate reviews. In this case, the additional 2,061 untruthful reviews from the input source did not cause any misclassification by the SVM module. A similar situation was applied to the SLM detection module. As a result, we did not observe any degradation or improvement in the classification performance of the respective modules.

In terms of the efficiency of the proposed computational models, it took 58 minutes and 11 seconds (i.e., 0.064 seconds per review) for the SLM to complete the untruthful review detection task, and 14 minutes and 24 seconds (i.e., 0.016 seconds per review) for the SVM to complete the non-review detection task. Both of the proposed models achieved high true positive percentages and low misclassification percentages. Our prototype system is able to detect different types of fake reviews (e.g., untruthful reviews and non-reviews) and manage the uncertainty arising in the detection process by producing a probabilistic ranking of suspicious fake reviews. Moreover, the proposed SLM model is able to infer spammers' obfuscation strategies by examining the associated concepts embedded in suspicious untruthful reviews. Therefore, our prototype system satisfies all of the requirements of an effective review spam detection system.

6.6. Discussion of Our Design Process

In design science research, the design process and the design artifacts go hand-in-hand [Hevner et al. 2004; March and Storey 2008; Peffers et al. 2008]. Accordingly, our design process and experience may be beneficial to other researchers engaging in similar work. In accordance with the design science research methodology [Hevner et al. 2004; Peffers et al. 2008], our research began with the identification of a relevant and important business problem, that is, the increasingly serious review spam problem [Dellarocas 2003, 2006; Lim et al. 2010]. Given that existing technologies are unable to provide a solution to this problem, as evidenced by the numerous cases of fake reviews reported in the press or brought to attention in legal circles, our aim is to build novel artifacts capable of automatically detecting fake reviews posted on the Internet. Before building these innovative artifacts (e.g., the computational model for untruthful review detection), we conducted an extensive literature review of the related research areas to develop insights into the theoretical foundations and potential techniques that can be employed to develop the necessary construct, model, method, and instantiation. Moreover, our extensive literature review revealed that the evaluation of design artifacts could be a challenging task, as the standard benchmark dataset was not available. In fact, construction and evaluation are the two major design processes in design science research [Hevner et al. 2004; Peffers et al. 2008]. During the initial phase of the building process, we examined feasible ways of evaluating the artifacts and building a test dataset. Designing the evaluation procedure and constructing the test dataset early on was a valuable experience that other researchers in similar fields should take into account. It takes a great deal of time and tremendous effort to compose a test dataset of reasonable quality. Unfortunately, without an evaluation procedure and test dataset in place, it is impossible to assess one's design alternatives!

We began the design process of building the artifacts by examining a number of existing methods, such as supervised machine learning techniques [Chen et al. 2009;

Cormack et al. 2007a; Jindal and Liu 2008] and heuristic methods [Lim et al. 2010], to assess whether they could be adapted and applied to review spam detection. We used an iterative construction and evaluation process to examine different design alternatives. Surprisingly, the well-known supervised machine learning techniques did not produce satisfactory results for the detection of untruthful reviews. Initially, we thought that there may have been errors in our experimental artifacts or even our experimental procedures. Hence, we carefully re-examined each step of the experimental procedure and inspected the source codes of the implemented computer programs. After a few more rounds of experimentation and evaluation, we decided that these techniques might not be appropriate for the detection of untruthful reviews. The failure of our early attempts in artifact construction led us to re-examine the distinctive characteristics of the problem (e.g., the characteristics of untruthful reviews). This analysis led us to focus on unsupervised techniques for detecting semantic content overlapping among untruthful reviews. Our lesson learned is that early failure in the construction of artifacts may not be such a bad thing, as it can help develop deeper insights into the problem domain and eventually lead to the construction of the desired artifacts. Moreover, an iterative process of construction and evaluation of design alternatives is very useful in facilitating the discovery of a good design solution. We acknowledge that our current design artifacts and the evaluation process we deployed may not be optimal. However, we are confident that the iterative construction and evaluation process (i.e., the core design science research process) will enable us to improve the effectiveness of the proposed artifacts in future research.

6.7. Limitations of Our Study

As the detection of untruthful reviews was conducted for each individual product category, some untruthful reviews across different product categories might not be detected. Future research will examine the percentage of untruthful reviews that occur in multiple product categories. Furthermore, our proposed method may not be able to detect untruthful reviews in cases where the review collections contain only a single untruthful review. Nevertheless, as spamming behavior tends to be repeated behavior, as found in a previous study [Abbasi et al. 2008], the proposed method should be able to detect most of the untruthful reviews that involve multiple copies of similar semantic contents. Our evaluation dataset was composed based on the spam found in reviews with high cosine similarity scores and the ham found in reviews with low cosine similarity scores. Therefore, the choices of the particular cosine similarity thresholds for the test case selection may have affected the detection results reported in this section. Moreover, since not every candidate legitimate review was inspected by the human annotators, there was a chance that some of these legitimate reviews was actually spam. Nonetheless, our random inspection of the candidate legitimate reviews in the evaluation dataset indicated that all the inspected reviews were the legitimate ones. Accordingly, we believe that our evaluation dataset has relatively good quality. Finally, the proposed SLM detection method resulted in 1.3% of the legitimate reviews being misclassified as spam. Further improvement of the accuracy of the SLM method will be part of our future research.

7. APPLICATION OF THE DESIGN ARTIFACTS

Many researchers have expressed concern about the trustworthiness of online consumer reviews [Cheung et al. 2009; Dellarocas 2003, 2006]. Nonetheless, few empirical studies have assessed the trustworthiness of consumer reviews posted to the Internet. Whether online consumer reviews are trustworthy or not is an important question that firms and individual consumers need to address before using these reviews to make marketing or purchasing decisions. In our empirical study of online consumer reviews,

Table V. The Spam Rate for Online Consumer Reviews

Product Category	# of Reviews Downloaded	# of Untruthful Reviews	# of Non-Reviews	% of Spam
Automotive	108,423	1,584	35	1.49%
Beauty	97,663	3,247	58	3.38%
Grocery	122,474	3,604	56	2.99%
Cameras	126,238	886	31	0.73%
Computers	515,182	10,901	114	2.14%
Books	481,291	7,836	85	1.65%
DVD	173,674	3,237	43	1.89%
Music	229,154	3,248	37	1.43%
Software	58,599	1,802	51	3.16%
Video Games	406,291	4,961	36	1.23%
Average		4,131	55	2.01%

random samples of reviews were downloaded from amazon.com, the details of which are shown in columns 1, 2, and 3 of Table I. We were able to estimate the trustworthiness of these online consumer reviews by applying our SLM and SVM computational models to detect the untruthful reviews and non-reviews within this dataset. Our empirical findings are highlighted in Table V. It should be noted that the numbers of untruthful reviews and non-reviews depicted in Table V represent all of the fake reviews detected by our prototype system based on the sample of downloaded reviews, whereas similar figures shown in Table I represent the fake reviews detected based on the evaluation dataset alone.

The average spam rate (including untruthful reviews and non-reviews) is 2.01%, based on the sample of 2,318,989 reviews randomly downloaded from amazon.com. Although the spam rate for online consumer reviews is lower than the average spam rate of 10–15% on the Web [Gyöngyi and Garcia-Molina 2005], the problem of review spam should not be underestimated given the large number of fake reviews. This is because some commercial sites, such as amazon.com, show the overall review details in reverse chronological order. If a considerable amount of spam has been recently generated, then the reviews shown to the reader on the first review page will mostly be fake. Accordingly, consumers' purchase decisions may be heavily influenced by the fake reviews. As a whole, consumers and firms should be very careful in making purchasing or business decisions based solely on online consumer reviews. According to our empirical study, untruthful reviews rather than non-reviews appear to be the main source of spam. This may be explained by our observation that certain explicit and discriminatory features are often available to distinguish non-reviews (e.g., commercial advertisements) from legitimate reviews. Therefore, it is relatively easy for a Web site to employ a manual or automatic process to detect and remove non-reviews. However, the actual business implications of fake reviews for firms' marketing strategies and sales performance require further study.

Our design artifacts enable firms or individual consumers to analyze the specific spamming behavior related to online consumer reviews. Figure 10 depicts the relationship between the total number of reviews and the number of fake reviews (untruthful reviews and non-reviews) attached to a product for the product categories of "Computers" and "Books" (mainly history books and entertainment books) cataloged at Amazon. Apparently, no direct linear relationship exists between the number of reviews of a product and its fake reviews. Both non-popular products (few people commenting on them) and popular products (many people commenting on them) could attract many fake reviews according to our spam analysis. Since the limitation of the particular version of Amazon Web service we utilized for this study, we could only download a maximum of 1,005 reviews for an arbitrary product. The popular products may even have a larger volume of reviews not shown in our scatter diagrams. Nevertheless, our

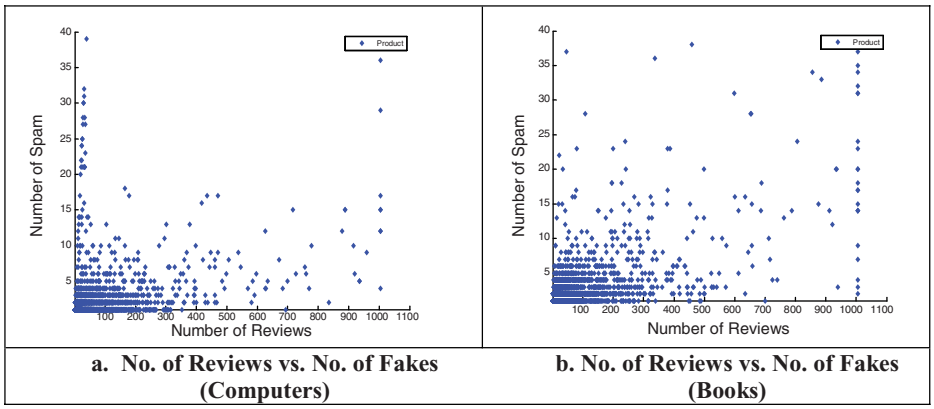

Fig. 10. The relationship between the number of reviews and number of fakes.

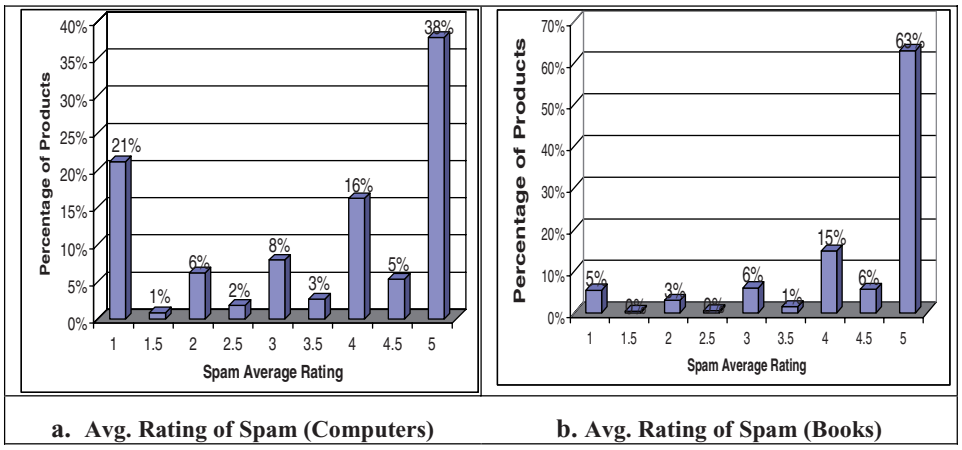

Fig. 11. The average ratings of fake reviews attached to products.

diagrams should be able to highlight the general spam patterns. As the other product categories of Amazon also reveal similar spam patterns, we mainly use the categories of "Computers" and "Books" as examples to illustrate the general patterns of fake reviews in this section.

Figure 11 shows the average ratings of the fake reviews (i.e., untruthful reviews and nonreviews) attached to products in the "Computers" and "Books" product categories, respectively. When fake reviews are attached to products, the majority appears to have an average review rating of 5 (38% for "Computers" and 63% for "Books"). In other words, the use of spam to promote a product is quite common. The comparatively positive fake reviews observed in our study are consistent with the findings of a previous study, that the review ratings at the Amazon site are generally skewed toward the positive side [Danescu-Niculescu-Mizil et al. 2009]. However, for the category of "Computers," quite a number of products (28%) have negative fake reviews (i.e., fake reviews with ratings between 1 and 2). This suggests that fake reviews could be used to defame products as well as promote products in certain product categories. Our empirical findings confirm the observations and predictions discussed in the existing literature, that is, fake reviews can be used to promote a firm's own products or damage the reputation of their competitor's products [Dellarocas 2003, 2006].

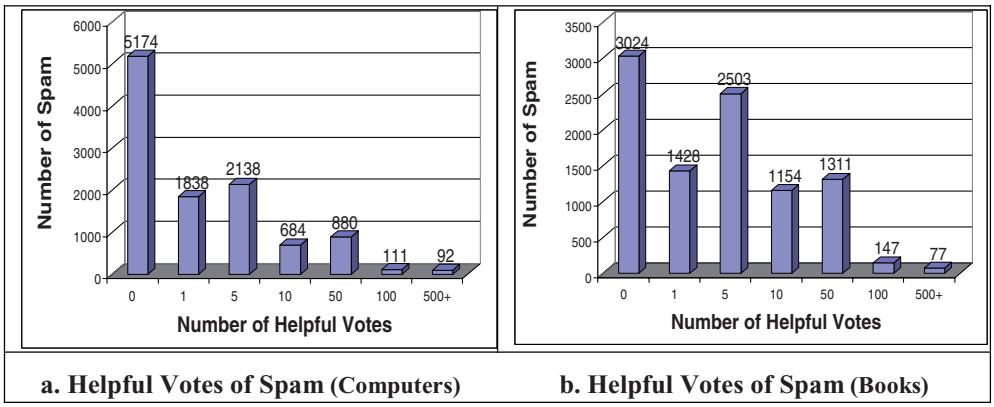

Fig. 12. Number of helpful votes in fake reviews.

Figure 12 depicts the relationship between helpful votes and fake reviews. For the "Computers" category, it is shown that a relatively large number of fake reviews receive zero helpful votes, because fake reviews may not help consumers to evaluate true product quality. However, fake reviews may also attract large numbers of helpful votes (e.g., from 5 helpful votes to 500+), with 3,905 (43%) and 5,192 (63%) fake reviews attracting five or more helpful votes in the "Computers" and "Books" categories, respectively. It seems that the use of helpful votes alone may not effectively distinguish between spam and ham. This finding is consistent with the observations of previous studies that helpful votes may not be a good predictor in identifying spam [Danescu-Niculescu-Mizil et al. 2009; Jindal and Liu 2008]. Indeed, user-contributed helpful votes may also be fake. Nevertheless, more research is needed to examine the effectiveness of using helpful votes alone or in combination with other features to identify fake reviews (Figure 12).

8. CONCLUSIONS

Guided by the design science research methodology, one of the main contributions of our research is the development and instantiation of novel computational models to combat online review spam. The prototype system is evaluated based on TREC-like evaluation procedures and performance measures. Our experimental results confirm that semantic language modeling and a text mining-based computational model are effective for the detection of untruthful reviews, even if spammers exercise obfuscation strategies. In particular, the proposed computational model outperforms other well-known baseline models in analyzing the Amazon review dataset. Moreover, the proposed SVM computational model is more effective in detecting non-reviews than other supervised machine learning models. The proposed computational models achieve a true positive rate of over 95% in fake review detection. Empowered by the design artifacts, an empirical study of the trustworthiness of online consumer reviews is then performed. Based on a sample of over 2 million Amazon reviews, it is found that around 2% of these reviews are fake. Although the average spam rate is not particularly high, the display sequence of fake reviews may greatly influence consumers' purchasing decisions. For instance, if a Web site displays consumer reviews in reverse chronological order and a large number of fake reviews have been recently generated, the first page of reviews shown to consumers is likely to be filled by spam.

Although our prototype system is not a fully-fledged commercial system for review spam detection, our design artifacts clearly provide the critical modules for a fully

operational review spam detection system. A managerial implication of our research is that business managers or marketers will utilize our design artifacts to detect and remove fake reviews related to their products and services. This will enable more effective product design strategies and marketing plans to be developed based on the sheer volume of genuine user-contributed consumer reviews. A societal implication of our research is that ordinary consumers will be able to have a better comparison shopping experience by applying our design artifacts to locate and browse genuine product reviews. Future research involves the evaluation of the effectiveness and efficiency of the design artifacts based on a larger dataset (e.g., the entire review collections from different e-commerce Web sites). In addition, we will examine more sophisticated language modeling approaches, such as n-gram language models, to improve the effectiveness of the untruthful review detection method. The semantic granularity of text [Yan et al. 2011] will also be examined as a candidate feature for untruthful review detection. Finally, the impact of fake reviews on product sales will be examined based on econometric analysis.

REFERENCES

Abbasi, A., Zhang, Z., Zimbra, D., Chen, H., and Nunamaker Jr., J. F. 2010. Detecting fake websites: The contribution of statistical learning theory. *MIS Quart. 34*, 3, 435–461.

Abbasi, A., Chen, H., Nunamaker Jr., J. F. 2008. Stylometric identification in electronic markets: Scalability and robustness. *J. Manag. Inf. Syst. 25*, 1, 49–78.

Agrawal, R. and Srikant, R. 1994. Fast algorithms for mining association rules in large databases. In *Proceedings of 20th International Conference on Very Large Data Bases*. 487–499.

Benevenuto, F., Rodrigues, T., Almeida, V., Almeida, J., and Gonçalves, M. 2009. Detecting spammers and content promoters in online video social networks, In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 620–627.

Berger, A. and Lafferty, J. 1999. Information retrieval as statistical translation. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 222–229.

Bratko, A., Filipič, B., Cormack, G. V., Lynam, T. R., and Zupan, B. 2006. Spam filtering using statistical data compression models. *J. Mach. Learn. Res. 7*, 2673–2698.

Chang, M. W., Yih, W. T., and Meek, C. 2008. Partitioned logistic regression for spam filtering. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 97–105.

Chen, F., Tan, P., and Jain, A. 2009. A co-classification framework for detecting web spam and spammers in social media web sites. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. 1807–1810.

Cheung, M. Y., Luo, C., Sia, C. L., and Chen, H. 2009. Credibility of electronic word-of-mouth: Informational and normative determinants of on-line consumer recommendations. *Int. J. Electron. Commerce 13*, 4, 9–38.

Chowdhury, A., Frieder, O., Grossman, D., and McCabe M. 2002. Collection statistics for fast duplicate document detection. *ACM Trans. Inf. Syst. 20*, 2, 171–191.

Cormack, G. V. and Lynam, T. R. 2005. TREC 2005 spam track overview. http://plg.uwaterloo.ca/∼gvcormac/trecspamtrack05.

Cormack, G. V. 2007. TREC 2007 spam track overview. http://trec.nist.gov/pubs/trec16/papers/SPAM.OVERVIEW16.pdf.

Cormack, G. V., Hidalgo, J., and Sánz, E. 2007a. Spam filtering for short messages. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*. 313–319.

Cormack, G. V., Hidalgo, J., and Sánz, E. 2007b. Online supervised spam filter evaluation, *ACM Trans. Inf. Syst. 25*, 3, Article 11.

Danescu-Niculescu-Mizil, Kossinets, C., Kleinberg, J., and Lee, L. 2009. How opinions are received by online communities: A case study on amazon.com helpfulness votes. In *Proceedings of the 18th International Conference on World Wide Web*. 141–150.

Dellarocas, C. 2003. The digitization of word of mouth: Promise and challenges of online feedback mechanisms. *Manag. Sci. 49*, 10, 1407–1424.

Dellarocas, C. 2006. Strategic manipulation of internet opinion forums: Implications for consumers and firms. *Manag. Sci. 52*, 10, 1577–1593.

FETTERLY, D., MANASSE, M., AND NAJORK, M. 2005. Detecting phrase-level duplication on the world wide web. In *Proceedings of the 28th Annual International ACM SIGIR Conference*. 170–177.

GEFEN, D., BENBASAT, I., AND PAVLOU, P. A. 2008. A research agenda for trust in online environments. *J. Manag. Inf. Syst. 24*, 4, 275–286.

GHOSE, A., AND IPEIROTIS, P. G. 2007. Designing novel review ranking systems: Predicting the usefulness and impact of reviews. In *Proceedings of the 9th International Conference on Electronic Commerce*. 303–309.

GOLDBERG, D. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, MA.

GYÖNGYI, A. AND GARCIA-MOLINA, H. 2005. Web spam taxonomy. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web*. 39–47.

HAND, D. AND TILL, R. 2001. A simple generalisation of the area under the ROC curve for multiple class classification problems. *Mach. Learn. 45*, 2, 171–186.

HEVNER, A., MARCH, S., PARK, J., AND RAM, S. 2004. Design science in information systems research. *MIS Quart. 28*, 1, 75–105.

JINDAL, N. AND LIU, B. 2007a. Analyzing and detecting review spam. In *Proceedings of the 7th IEEE International Conference on Data Mining*. 547–552.

JINDAL, N. AND LIU, B. 2007b. Review spam detection. In *Proceedings of the 16th International Conference on World Wide Web*. 1189–1190.

JINDAL, N. AND LIU, B. 2008. Opinion spam and analysis. In *Proceedings of the International Conference on Web Search and Web Data Mining*. 219–229.

JINDAL, N., LIU, B., AND LIM, E. P. 2010. Finding unusual review patterns using unexpected rules. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. 1549–1552.

JOACHIMS, T. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*. 137–142.

JOACHIMS, T. 1999. Making large–scale SVM learning practical. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge, MA.

KIM, S.-M., PANTEL, P., CHKLOVSKI, T., AND PENNACCHIOTTI, M. 2006. Automatically assessing review helpfulness. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 423–428.

KULLBACK, S. AND LEIBLER, R. A. 1951. On information and sufficiency. *Ann. Math. Statist. 22*, 1, 79–86.

LAFFERTY, J. AND ZHAI, C. 2001. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 111–119.

LAU, R. Y. K., LIAO, STEPHEN, S. Y., AND XU, K. 2010. An empirical study of online consumer review spam: A design science approach. In *Proceedings of the 31st International Conference on Information Systems*.

LAU, R. Y. K., SONG, D., LI, Y., CHEUNG, C. H., HAO, J. X. 2009a. Towards a fuzzy domain ontology extraction method for adaptive e-learning, *IEEE Trans. Knowl. Data Engin. 21*, 6, 800–813.

LAU, R. Y. K., LAI, C. L., MA, J., AND LI, Y. 2009b. Automatic domain ontology extraction for context-sensitive opinion mining. In *Proceedings of the 30th International Conference on Information Systems*.

LAU, R. Y. K., LAI, C. L., AND LI, Y. 2009c. Leveraging the Web context for context-sensitive opinion mining. In *Proceedings of the IEEE International Conference on Computer Science and Information Technology*. 467–471.

LAU, R. Y. K., BRUZA, P. D., AND SONG, D. 2008. Towards a belief revision based adaptive and context sensitive information retrieval system. *ACM Trans. Inf. Syst. 26*, 2, Article 8.

LAU, R. Y. K., TANG, M., WONG, O., MILLINER, S., AND CHEN, Y. 2006. An evolutionary learning approach for adaptive negotiation agents. *Int. J. Intel. Syst. 21*, 1, 41–72.

LAU, R. Y. K. 2003. Context-sensitive text mining and belief revision for intelligent information retrieval on the web. *J. Web Intell. Agent Syst. 1*, 3-4, 151–172.

LIM, E. P., NGUYEN, V. A., JINDAL, N., LIU, B. AND LAUW, H. W. 2010. Detecting product review spammers using rating behaviors, In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. 939–948.

LIN, Y. R., SUNDARAM, H., CHI, Y., TATEMURA, J., AND TSENG, B. L. 2008. Detecting splogs via temporal dynamics using self-similarity analysis, *ACM Trans. Web 2*, 1, Article 4.

LIU, X. AND CROFT, B. 2004. Cluster-Based retrieval using language models. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 186–193.

LIU, Y., HUANG, X., AN, A., AND YU, X. 2008. Modeling and predicting the helpfulness of online reviews. In *Proceedings of the 8th IEEE International Conference on Data Mining*. 443–452.

MACDONALD, C. AND OUNIS, I. 2007. Overview of the TREC 2007 blog track. In *Proceedings of the 16th Text REtrieval Conference*. http://trec.nist.gov/pubs/trec16/.

MACDONALD, C., OUNIS, I., AND SOBOROFF, I. 2009. Is spam an issue for opinionated blog post search? In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 710–711.

MARCH, S. T., AND STOREY, V. C. 2008. Design science in the information systems discipline. *MIS Quart. 32*, 4, 725–730.

MARTINEZ-ROMO, J. AND ARAUJO, L. 2009. Web spam identification through language model analysis. In *Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web*. 21–28.

MISHNE, G., CARMEL, D., AND LEMPEL, R. 2005a. Blocking blog spam with language model disagreement. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web*. 1–6.

MILLER, G., BECKWITH, R., FELLBAUM, C., GROSS, D., AND MILLER, K. 1990. Introduction to WordNet: An on-line lexical database. *J. Lexicogr. 3*, 4, 234–244.

MITCHELL, T. 1997. *Machine Learning*. McGraw-Hill, New York.

NADAS, A. 1984. Estimation of probabilities in the language model of the IBM speech recognition system, *IEEE Trans. Acoust. Speech Signal Process. 32*, 4, 859.

NIE, J. Y., CAO, G., AND BAI, J. 2006. Inferential language models for information retrieval. *ACM Trans. Asian Lang. Inf. Process. 5*, 4, 296–322.

NTOULAS, A., NAJORK, M., MANASSE, M., AND FETTERLY, D. 2006. Detecting spam web pages through content analysis. In *Proceedings of the 15th International Conference on World Wide Web*. 83–92.

PAPADIMITRIOU, C. H. 1994. *Computational Complexity*. Addison-Wesley, Reading, MA.

PEFFERS, K., TUUNANEN, T., ROTHENBERGER, M., AND CHATTERJEE, S. 2008. A design science research methodology for information systems research. *J. Manag. Inf. Syst. 24*, 3, 45–77.

PERRIN, P. AND PETRY, F. 2003. Extraction and representation of contextual information for knowledge discovery in texts. *Inf. Sci. 151*, 125–152.

PISKORSKI, J., SYDOW, M., AND WEISS, D. 2008. Exploring linguistic features for web spam detection: A preliminary study. In *Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web*. 25–28.

PONTE, J. AND CROFT, B. 1998. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 275–281.

PORTER, M. 1980. An algorithm for suffix stripping. *Program 14*, 3, 130–137.

RILOFF, E. M., WILSON, T., HOFFMANN, P., SOMASUNDARAN, S., KESSLER, J., WIEBE, J., CHOI, Y., CARDIE, C., AND PATWARDHAN, S. 2005. Opinionfinder: A system for subjectivity analysis. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. 34–35.

SALTON, G. AND MCGILL, H. J. 1983. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York.

SANDERSON, M. AND CROFT, B. 1999. Deriving concept hierarchies from text. In *Proceedings of the 22nd Annual International ACM SIGIR Conference*. 206–213.

SHANNON, C. 1948. A mathematical theory of communication. *Bell Syst. Tech. J. 27*, 379–423.

XIAO, B. AND BENBASAT, I. 2011. Product-related deception in e-commerce: A theoretical perspective. *MIS Quart. 35*, 1, 169–195.

YAN, X. LAU, R. Y. K., SONG, D., LI, X., AND MA, J. 2011. Towards a semantic granularity model for domain-specific information retrieval. *ACM Trans. Inf. Syst. 29*, 3, Article 15.

ZHELEVA, E., KOLCZ, A., AND GETOOR, L. 2008. Trusting spam reporters: A reporter-based reputation system for email filtering. *ACM Trans. Inf. Syst. 27*, 1, Article 3.

ZHOU, B. AND PEI, J. 2009. Link spam target detection using page farms. *ACM Trans. Knowl. Discov. Data 3*, 3, Article 13.

Received April 2011; revised August 2011; accepted September 2011