

The McMaster Journal of Communication  
Vol 12 No 1  
2020

Electronic word-of-mouth marketing  
on Amazon:  
Exploring how and to what extent Amazon  
reviews affect sales

Kristine D'Arbelles, Pauline Berry, and Ashika Theyyil

McMaster University

Copyright ©

2020 by the authors.

The McMaster Journal of Communication is produced by The Berkeley Electronic Press  
(bepress).

<http://digitalcommons.mcmaster.ca/mjc>

# Electronic word-of-mouth marketing on Amazon: Exploring how and to what extent Amazon reviews affect sales

Kristine D'Arbelles<sup>1</sup>, Pauline Berry<sup>2</sup>, and Ashika Theyyil<sup>3</sup>

## Abstract

Consumers today base many of their decisions on peer referrals and online reviews. With the omnipresence of social media and online reviews, electronic word-of-mouth marketing (eWOM) has become a priority for many companies for both business growth and reputational management. The objective of this study is to examine the effectiveness of eWOM and its impact on sales. This study also seeks to help organizational leaders understand the significance of eWOM and its role in effective consumer and stakeholder relations, and in overall brand management. The researchers of this project explored eWOM by examining Amazon reviews from two different Kickstarter companies to determine which elements of online reviews impact product sales. By overlaying Amazon review data and sales figures from each Kickstarter company, researchers were able to determine the review factors that companies should focus on to increase their sales and grow their brands. The results of this study show that products with a high volume of positive reviews made by verified purchasers positively correlate to product sales.

*Keywords:* electronic word-of-mouth marketing, Amazon, online reviews, Kickstarter, sales, reputational management, brand reputation, online reputation management

---

<sup>1</sup> Currently enrolled graduate student of the Master of Communications Management program at McMaster University. E-mail: kristinedarbelles@gmail.com

<sup>2</sup> Currently enrolled graduate student of the Master of Communications Management program at McMaster University. E-mail: paulinebonnie@gmail.com

<sup>3</sup> Currently enrolled graduate student of the Master of Communications Management program at McMaster University. E-mail: attheyyil@ryerson.ca

## **Definitions**

**WOM:** Word-of-mouth or WOM “is the act of consumers providing information about goods, services, brands, or companies to other consumers” (Babic, Sotgiu, de Valck, & Bijmolt, 2016, p. 297).

**eWOM:** Electronic word-of-mouth or eWOM is “information about goods, services, brands, or companies communicated to other consumers through the Internet (through, e.g., reviews, tweets, blog posts, “likes,” “pins,” images, video testimonials)” (Babic et al., 2016, p. 297).

**Verified purchaser:** A verified purchaser is a tag created by Amazon to call out reviewers that “bought the product themselves, without receiving a discount from the vendor” (Hanbury, 2018, para. 28).

**Top reviewers:** Based on the review’s helpfulness rating, the recency of the review, and the number of reviews the reviewer has left on Amazon (Amazon, 2018).

**Dependant variable:** In this study, the dependent variable is the sales rank of Product A and Product B.

**Independent variable:** Any variable we incorporate (i.e. customer reviews, ratings, review length) into the study that (might) affect the dependant variable.

**Review valence:** Whether the review is associated with positive or negative sentiment.

**Review volume:** The total number of reviews for an Amazon product.

**Helpfulness rating:** The number of votes online shoppers give to a specific product review (Bhargava, n.d., p. 1).

**Star ratings:** A rating scale out of five that “allows you to share information on a product’s attributes” (Amazon, 2018).

**Bivariate correlation:** How two variables change together at the same time.

**Partial correlation:** Measures the degree of association between two random variables.

# Electronic word-of-mouth marketing on Amazon: Exploring how and to what extent Amazon reviews affect sales

Kristine D'Arbelles, Pauline Berry, and Ashika Theyyil  
McMaster University

## Introduction

Consumers today base many of their decisions on peer referrals and online reviews. With the omnipresence of social media and online reviews, electronic word-of-mouth marketing (eWOM) has become a priority for many companies for both business growth and reputational management. This research project examines the growth of two start-ups, Kickstarter A and Kickstarter B, who sell their products, Product A and Product B, on Amazon. Product A is a children's science craft kit and Product B is a reflective vest for athletes. Company identities and product information have been stripped from this research to maintain business confidentiality.

The objective of this research study is to understand whether selling products on Amazon, coupled with eWOM, drives positive net sales. Evaluating the effectiveness of eWOM is not a new endeavour. In fact, over the past 15 years, "more than 100 studies have investigated [how] eWOM is linked to the bottom line" (Babic et al., 2016, p. 298). However, "research in this arena is fragmented and we have yet to understand ... what aspects of online reviews influence sales" (Babic et al., 2016, p. 298). This study does not claim it will solve this equation, but its authors hope to add yet another layer to this area of research.

## Literature Review

### Electronic word-of-mouth marketing

With the rapid growth of social media, technology, and e-commerce platforms, electronic word-of-mouth (eWOM) has become an irrefutable component of today's sales model. EWOM uses digital channels (online reviews, likes, social media, etc.) to communicate information

“about goods, services, brands, or companies to other consumers” (Babic et al., 2016, p. 298). According to Naveen and Tung (2007), eWOM “has played an important role in customers’ perception of...brand name[s]” (p. 38). Silverman (2001) posits that eWOM has “credibility, reach, speed, and agility... it gets people to act, it’s a purchase trigger” (p. 24). According to Zhu and Zhang (2010) eWOM is “crucial for niche product producers” to sell their products online, because online reviews are a “primary source for consumer [product] information” (p. 145). According to Pezzuti and Leonhardt (2018), eWOM is important for all aspects of a business as it “influences a range of important marketing outcomes such as product sales, customer loyalty, and stock market performance” (p. 177).

Given the positive effects of online reviews, retailers are incorporating eWOM into their marketing toolkits. Enticing customers to leave online reviews has become a primary focus for many companies. Existing research on why consumers participate in eWOM “primarily focused on...the need to belong, the desire to help others, self-enhancement, and pursuing economic incentives” (Pezzuti & Leonhardt, 2018, p. 178). Forman, Ghose, and Wiesenfeld (2008) argue consumers are more likely to participate in eWOM when identity disclosure and location information are left by previous reviewers (p. 295). Authors posit that “a common geography [and other relatable traits] lowers perceived differences... and serves as a salient basis for a feeling of similarity with other members of the group” (Forman, Ghose, & Wiesenfeld, 2008, p. 295), fueling the likelihood consumers will leave a review or purchase the product.

### **Online reviews and sales**

With information at their fingertips, consumers are more empowered and educated than ever before. In a global survey of 5,000 shoppers, Floyd, Freling, Alhoqail, Cho, and Freling (2014) discovered that the number one place consumers go to for product information, prior to making a purchase, is “retailer sites” for online reviews (p. 217). The same study revealed that “70% of consumers...trust online product reviews” (Floyd, Freling, Alhoqail, Cho & Freling, 2014, p. 217). Zhu and Zhang (2010) assert that “buyers ... use online review systems as the primary source for quality information” (p. 145). Hence, there is a rather strong relationship between online reviews and sales (Floyd et al., 2014; Naveen & Tung, 2007; Chevalier & Mayzlin, 2006).

Many studies assert that more reviews stimulate more sales. In their 2014 study, Floyd et al. (2014) revealed that “the volume of online product reviews was positively associated with automobile, box office, and movie sales” (p. 220). Correspondingly, in their study of digital micro product sales, researchers Naveen and Tung (2007) asserted that the “volume of [online reviews]...has consistently shown to be a reliable predictor of sales” (p. 38). Additionally, Naveen and Tung (2007) found that a “1% increase in the number of reviews increased sales by 0.12%” (p. 44). This finding reveals that review volume has a direct impact on product sales. Hanbury (2018) also stated that “a high volume of customer reviews...improves conversion rates” and improves the products’ rank in Amazon search results (para. 4-5). Thus, existing literature shows a positive correlation between review volume and sales.

Many studies (Babic et al., 2016; Forman et al., 2008) argue that review valence is a better predictor of product sales than review volume. According to Forman et al. (2008), one reason “valence of consumer product reviews may influence sales is that [they] may serve as a proxy for underlying product quality” (p. 297). However, existing scholarship on review valence is fragmented. According to some studies, negative consumer reviews “are more powerful in decreasing sales than positive reviews are in increasing them” (Forman et al., 2008, p. 298). Conversely, other study results claim that negative reviews positively influence product sales (Babic et al., 2016, p. 298). In their research, Chevalier and Mayzlin (2006) highlighted that “relatively rare 1-star reviews carry a lot of weight with consumers” (p. 15), and thus, are more detrimental to sales, despite the presence of positive reviews.

Similarly, Floyd et al. (2014) believed that the reason negative reviews decrease sales more than positive reviews increase them is because “negative information is more strong, influential, predictive, and difficult to resist than positive information” (p. 223-224). Hao, Yi, Le, and Cheng (2010) argued that negative reviews decrease sales due to prospect theory which states that “one experience of loss appears greater [to consumers] than that of equivalent gain” (p. 4). Researchers highlighted that with a negative review, consumers are aware of a potential risk and “adjust behavior to avoid the potential risk of purchasing the product” (p. 4). However, according to the results of their 2007 study, Naveen and Tung found that “over 90% of the reviews [found on Amazon] were rated greater than 4, with only 10% of reviews being rated lower than 4” (p. 42). This finding asserted that the average customer rating might be as good of a sales predictor as originally proposed.

Forman et al. (2008) hypothesized that reviews containing “identity descriptive information” (p. 292) impact consumer purchase decisions more than any other review variables because they “influence perceived helpfulness...by reinforcing community norms” and increasing trust between consumers (p. 296). Similarly, Floyd et al. (2014) claimed that when “source characteristics” of the reviewer were explicitly stated in the review, the review was “perceived as more credible...and was associated with greater product sales” (p. 224).

### **Kickstarter Community**

Kickstarter began in April 2009 as an “ubiquitous, multi-category reward-based” crowdfunding platform (Calvo, 2015, p. 20). Crowdfunding can be defined as “an open call...through the internet for the provision of financial resources...to support initiatives for specific purposes” (Calvo, 2015, p. 16). Kickstarter, a well-known crowdfunding platform, has over 130,000 projects hosted on its site; 57,000 of which have successfully achieved their crowdfunding objectives. The crowdfunding market as a whole is expected to grow to \$100 billion in annual revenue within the next 20 years (Calvo, 2015). According to Mollick (2015), the success of a Kickstarter project depends on the “quality of the kickstarter page, the [frequency and quality] of communication with funders, and the word-of-mouth exposure on social networking sites” (Calvo, 2015, p. 22). Hence, eWOM is integral to project success on Kickstarter.

In an attempt to “offer Kickstarter products a way to reach more customers,” Amazon teamed up with Kickstarter and “dedicated a section on Amazon’s website” for Kickstarter companies (Perez, 2016, para. 1). As part of its Kickstarter offering, Amazon handles the aspects of the sales process that start-ups don’t have time to manage as they scale (Murphy, 2015, para. 5). These aspects include “managing inventory, marketing, customer service, and shipping” (Murphy, 2015, para. 5). By leveraging Amazon’s Kickstarter platform, Kickstarter companies can tap into Amazon’s extensive seller network, increase their market exposure, and grow their business (Murphy, 2015, para. 5).

### **Hypotheses and Research Questions**

This research study focuses on two areas, the effectiveness of Amazon reviews and the value of electronic word-of-mouth (eWOM) marketing on sales. The objective of the research is to discern the following hypothesis:

- **H<sub>1</sub>**- A company that sells its products on Amazon and primarily relies on word-of-mouth marketing can increase sales and grow their business.
- **H<sub>null</sub>**- Amazon and word-of-mouth marketing do not impact company sales.

Further, the main questions this study aims to answer include:

**RQ<sub>1</sub>**: Is there a positive correlation between review valence and product sales?

- Researchers overlaid star rating data with sales data and investigated whether a correlation between the two variables exists.

**RQ<sub>2</sub>**: Is there a positive correlation between review volume and product sales?

- Researchers overlaid the total number of reviews with sales data and examined whether sales increased (or decreased) with the total number of reviews.

**RQ<sub>3</sub>**: Is there a positive correlation between review validity of the reviewer, such as whether or not the reviewer was a verified purchaser, and whether the username looks like a real person’s name, and product sales?

- Researchers examined the relationship between the total number of reviews that were classified as verified purchasers and their impact on sales; researchers also overlaid the total number of reviews with real-looking usernames and sales data. The legitimacy of a username was hand-coded by the researchers. Real usernames were identified as usernames with one or two words, that had no numbers and looked like a real person's name (i.e. Mark, Laura Jones). Not-real usernames were identified as usernames with numbers and letters (i.e. CoolKat123), or usernames with multiple words that form a short phrase or sentence (i.e. This is Me).

**RQ4:** Is there a positive correlation between the length of a review and product sales?

- Researchers compared sales data with the average word count of reviews over time.

### **Data Collection and Analysis/Sampling Technique**

The researchers chose two Kickstarter companies for this research study: Kickstarter A and Kickstarter B. Both companies launched on Kickstarter and currently sell their products on Amazon. The two products this study will examine are Product A (\$59) and Product B (\$49.95).

### **Units of Measurement**

The analysis completed for this study includes nominal measures and ratio measures. Below is a breakdown of each unit of measurement:

- Nominal measures:
  - Validity of reviewer: Verified purchaser and username.
  - Questions the researchers asked: Was the review a verified purchase? Yes or No. Did usernames look like a real person's name? Yes or No.
- Ratio measures:
  - Review valance through star ratings on each review.
  - Time analysis comparing activity on Amazon with sales.
  - Length of review content.
  - Questions the researchers asked includes: How many total reviews had 4 star ratings? How many total reviews had 3 star ratings? Etc. What is the ratio of reviews made by verified to unverified users?

### **Measurement Techniques**

The online review dataset was extracted from Amazon and includes review dates for each product. To extract the data from Amazon, researchers used PyCharm, a Python coding tool, to write code that scrapes the review pages for both Product A and Product B. The complete and fully annotated code can be reviewed in Appendices A and B, or by visiting the researchers' GitHub page at <https://github.com/kristinedarbelles/amazonscraper>. The coder, Kristine D'Arbelles, who is a strategic communications manager by trade, used BeautifulSoup, a Python package, to parse the HTML code on the company review pages. Crummy.com (n.d.) was heavily consulted while building the code, as it had extensive documentation on BeautifulSoup. In addition, Python's online wiki was consulted when trying to build more complicated aspects of the code, such as the WhileLoop, which automatically iterates the code on all the review pages (WhileLoop, n.d.). Finally, as the coder was new to concepts of coding, she referenced Stack Overflow (n.d.) to figure out solutions to simple problems, such as converting variables to integers, converting a string to datetime (for better database integration), and inputting a delay and headers in the code to prevent Amazon from blocking the scraping program.



The code resulted in extracting several elements from each review on both product pages. These elements are:

Date of review	Body text	Helpfulness rating
Star review	Username and user ID	Number of comments on the review
Title of review	Verified purchaser	

Once data was scraped from Amazon, it was then ported into an SQL database, where the researchers made queries to answer their research questions. Prior to creating queries, quality tests were performed to ensure the data was correctly ported over to the database. The quality tests used SQL code to parse through the database and try and find errors. The complete SQL code used for the quality tests can be reviewed in Appendix C.

### **Sampling Technique**

The representative sample for this research study was online customer reviews present on Amazon. While it is possible that there are customers outside of this measure that purchase the products studied, the sample population extracted captures the active online customer market. Specifically, the authors used non-probability/non-random sampling for this research study. Samples included:

- Customers who reviewed each product on Amazon; totalling 1,100 reviews.
  - Time period: First 20 months of business.
  - Kickstarter A: May 2013 to December 2014.
  - Kickstarter B: October 2012 to May 2014.
- The sampling units included: Amazon reviews and sales figures.
- The corpus of this study combines 1,100 Amazon reviews and sales data from both Kickstarter A and Kickstarter B.

### **Methods**

Two types of methods were used to analyze the data. These include: correlational and content analysis.

#### **Correlational analysis**

The researchers examined the correlation between the company's sales and customer reviews on Amazon. Research on review valence was conducted using the reviews' star rating. Review metadata, such as username, length of body content, and whether or not the review was written by a verified purchaser, were also analyzed. These findings were overlaid with sales figures to establish trends and correlations between the data sets.

## Content analysis

The researchers examined manifest and latent content in the analysis. The manifest content, which can be measured and quantified (Stacks, 2017), pulled the following items into the researchers' dataset:

- Date of Amazon review
- Number of Amazon reviews
- Star rating of Amazon reviews
- Helpfulness rating for Amazon reviews
- Sales figures from Product A and Product B

The latent content within the study, things that have deeper meaning and elicit themes (Stacks, 2017), include:

- The validity of the user who wrote the Amazon review. The researchers identified username and user ID from the dataset to determine if there were duplicate or multiple reviews by the same person.
- Whether the user that left the review made a verified purchase: Yes or no. This is important as verified reviews are more likely to be at the top of the review list, which is what potential customers see first.

## Research Analysis and Discussion

In line with the theories outlined in the literature review, total reviews are strongly correlated with an increase in sales for both companies. As seen in Figures 1 and 2, as reviews increase, so do sales.

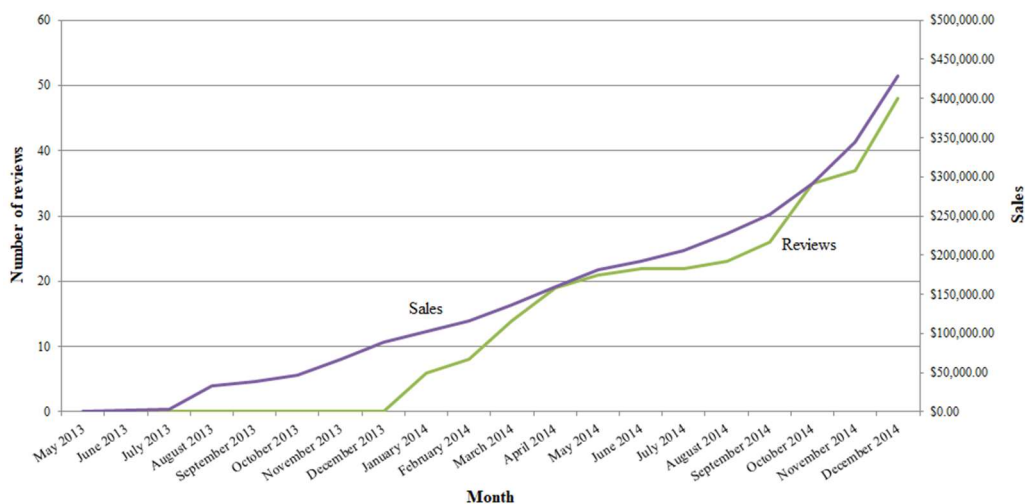


Figure 1. Total number of reviews and sales over time for Kickstarter A.

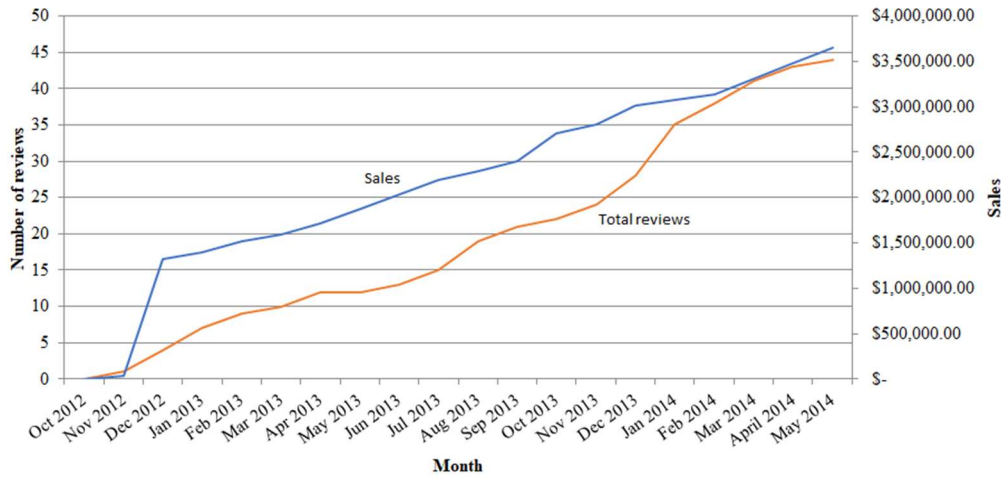


Figure 2. Total number of reviews and sales over time for Kickstarter B.

In a study of book reviews, Chevalier and Mayzlin (2006) argued that “the coefficients for the average star value suggest that sales improve when books are rated more highly” (p. 14). The data in this study confirmed Chevalier and Mayzlin’s study findings. Figures 3 and 4 demonstrate that reviews given a five-star rating tracked sales more closely than four-, three-, two- or one-star ratings. The results showed this is true for both companies.

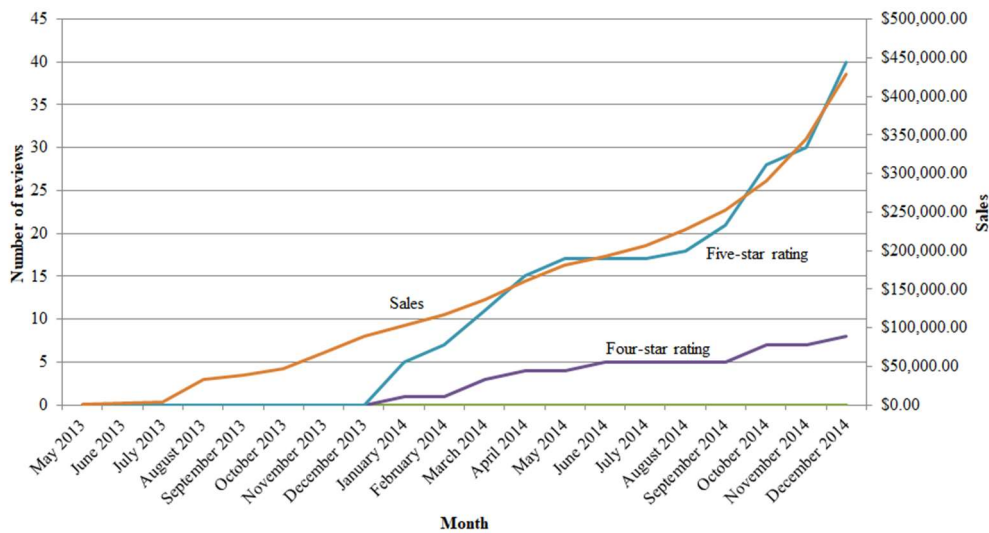


Figure 3. Number of reviews with five-, four-, three-, two- and one-star ratings compared to sales for Kickstarter A.

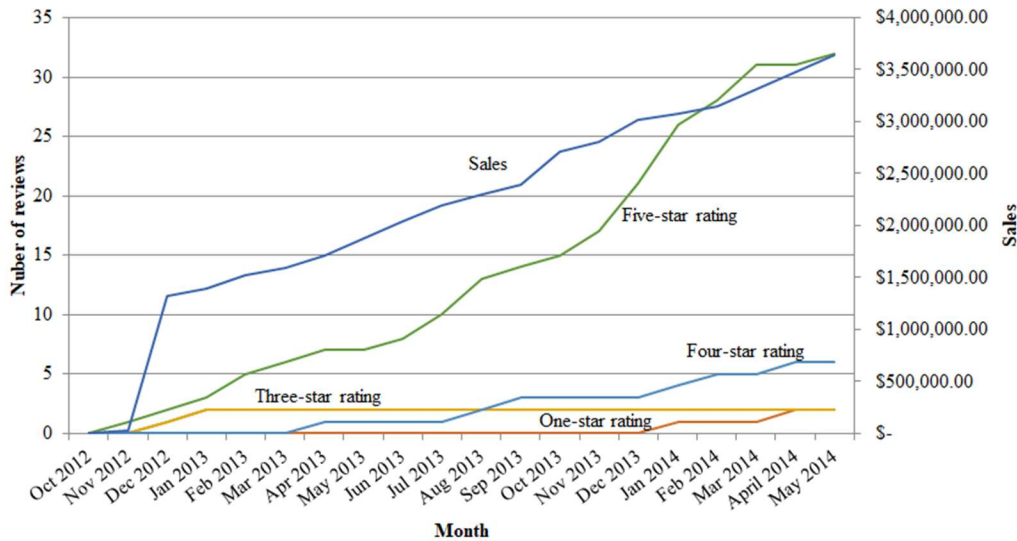


Figure 4. Number of reviews with five-, four-, three-, two- and one-star ratings compared to sales for Kickstarter B.

While the star rating seems to have a positive correlation with sales, the researchers cannot accurately claim that the increase in sales is due to the high star ratings. More than 87% of Kickstarter A's reviews had a five-star rating. One-star, two-star, three-star and four-star ratings accounted for only an eighth of the data. Further, nearly three quarters (73.84%) of Kickstarter B's reviews had five-star ratings. Kickstarter B's one-star, two-star, three-star and four-star review ratings only accounted for 26% of total reviews. There is not enough data on the lower star ratings to effectively determine their impact on sales.

As Forman et al. (2008) highlighted in their study of online reviews, "attributes of a message source often exert direct effects on message recipient's attitudes and behaviors, independent of the message content" (p. 292). In other words, if the review looks like it came from a real person, then it is more likely to influence the purchase behavior of potential customers. This theory was verified in the analysis of Kickstarter A and Kickstarter B data. As seen in Figures 5 and 6, reviews from verified purchasers increased with the increase of sales. In particular, reviews written by verified purchasers, reviewers who are not paid to write reviews, are seen to have a strong correlation with sales.

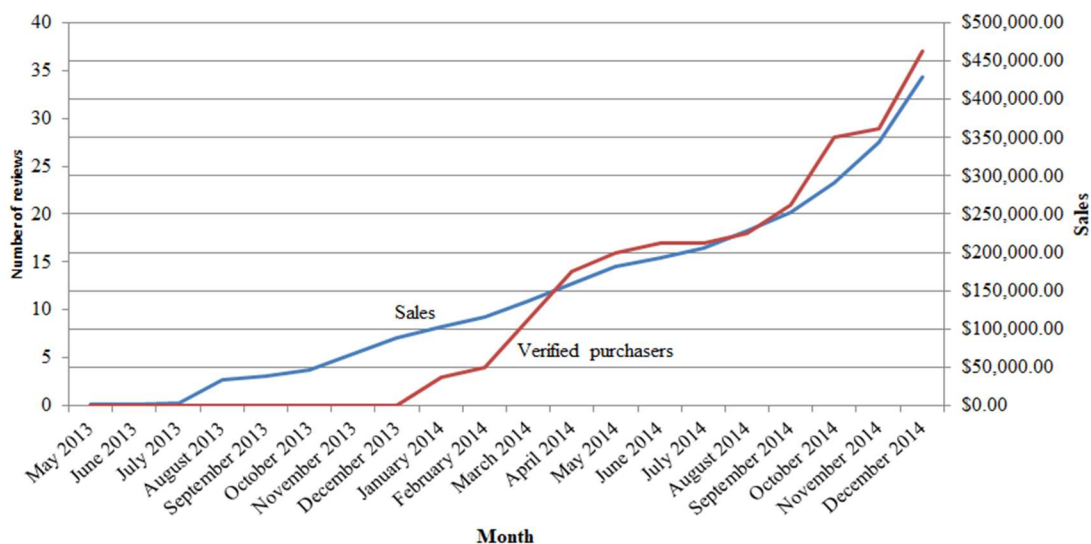


Figure 5. Reviews written by verified purchasers and sales of Kickstarter A.

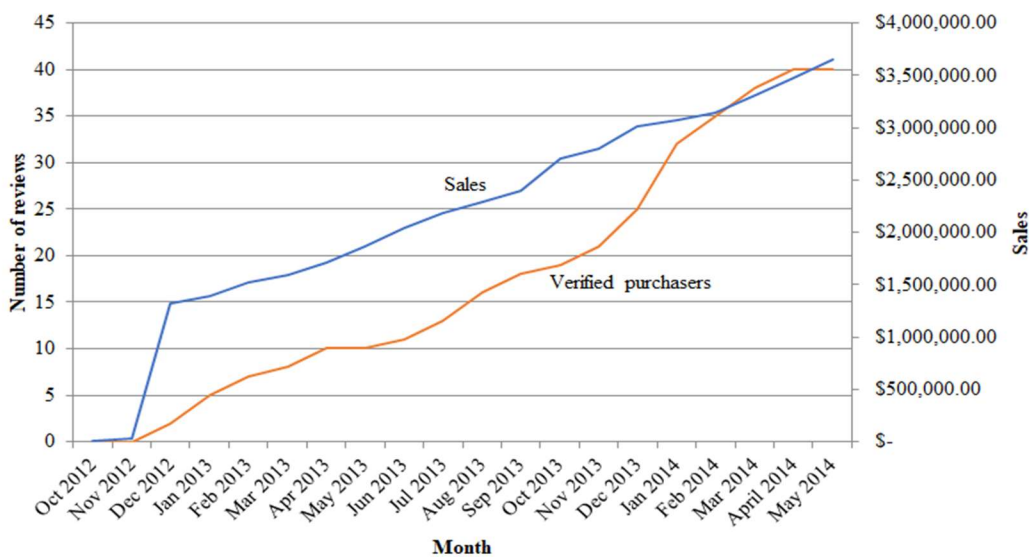


Figure 6. Reviews written by verified purchasers and sales of Kickstarter B.

However, not enough information is available to test the impact of reviews by verified purchasers on sales. Nearly all Kickstarter A reviews (91.92%) and Kickstarter B reviews (92.15%) were from verified purchasers. There is an insufficient amount of data from reviews written by unverified users to be able to establish their impact on sales.

Another attribute this study analyzed was the impact of a reviewer’s username on product sales. Reiterating the theory brought forth by Forman et al. (2008), review usernames that look like real people contribute to an increase in sales. However, Figures 7 and 8 contradict Forman et al.’s (2008) theory, as the study’s findings found pseudonym-type usernames tended to correlate more closely with sales rather than reviews with real names. This result is statistically significant

as username types (pseudonym and real name) were split almost evenly within the corpus of Amazon review data. Real looking usernames (ex. Jane Doe) appeared 58.33% of the time in Kickstarter A reviews and 56.82% of the time in Kickstarter B reviews. Pseudonym usernames (ex. Blacklight1234) appeared 41.67% of the time on Kickstarter A reviews and 43.18% in Kickstarter B reviews.

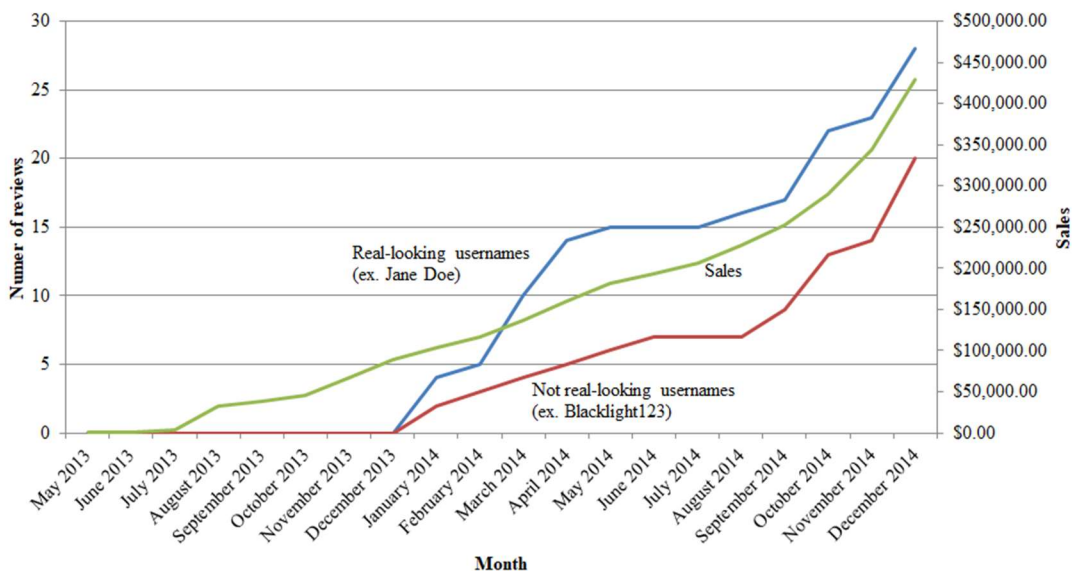


Figure 7. Attribute of username and sales of Kickstarter A.

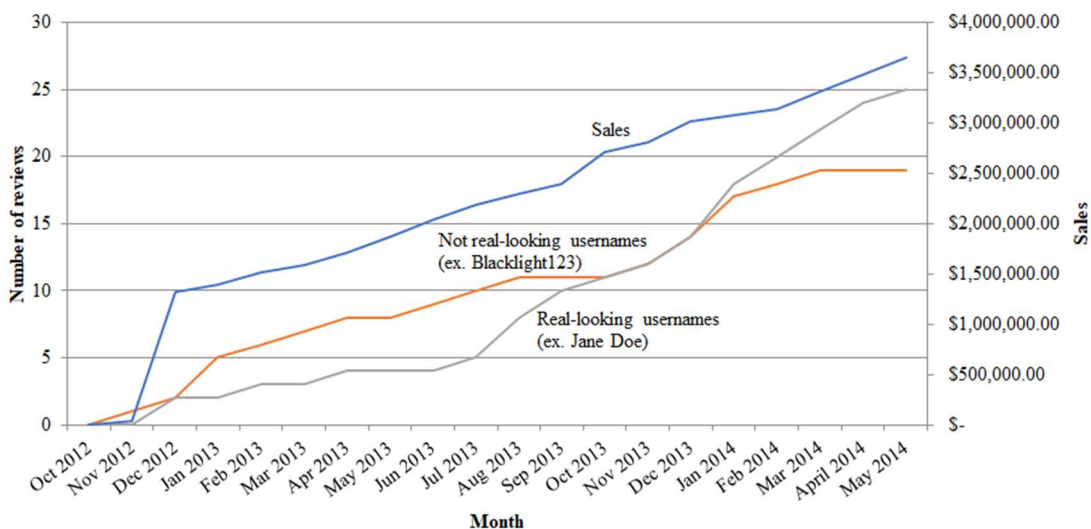


Figure 8. Attribute of username and sales of Kickstarter B.

Finally, the last review factor analyzed was the length of the review and its impact on sales. The length of the review is measured via word count. Figure 9 demonstrates that there is no clear correlation between average word count of Kickstarter A reviews and monthly sales.

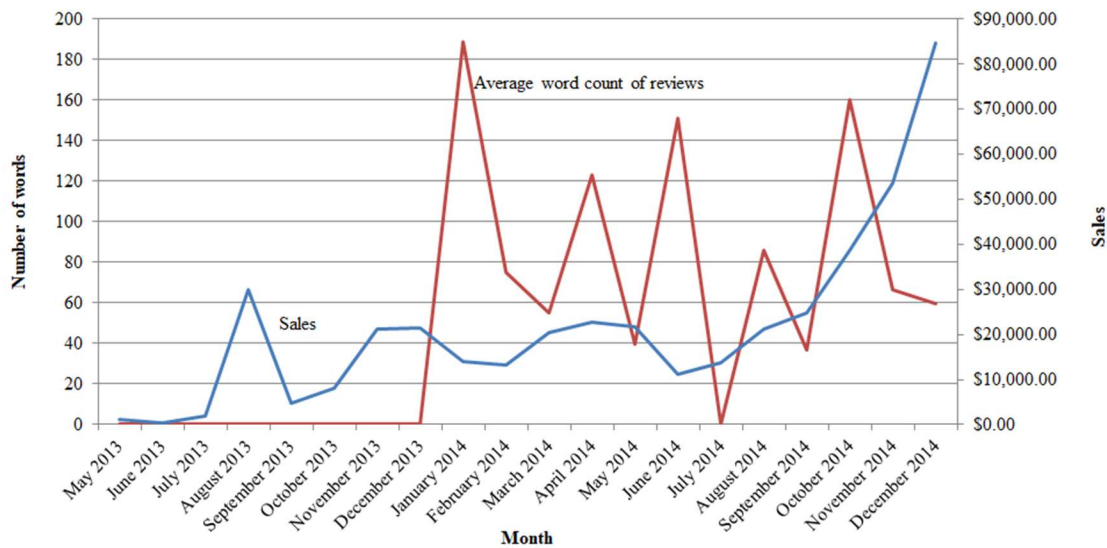


Figure 9. The average word count of Kickstarter A reviews in a given month compared with monthly Kickstarter A sales.

Contrastingly, the study shows that the word count in Kickstarter B’s online reviews correlates heavily with its product sales. To highlight, Kickstarter B’s sales increase in December, October, and January, where the length of reviews is higher. In months where the length of reviews is lower (i.e. February, May and November), Kickstarter B’s sales decrease.

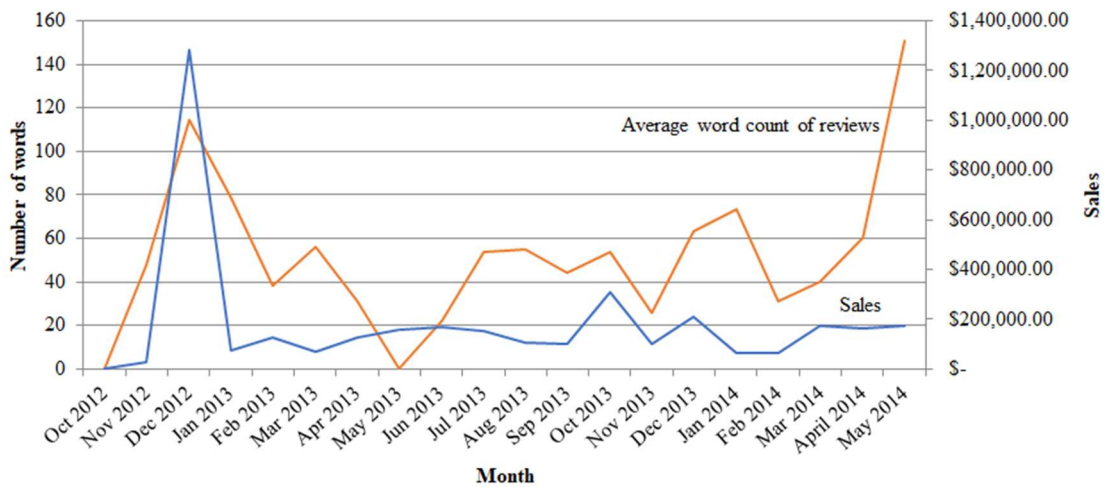


Figure 10. The average word count of Kickstarter B reviews in a given month compared with monthly Kickstarter B sales.

Given the differing results for each company, the researchers examined Kickstarter B data to see if there were other factors that may have affected its high correlation. The researchers noticed that the two longest reviews written about the Kickstarter B product, at 492 words and



413 words, occurred during a month when Kickstarter B saw an increase in sales. However, upon further analysis, the cause for the spike in sales is likely due to seasonality, and less likely due to a long review. The longest review written about Kickstarter B's product was published in April 2014, or the spring, when runners prepare for summer activity – given this product's seasonality, this was an interesting find. The review of 413 words was written in October 2014, or the fall, when runners prepare for winter running and holiday gifts. On the other hand, April also saw one of the shortest reviews written, at only 21 words. Therefore, researchers concluded there was not enough data to answer their fourth research question. More research is required to better establish the correlation between length of reviews and sales.

### **Recommendations, Conclusions & Future Directions**

This study indicates that there is only a small (albeit, positive) correlation between review length and sales. This study also finds that there is minimal benefit in including personal identifiers in reviews. In practice, sales aren't made because a review looks like it comes from a legitimate source. This study has shown that other factors, such as star rating, the total number of reviews and the status of the reviewer as a verified purchaser, have a much more positive effect on sales.

As part of the research questions, the authors examined whether the reviewer's public identity impacted sales. The research shows that the public identity of a reviewer has a weak correlation with sales, weaker than the positive correlation seen with anonymous-type usernames, such as Blacklight123. This could be attributed to the growing anonymous online culture - an avenue of research that warrants further exploration and could help companies understand the impact of anonymous online reviews on sales and brand reputation.

The key findings of this study also conclude that if a company has a large volume of online reviews that are positive, hold a five-star rating, and are written by a verified purchaser, the company should see an increase in sales. To optimize the effect of online reviews, the researchers recommend that companies focus less on the length of the review and personal details of the reviewer, and more time collecting online reviews with high star ratings from verified purchasers. More interestingly, this research reinforces Aula and Heinonen's (2016) theory that in today's relationship-building model, "who we trust has become completely random" (p. 45), and controlling an organization's online reputation seems insurmountable. However, as this study's results have shown, there are tangible steps management can take to help control their online narrative.

To further enhance and increase the study's application, the authors recommend that future researchers examine eWOM from products that are sold by larger online retailers like Walmart, Target, and Costco. These retailers would provide a more comprehensive data set. Additionally, the authors recommend future researchers explore and compare other third-party review sites to the reviews found on Amazon, especially given that Amazon tends to produce more positive reviews (Naveen & Tung, 2007). Reviewing other sites that have a more balanced set of comments could help provide a more realistic valence of the reviews. Additional data is required to decide whether the length of reviews impacts sales, as the study found minimal



correlation between these two factors. Albeit, the researchers found a correlation between the seasonality of Kickstarter A and Kickstarter B's review length. However, to be able to discern the factors impacting sales, the researchers recommend that necessity products be analyzed instead (for example, soap or food) to eliminate the impact of seasonality in the study's results.

Future studies can glean deeper insights by overlaying online review data with social media data. With these data points, researchers can better examine the growth of eWOM, not just on review sites, but on social media platforms as well, providing a more realistic indication of eWOM's growth and its impact on sales.

### **Validity**

The nature of this study requires a high dependency on the participation of external parties (For example, Kickstarter and start-ups) and the data they are willing to supply, thus the datasets available may vary from study to study.

Using the theory of reliability, the researchers deliberately embedded repeatability and scalability into the study by explicitly outlining its research design, data collection process, and analysis (Boslaugh, 2012, p. 13). Further, to assess validity, the concept of concurrent validity was tested. Concurrent validity states that when performance is measured at the same time, inferences can be made to predict future behavior (Boslaugh, 2012). More specifically, in this study both products had successful sales on Amazon. Therefore, they have high concurrent validity. However, while absence of concurrent validity will not confirm a null hypothesis, different results from each company opened the door for further discussion and questions for future research projects.

To confirm the validity of the data, the researchers made a fundamental decision to not trust the first export of data. As the data was scraped from a website and ported into an SQL database, researchers could not determine whether the data made it from the website into the database correctly. Therefore, the researchers performed several SQL queries to test the validity of the data, which can be reviewed in Appendix C.

### **Limitations**

Some of the limitations encountered throughout the study included the limited amount of reviews for Product A and Product B. The reviews that the researchers extracted did not capture the entire universe of Kickstarter companies, making the research sample limited in scope. Secondly, the researchers only analyzed two retailers, Kickstarter A and Kickstarter B; thus, the findings could be a coincidence as there are many other potential factors that the researchers might not have explored due to limited time and scope of the study. Lastly, the researchers of this study were only able to obtain the first 20 months of sales data from Kickstarter A and Kickstarter B. For a deeper analysis, the researchers recommend that future studies include more financial data. With a more robust financial data set, researchers will be able to establish concrete trends and correlations.

The consideration for potential bias should be considered when reviewing this study. Additionally, bias may surface within the study if the researchers did not present enough contrary evidence. This can lead to confirmation bias where researchers use filtered information to justify their hypotheses. A lack of contrary evidence and diverse perspectives can limit a holistic viewpoint and increase the bias in the findings.

## References

- Aula, P., & Heinonen, J. (2016). *The reputable firm: How digitalization of communication is revolutionizing reputation management*. Cham: Springer.
- Babic, A., Sotgiu, F., de Valck, K., & Bijmolt, T. H. (2016). The effect of electronic word of mouth on sales: A meta-analytic review of platform, product, and metric factors. *Journal of Marketing Research*, 53(3), 297–318. doi: 10.1509/jmr.14.0380.
- Bhargava, S. (n.d.). Predicting Amazon review helpfulness. *University of California, San Diego*, 1-7. Retrieved from <https://pdfs.semanticscholar.org/471e/55a93e841bbb6887cf3e70b6925fdeba9eb9.pdf>
- Boslaugh, S. (2012). *Statistics in a Nutshell* (2<sup>nd</sup> ed.). Sebastopol, CA: O'Reilly.
- Calvo, S. (2015). Funding characteristics of an established crowdfunding platform. *Michigan Journal of Business*, 9(1), 10-49.
- Chevalier, J.A., & Mayzlin, D. (2006). The effect of word of mouth on sales: Online book reviews. *Journal of Marketing Research*, 43(3), 345-354.
- Crummy. (n.d.). *Beautifulsoup documentation* [Online]. Retrieved from <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
- Forman, C., Ghose, A., & Wiesenfeld, B. (2008). Examining the relationship between reviews and sales: The role of reviewer identity disclosure in electronic markets. *Information Systems Research*, 19(3), 291-313. doi 10.1287/isre.1080.0193.
- Floyd, K., Freling, R., Alhoqail, S., Cho, H. Y., & Freling, T. (2014). How online product reviews affect retail sales: A meta-analysis. *Journal of Retailing*, 90(2), 217-232. doi:10.1016/j.jretai.2014.04.004.
- Github. (n.d.). *Amazon scraper* [Online]. Retrieved from <https://github.com/kristinedarbelles/amazonscraper>
- Hanbury, M. (2018, April 9). Amazon has an underground subculture that trades reviews for deals: And the company is finally cracking down. *Business Insider*. Retrieved from <https://www.businessinsider.com/amazon-bad-review-practices-crackdown-2018-4>
- Hao, Y., Ye, Q., Li, Y., & Cheng, Z. (2010). How does the valence of online consumer reviews matter in consumer decision making? Differences between search goods and experience goods. *2010 43rd Hawaii International Conference on System Sciences (HICSS)*, 1-10. Retrieved from <http://eds.b.ebscohost.com.libaccess.lib.mcmaster.ca/eds/detail/detail?vid=0&sid=6cd78160-d82b-49e3-9005->

[a1be21857938%40sessionmgr101&bdata=JnNpdGU9ZWRzLWxpdmUmc2NvcGU9c2l0ZQ%3d%3d#AN=81665793&db=edb](http://a1be21857938%40sessionmgr101&bdata=JnNpdGU9ZWRzLWxpdmUmc2NvcGU9c2l0ZQ%3d%3d#AN=81665793&db=edb)

- Kickstarter. (n.d.). *About us* [Online]. Retrieved from <https://www.kickstarter.com/about?ref=global-footer>
- Perez, S. (2016, July 27). Amazon debuts a dedicated shop for Kickstarter products. *Tech Crunch*. Retrieved from <https://techcrunch.com/2016/07/27/amazon-debuts-a-dedicated-shop-for-kickstarter-products/>
- Pezzuti, T., & Leonhardt, J. M. (2018). Is online sharing and word of mouth more prevalent among collectivist consumers? *Academy of Marketing Science*, 177-187. doi: 10.1007/978-3-319-66023-3\_65
- Murphy, M. (2015, July 28). You can now buy your favorite Kickstarter products on Amazon. *Quartz*. Retrieved from <https://qz.com/465725/you-can-now-buy-your-favorite-kickstarter-products-on-amazon>
- Naveen, A., & Tung, B. (2007). The impact of electronic-word-of-mouth on digital microproducts: An empirical investigation of Amazon shorts. *Proceedings*, 27. Retrieved from <http://aisel.aisnet.org/ecis2007/27>
- Silverman, G. (2001). *The secrets of word-of-mouth marketing: How to trigger exponential sales through runaway word of mouth*. New York, NY: AMACOM.
- Stacks, D. W. (2017). *Primer of public relations research* (3rd ed.). New York, NY: The Guilford Press. [Kindle Edition].
- Stack Overflow. (n.d.). *Current community* [Online]. Retrieved July 28, 2018 from <https://stackoverflow.com/>
- WhileLoop. (n.d.). *While loops: Usage in Python* [Online]. Retrieved July 28, 2018 from <https://wiki.python.org/moin/WhileLoop>
- Zhu, F. & Zhang, X. (2010). Impact of online consumer reviews on sales: The moderating role of product and consumer characteristics. *Journal of Marketing*, 74, 133–148.

## Appendix A

Below is the complete Python code used to scrape database reviews for Product A.

```
1  import requests
2  # Based on a Google search, the most popular Python package used to scrape web pages.
3  from bs4 import BeautifulSoup
4  from datetime import datetime
5  import time
6  import sqlite3
7  import sys
8  import urllib.request
9
10 def create_connection(db_file):
11     try:
12         conn = sqlite3.connect(db_file)
13         return conn
14     except Exception as e:
15         print(e)
16     return None
17 # Code below is inserting the variables into a SQL database
18 def insert_review(conn,
19 ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments):
20     sql = ''' INSERT INTO Reviews[COMPANY NAME]
21 (ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments)
22         VALUES(?,?,?,?,?,?,?,?,?) '''
23     cur = conn.cursor()
24     cur.execute(sql,
25 (ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments))
26     conn.commit()
27     return cur.lastrowid
28
29 # Code below is scraping reviews for Tracer360.
30 urlstring = [LINK TO AMAZON REVIEW PAGE]
31 headers = {'User-agent': 'Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/537.36
32 (KHTML, like Gecko) Chrome/37.0.2062.120 Safari/537.36'}
33
34 req = urllib.request.Request(urlstring, headers=headers)
35 page = urllib.request.urlopen(req)
36
37 #print(page.read())
38
39 pagenumber = 1
40 totalreviews = 0
41 myconnection = create_connection("database/data.db")
42 if myconnection is None:
43     print("could not connect to database. aborting.")
44     sys.exit(1)
45
46 # Using a while loop to scrape all pages with reviews.
```

```
43 while True:
44     print('Processing page: {}'.format(pagenum))
45     soup = BeautifulSoup(page.read(), "html.parser")
46
47     review_list = soup.find("div", {"id": 'cm_cr-review_list'})
48     # Encountered an error when 'cm_cr-review_list' wasn't found, so I consulted a
Python expert who helped me create a condition to try again.
49     if review_list is None:
50         time.sleep(1)
51         req = urllib.request.Request(urlstring, headers=headers)
52         page = urllib.request.urlopen(req)
53
54         print('Processing page: {}'.format(pagenum))
55         soup = BeautifulSoup(page.read(), "html.parser")
56
57         review_list = soup.find("div", {"id": 'cm_cr-review_list'})
58
59     # Here I am extracting the code that contains the list of reviews.
60     reviews = review_list.findChildren(attrs={"data-hook": "review"}, recursive=False)
61
62     # This is the code to stop the loop when all pages have been scraped.
63     if len(reviews) == 0:
64         break
65     for review in reviews:
66         # Initializing variables to none
67         output_id = None
68         output_datespan = None
69         output_starrating = None
70         output_title = None
71         output_body = None
72         output_authorname = None
73         output_authorid = None
74         output_avp = None
75         output_helpful = None
76         output_comment = None
77
78         totalreviews = totalreviews + 1
79
80         # Extracting the ID of each review.
81         output_id = review.get("id")
82         print("id is {}".format(output_id))
83
84         # Extracting the date the review was published, and making sure the object is a
datetime object. This will help with database queries during data analysis.
85         datespans = review.findChildren(attrs={"data-hook": "review-date"},
recursive=True)
86         for datespan in datespans:
87             output_datespan = datespan.text.replace("on ", "")
88             output_datespan = datetime.strptime(output_datespan, "%B %d, %Y")
89             print("date is {}".format(output_datespan))
90
91         # Extracting the star rating, and making sure the object is an integer.
```

```
92     starratings = review.findChildren(attrs={"data-hook": "review-star-rating"},
recursive=True)
93     for starrating in starratings:
94         for classname in starrating.get("class"):
95             if "a-star-" in classname:
96                 output_starrating = int(classname.replace("a-star-", ""))
97                 print("star rating is {}".format(output_starrating))
98
99     # Extracting title of review.
100    titles = review.findChildren(attrs={"data-hook": "review-title"},
recursive=True)
101    for title in titles:
102        output_title = title.text
103        print("title is {}".format(output_title))
104
105    # Extracting body content of review.
106    bodies = review.findChildren(attrs={"data-hook": "review-body"},
recursive=True)
107    for body in bodies:
108
109        # Amazon includes HTML code in the body. I consulted a Python expert to
write code that would remove the HTML. This will allow me to more easily run a bag of
words analysis.
110        bodylist = [item for item in body.contents if isinstance(item, str)]
111        output_body = " ".join(bodylist)
112        print("body is {}".format(output_body))
113
114    # Extracting author of review. I extracted both the username and ID.
115    authors = review.findChildren("a", attrs={"data-hook": "review-author"},
recursive=True)
116    for author in authors:
117        output_authurname = author.text
118        output_authorid = author.get("href").split(".")[1]
119        print("author is {}".format(output_authurname))
120        print("account id is {}".format(output_authorid))
121
122    # This is testing whether or not the review was posted by a "verified
purchaser". Object is a boolean object. This will help with database queries.
123    avps = review.findChildren(attrs={"data-hook": "avp-badge"}, recursive=True)
124    output_avp = True if len(avps) > 0 else False
125    print("verified purchaser: {}".format(output_avp))
126
127    # Extracting the number of people found each review helpful, and making sure
the object is an integer.
128    helpfulstatements = review.findChildren(attrs={"data-hook": "helpful-vote-
statement"}, recursive=True)
129    for helpfulstatement in helpfulstatements:
130        output_helpful = helpfulstatement.text.split(" ")[0]
131
132    # When only one person finds a review helpful, Amazon displays the content
as the word "One". I consulted a Python expert to help add code to handle the scenario
- converting the string "One" to the integer 1.
```

```
133         output_helpful = 1 if output_helpful == "One" else int(output_helpful)
134         print("Number of helpful votes is {}".format(output_helpful))
135
136         # Extracting the number of comments on each review, and making sure the object
is an integer.
137         comments = review.findChildren("span", attrs={"class": "review-comment-total"},
recursive=True)
138         for comment in comments:
139             output_comment = int(comment.text)
140             print("Number of comments is {}".format(output_comment))
141
142             print("-----")
143
144             insert_review(myconnection,
output_id,output_datespan,output_starrating,output_title,output_body,output_authername,
145                 output_authorid,output_avp,output_helpful,output_comment)
146
147             # We pause the scraping for one second. Other coders who have scraped Amazon have
suggested following this best practice so that Amazon doesn't block your code.
148             time.sleep(1)
149
150             # Here I am getting the next page. Once the next page is obtained, it goes back to
the top of the loop where it collects all the above information for reviews on that
page. The .format function replaces the page number embedded within the URL.
151             pagenumber = pagenumber + 1
152             urlstring = [LINK TO AMAZON PAGE].format(pagenumber, pagenumber)
153             req = urllib.request.Request(urlstring, headers=headers)
154             page = urllib.request.urlopen(req)
155
156         myconnection.close()
157         print("Total reviews = {}".format(totalreviews))
```

## **Appendix B**

Below is the complete Python code used to scrape database reviews for Product B.

```
1  import requests
2  # Based on a Google search, the most popular Python package used to scrape web
pages.
3  from bs4 import BeautifulSoup
4  from datetime import datetime
5  import time
6  import sqlite3
7  import sys
8  import urllib.request
9
10 def create_connection(db_file):
11     try:
12         conn = sqlite3.connect(db_file)
13         return conn
14     except Exception as e:
15         print(e)
```



```
16     return None
17 # Code below is inserting the variables into a SQL database
18 def insert_review(conn,
19     ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments):
19     sql = ''' INSERT INTO Reviews[COMPANY NAME]
20     (ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments)
21     VALUES(?,?,?,?,?,?,?,?,?) '''
22     cur = conn.cursor()
23     cur.execute(sql,
24     (ReviewID,Date,Starrating,Title,Body,Username,UserID,Verified,Helpful,Comments))
25     conn.commit()
26     return cur.lastrowid
27
28 # Code below is scraping reviews for Tracer360
29 urlstring = [LINK TO AMAZON PAGE]
30 headers = {'User-agent': 'Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/537.36
31 (KHTML, like Gecko) Chrome/37.0.2062.120 Safari/537.36'}
32
33 req = urllib.request.Request(urlstring, headers=headers)
34 page = urllib.request.urlopen(req)
35
36 #print(page.read())
37
38 pagenumber = 1
39 totalreviews = 0
40 myconnection = create_connection("database/data.db")
41 if myconnection is None:
42     print("could not connect to database. aborting.")
43     sys.exit(1)
44
45 # Using a while loop to scrape all pages with reviews.
46 while True:
47     print('Processing page: {}'.format(pagenumber))
48     soup = BeautifulSoup(page.read(),"html.parser")
49
50     review_list = soup.find("div", {"id": 'cm_cr-review_list'})
51     # Encountered an error when 'cm_cr-review_list' wasn't found, so I consulted a
52     Python expert who helped me create a condition to try again.
53     if review_list is None:
54         time.sleep(1)
55         req = urllib.request.Request(urlstring, headers=headers)
56         page = urllib.request.urlopen(req)
57
58         print('Processing page: {}'.format(pagenumber))
59         soup = BeautifulSoup(page.read(), "html.parser")
60
61         review_list = soup.find("div", {"id": 'cm_cr-review_list'})
62
63     # Here I am extracting the code that contains the list of reviews.
64     reviews = review_list.findChildren(attrs={"data-hook": "review"},
65     recursive=False)
66
```

```
62 # This is the code to stop the loop when all pages have been scraped.
63 if len(reviews) == 0:
64     break
65 for review in reviews:
66     # Initializing variables to none
67     output_id = None
68     output_datespan = None
69     output_starrating = None
70     output_title = None
71     output_body = None
72     output_authorname = None
73     output_authorid = None
74     output_avp = None
75     output_helpful = None
76     output_comment = None
77
78     totalreviews = totalreviews + 1
79
80     # Extracting the ID of each review.
81     output_id = review.get("id")
82     print("id is {}".format(output_id))
83
84     # Extracting the date the review was published, and making sure the object
is a datetime object. This will help with database queries during data analysis.
85     datespans = review.findChildren(attrs={"data-hook": "review-date"},
recursive=True)
86     for datespan in datespans:
87         output_datespan = datespan.text.replace("on ", "")
88         output_datespan = datetime.strptime(output_datespan, "%B %d, %Y")
89         print("date is {}".format(output_datespan))
90
91     # Extracting the star rating, and making sure the object is an integer.
92     starratings = review.findChildren(attrs={"data-hook": "review-star-
rating"}, recursive=True)
93     for starrating in starratings:
94         for classname in starrating.get("class"):
95             if "a-star-" in classname:
96                 output_starrating = int(classname.replace("a-star-", ""))
97                 print("star rating is {}".format(output_starrating))
98
99     # Extracting title of review.
100     titles = review.findChildren(attrs={"data-hook": "review-title"},
recursive=True)
101     for title in titles:
102         output_title = title.text
103         print("title is {}".format(output_title))
104
105     # Extracting body content of review.
106     bodies = review.findChildren(attrs={"data-hook": "review-body"},
recursive=True)
107     for body in bodies:
108
```

```
109         # Amazon includes HTML code in the body. I consulted a Python expert to
write code that would remove the HTML. This will allow me to more easily run a bag
of words analysis.
110         bodylist = [item for item in body.contents if isinstance(item,str)]
111         output_body = " ".join(bodylist)
112         print("body is {}".format(output_body))
113
114         # Extracting author of review. I extracted both the username and ID.
115         authors = review.findChildren("a", attrs={"data-hook": "review-author"},
recursive=True)
116         for author in authors:
117             output_authorname = author.text
118             output_authorid = author.get("href").split(".")[1]
119             print("author is {}".format(output_authorname))
120             print("account id is {}".format(output_authorid))
121
122         # This is testing whether or not the review was posted by a "verified
purchaser". Object is a boolean object. This will help with database queries.
123         avps = review.findChildren(attrs={"data-hook": "avp-badge"},
recursive=True)
124         output_avp = True if len(avps) >0 else False
125         print("verified purchaser: {}".format(output_avp))
126
127         # Extracting the number of people found each review helpful, and making
sure the object is an integer.
128         helpfulstatements = review.findChildren(attrs={"data-hook": "helpful-vote-
statement"}, recursive=True)
129         for helpfulstatement in helpfulstatements:
130             output_helpful = helpfulstatement.text.split(" ")[0]
131
132         # When only one person finds a review helpful, Amazon displays the
content as the word "One". I consulted a Python expert to help add code to handle
the scenario - converting the string "One" to the integer 1.
133         output_helpful = 1 if output_helpful == "One" else int(output_helpful)
134         print("Number of helpful votes is {}".format(output_helpful))
135
136         # Extracting the number of comments on each review, and making sure the
object is an integer.
137         comments = review.findChildren("span", attrs={"class": "review-comment-
total"}, recursive=True)
138         for comment in comments:
139             output_comment = int(comment.text)
140             print("Number of comments is {}".format(output_comment))
141
142         print("-----")
143
144         insert_review(myconnection,
output_id,output_datespan,output_starrating,output_title,output_body,output_authorn
ame,
145             output_authorid,output_avp,output_helpful,output_comment)
146
```

```
147     # We pause the scraping for one second. Other coders who have scraped Amazon
      have suggested following this best practice so that Amazon doesn't block your code.
148     time.sleep(1)
149
150     # Here I am getting the next page. Once the next page is obtained, it goes back
      to the top of the loop where it collects all the above information for reviews on
      that page. The .format function replaces the page number embedded within the URL.
151     pagenumber = pagenumber + 1
152     urlstring = [LINK TO AMAZON PAGE].format(pagenumber, pagenumber)
153     req = urllib.request.Request(urlstring, headers=headers)
154     page = urllib.request.urlopen(req)
155
156     myconnection.close()
157     print("Total reviews = {}".format(totalreviews))
```

## Appendix C

```
#Quality tests

#Checking for duplicates in columns ID, ReviewID, Body, UserID

select {COLUMN NAME}, count(*)
from Reviews
group by {COLUMN NAME}
having count(*)>1

#Returned 0 rows for ID, ReviewID, UserID
#Returned 1 row for Body: One duplicate found in Body, but confirmed to be
seperate reviews by different users.

#Checking for empty strings in ReviewID, Date, Title, Body, Username, UserID

select {COLUMN NAME}
from Reviews
where {COLUMN NAME} = ""

#Returned 0 rows

#Checking to make sure Starrating is between 1 and 5

select starrating
from Reviews
where starrating < 1 or
      starrating > 5

#Returned 0 rows

#Checking for Null in Starrating, Verified, Comments

select {COLUMN NAME}
from Reviews
where {COLUMN NAME} is null

#Returned 0 rows

#Checking for errors in Helpful column (null = 0 people found the review
helpful, not null = returns the number of people who found the review
helpful)

select helpful
from Reviews
where helpful < 1

#Returned 0 rows

#Checking for errors in Verified column (0 = non-verified purchaser, 1 =
verified purchaser)

select verified
from Reviews
```

group by verified

#Returned two rows, 0 and 1

#Checking for errors in Comments column (Comments should be an integer >= 0)

```
select comments
from Reviews
where comments < 0
```

#Returned 0 rows