On the Temporal Dynamics of Opinion Spamming: Case Studies on Yelp

Santosh K C
Department of Computer Science
University of Houston
501 Philip G. Hoffman Hall
Houston, TX 77204-3010
syantokc@gmail.com

Arjun Mukherjee
Department of Computer Science
University of Houston
501 Philip G. Hoffman Hall
Houston, TX 77204-3010
arjun@uh.edu

ABSTRACT
Recently, the problem of opinion spam has been widespread and has attracted a lot of research attention. While the problem has been approached on a variety of dimensions, the temporal dynamics in which opinion spamming operates is unclear. Are there specific spamming policies that spammers employ? What kind of changes happen with respect to the dynamics to the truthful ratings on entities. How do *buffered* spamming operate for entities that need spamming to retain threshold popularity and *reduced* spamming for entities making better success? We analyze these questions in the light of time-series analysis on Yelp. Our analyses discover various temporal patterns and their relationships with the rate at which fake reviews are posted. Building on our analyses, we employ vector autoregression to predict the rate of deception across different spamming policies. Next, we explore the effect of filtered reviews on (long-term and imminent) future rating and popularity prediction of entities. Our results discover novel temporal dynamics of spamming which are intuitive, arguable and also render confidence on Yelp's filtering. Lastly, we leverage our discovered temporal patterns in deception detection. Experimental results on large-scale reviews show the effectiveness of our approach that significantly improves the existing approaches.

Keywords
Opinion Spam; Time Series; Spam Detection.

1. Introduction
The increasing share of the online businesses in market economy has led to a larger influence and importance of the online reviews in purchase and decision making. As positive/negative reviews can either enhance/defame products, the essence of truthful opinions is being misused by deceptive opinion spamming. First reported in [6], opinion spam refers to deliberate attempts (e.g., writing fake reviews, giving unfair ratings) to promote/demote target products/services. Several high-profile cases of fake reviews have been reported in the news. While credit-card fraud is as low as 0.2%, opinion spam is prevalent [28] and it is estimated that up to 20% of online reviews could be fake [34]. The problem has also received significant research attention. Notable works include detecting individual spammers [17], group spammers [22, 23] using a variety of approaches such as rating behaviors [17], unexpected association rules [7], linguistic approaches [5, 16, 29], latent variable models [14, 20], semi-supervised methods [10, 11, 13, 15], etc. Other related works include identifying multiple aliases of the same author (sockpuppet) [30, 31, 33], generic deception detection [27], and deceptive content detection in forums [1, 9]. For a comprehensive survey, see [19].

WWW 2016, April 11-15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4143-1/16/04.
http://dx.doi.org/10.1145/2872427.2883087

While the above works have made important progresses, we still do not know the temporal dynamics that underpin the problem of opinion spam. How does opinion spamming operate on a daily basis? What are the dominant spamming policies? How do the spam injection rates vary upon increased/reduced popularity of entities? What factors are temporally correlated with opinion spamming? How effectively can we predict the long term/imminent future of popularity and average rating of an entity in the presence of deception and how accurately can future deception be predicted?

This papers aims to answer these questions in the light of time-series analysis. We use Yelp as a target of our case-study as it is one of the largest online consumer review hosting site for services (e.g., restaurants, hotels, etc.) in the commercial setting. The closest works to ours were attempted in following researches. Fei et al., (2013) [3] explored temporal burstiness patterns in product reviews for singular spammer detection. The rationale was spammers writing only one review per id (sockpuppets) could be detected as they tend to appear in product review bursts. In [4], rating distributional divergence were used to identify review spam and in [25] a hardness analysis of detection was presented based on real and pseudo fake reviews. Xie et al., [35] investigated temporal burstiness patterns in ratings of online stores. While [24] explored detection using fully unsupervised generative models, in [12], spatio-temporal patterns on the geographical distribution of spammers, were explored using internal data (e.g., IP addresses, cookies, etc.). Although these works have looked into the temporal dimension of spamming, their focuses were mostly on detection as opposed characterizing the very way opinion spamming works. They also did not explore the behaviors which are temporally correlated with spamming and future deception prediction which are the core focuses of this work.

We use the truthful and fake (spam) reviews of popular restaurants in Chicago from Yelp to characterize the dynamics of opinion spamming. We start by analyzing the time-series of fake ratings of each restaurant in our data. We notice similar patterns in the time-series of different restaurants that indicate presence of latent spamming policies/trends likely to be used by spammers. To uncover them, we employ spectral clustering [36] on the time series. Our analyses reveal that there exist three dominant trends of spam injection: *early*, *mid*, and *late* spamming across the restaurants in our data. For each restaurant in each of the three *early*, *mid*, and *late* spamming policies, we jointly characterize the time series of cumulative deceptive ratings with the time series of various other modalities (e.g., truthful like ratings, truthful dislike ratings, truthful review count, etc.). This yields interesting inferences that hint that deceptive like ratings (promotion spamming) is linked with different behavioral modalities of truthful reviews over time and the rating dynamics of truthful reviews can potentially determine the future deception rates for each restaurant.

To validate the relationship, we perform time-series correlation analysis. Cross correlation results show statistically significant correlations of time-series of truthful ratings (as covariate) with future deceptive like ratings (as response) confirming the previous result beyond mere coincidence across each of the three polices. It further reveals two interesting spamming trends: *buffered* and *reduced* spamming which reveal the adaptive spam injection rates for two kinds of restaurants: i) those that need spamming to retain

threshold popularity, ii) others that are more successful and consequently in lesser need of spamming.

Upon characterizing the spamming patterns, we predict future deceptive like ratings on a restaurant using vector auto regression. The predictions being decent, lead us to explore the question – How well can one predict the future truthful popularity (# of reviews) and average rating of a restaurant in the presence of deceptive reviews? Working using lasso regression and vector auto regression we develop models capable of long term and imminent future popularity/rating predictions. The analyses also facilitate indirect validation of Yelp's filtering. Lastly, we leverage the discovered temporal dynamics to devise a suite of novel time-series features. Experimental results show that the time-series features derived from our analyses significantly outperform the existing state-of-the-art approaches for deception detection demonstrating a value of our analysis beyond mere characterization of temporal dynamics.

2. Yelp as a Reference Dataset

Despite opinion spamming being prevalent [28], there are not many commercial websites that filter fake/deceptive reviews. Yelp is an exception and implements review filtering on a commercial scale. The filter has been in place for over a decade now and maintained by its dedicated anti-fraud team [26]. Although Yelp's filter may not be perfect, it is important to note that unlike other forms of generic Web spam (e.g., link [32], email [2], blog[8], etc.) that are relatively easier to detect, opinion spam is harder and usually requires a lot of internal signals [12, 22] and thus industrial opinion spam filters (e.g., Yelp) have a unique advantage. Thus, unlike previous small scale studies in [16, 22, 29], it is not possible to do large-scale analysis upon relying on data tagged by human experts or solicited ground truths fake reviews using Amazon Mechanical Turk. Even expert-annotation cannot fully eliminate the possibility of any noise. Also obtaining ground truths in the given domain and commercial setting is only possible upon spammer confessions or sting operations [18] which again cannot be performed at large scale. Thus, Yelp's filter although may not be perfect, nevertheless provides us a unique opportunity to understand the dynamics of spamming at large-scale in the commercial setting. In fact, there have been researches that put Yelp's filtering methods to test and have found it to be reasonably reliable [18, 26]. Hence, we choose Yelp as a reference dataset for characterizing the dynamics of opinions spamming.

As demonstrated by our experiments, we will see that the spamming patterns discovered are arguable, intuitive, and further pave the way for indirectly validating Yelp's filtering.

We use the Yelp dataset in [26] of 70 popular Chicago restaurants over a 5 year time span (see Table 1). The reviews filtered by Yelp are considered deceptive (fake/spam) while others as truthful. We refer to reviews with 1-3★ ratings as exhibiting "dislike" whereas reviews with 4-5★ ratings exhibiting "like" connotations. Although opinion spamming can take both promotion/demotion flavors (by injecting deceptive like/dislike reviews), our pilot studies revealed that majority (~75%) of the spam is focused on promotion as opposed to demotion. Hence we focus on promotion spamming. The next section lays the foundation for analyzing the dynamics of promotion spamming.

3. Determining Dominant Spamming Policies

Although opinion spamming can be interleaved throughout the entire lifespan of an entity, characterizing the dominant spamming patterns over time is the first step in understanding the dynamics of spamming. To find the number of promotion spamming policies, for each restaurant, we compute its time series of average cumulative rating of deceptive like (positive fake) reviews. The cumulative rating was computed for each time step by averaging the like fake ratings on that restaurant from start till that time step. The time series was further normalized and scaled to the range [0, 1] to gauge the relative promotion dynamics and facilitate time-series clustering on shape dynamics. The rationale here is that such a cumulative deceptive rating time series can quantify how spamming grew and faded over time for that restaurant. For each restaurant, its time-series starts at the date of the first review and continues until 60 months from the start. Time-series of all restaurants were aligned by pivoting on their starting time-step.

Table 1. Dataset Statistics

	Deceptive	Truthful
# of dislike (1-3★) reviews	1630	10042
# of like (4-5★) reviews	4465	30652
# of reviews	6095	40694
% of reviews	13.03%	86.97%
# of reviewers	5359	21761

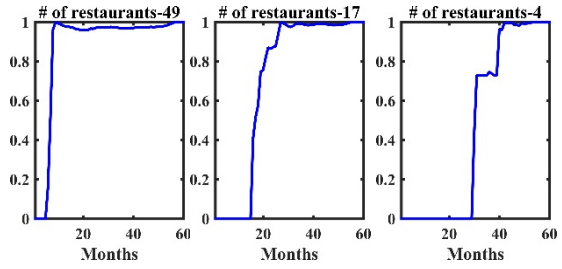
We hypothesize that there exist commonalities in spamming trends (policies) that exist across different restaurants. To characterize these spamming policies, we employed time-series clustering that can discover similar shapes in the deceptive rating time series of restaurants. We used the K-spectral Centroid (K-SC) time-series clustering algorithm in [36]. The distance function of K-SC is invariant to scaling and translation which is particularly suited to our domain in capturing similarities in spamming across restaurants with varying popularity (review volume) and different launch dates. Its distance measure, $d(x, y)$ for two time-series x, y is calculated as:

$$d(x, y) = min_{\alpha,q} \frac{||x - \alpha y_{(q)}||}{||x||} \qquad (1)$$

where $y_{(q)}$ is the result of shifting time series y by q time units, $||.||$ is the L_2 norm and α is the scaling coefficient to match the shape of two time series. Apart from clustering time-series having similar temporal patterns, it also yields the cluster centroid time-series for each cluster that is representative of that cluster.

We clustered the deceptive like rating time-series of restaurants in our data using the K-SC algorithm. As the dominant number of spamming policies are unknown, we explored different values for K. Fig. 1, 2, and 3 show the time-series centroid plots for $K = 3, 4, 6$ respectively. The centroid plot header also reports the # of restaurants for that cluster. We note that for $K = 6$, cluster 6 (Fig. 3.f) is empty. Out of the remaining five clusters for $K = 6$, cluster 1 (Fig. 3.a) has similar shape to cluster 1 (Fig. 2.a) in $K = 4$ and cluster 1 (Fig. 1.a) in $K = 3$ and all three have the same 49 restaurants. Cluster 2 (Fig. 3.b) in $K = 6$ and cluster 2 (Fig. 2.b) in $K = 4$ are quite similar due to their starting spikes and profile. Cluster 3 (Fig. 3.c) in $K = 6$ is similar to cluster 3 (Fig. 2.c) as both plummet in the right end and have similar starting spikes. Further, cluster 5 (Fig 3.e) in $K = 6$ and cluster 4 in $K = 4$ (Fig. 2.d) also have similar profiles. We see that the similar profiles are getting merged as K is reduced. Cluster 1 is same across all three values of K. Fifteen cluster 2 and two cluster 3 restaurants in $K = 6$ and $K = 4$ are clustered in cluster 2 in $K = 3$. Remaining restaurants from $K = 6$ (two from cluster 3 and one each from cluster 4, 5) and $K = 4$ (three from cluster 3 and one from cluster 4) are merged to cluster 3 in $K = 3$. Thus, we clearly see that there are three dominant promotion spamming policies corresponding to representative cluster centroids. We now explain each spamming policy using the plots for $K = 3$.

Cluster 1 of $K = 3$ (Fig. 1.a) refers to *early spamming* where the representative centroid shows steady spamming beyond the 5 months of launch. Although centroid has a zero till the fifth month, the deceptive like reviews of restaurants in the early spam cluster gradually build up a momentum from their inception. They tend to maintain continuous spamming until the end as depicted by the profile of cluster 1. The second cluster centroid (Fig. 1.b) refers to *mid spamming* policy where spamming is a bit delayed and starts rising only after the 14th month (after more than a year). On average, it takes about 10 more months to have the peak rating of 1 after gradual improvement in spamming. The third spamming policy (Fig. 1.c) starts rather late, stalls for 10 months before attaining the peak in deceptive like ratings. Only few restaurants exhibited such *late spamming*. Thus, we find three dominant

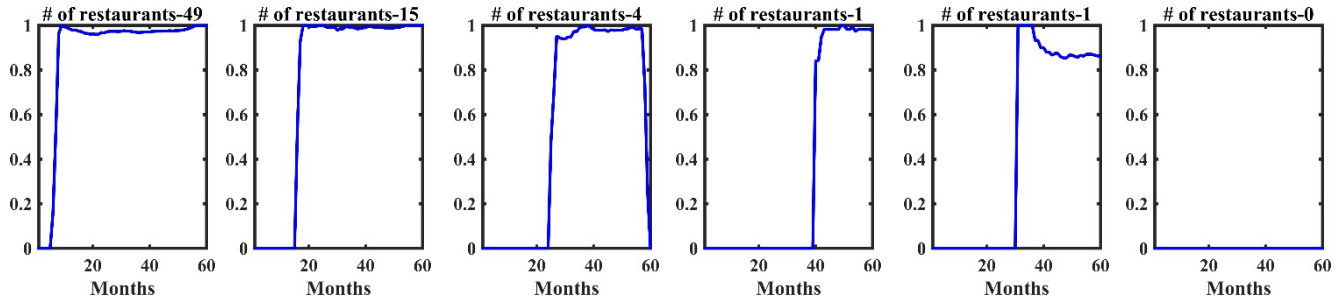


a) Cluster 1 b) Cluster 2 c) Cluster 3

Figure 1. Time series cluster centroids with $K = 3$ for cumulative rating of deceptive like reviews

a) Cluster 1 b) Cluster 2 c) Cluster 3 d) Cluster 4

Figure 2. Time series cluster centroids with $K = 4$ for cumulative rating of deceptive like reviews



a) Cluster 1 b) Cluster 2 c) Cluster 3 d) Cluster 4 e) Cluster 5 f) Cluster 6

Figure 3. Time series cluster centroids with $K = 6$ for cumulative rating of deceptive like reviews

spamming policies prevalent in restaurant promotion. The next section evaluates each spamming policy by assessing it in tandem with other behavioral modalities.

4. Dynamics of Spamming Policies

To study the dynamics of the three promotion spamming policies, we generated the time series of the ten modalities in Table 2. For each of the three spamming policies, we grouped all restaurants belonging to a policy and computed an additional set of normalized time-series on the ten modalities in Table 2. For each behavioral modality, we further employed time-series clustering of restaurants in a given policy (cluster) and chose the dominant sub-cluster of that modality in a given policy. We now explain the three spamming policies based on the centroids of the dominant sub-clusters of relevant behavioral modalities for a policy.

4.1 Early Spamming

Fig. 4.a shows the reference centroid plot for this pattern (Fig. 1.a) where 49 out of 70 restaurants employ this policy. The restaurants employing early spamming wait for the truthful reviews for the initial period of around five months. Then they start spamming as shown in Fig. 4.a. It is interesting to note that the average truthful rating is seen rapidly dropping in the initial months (Fig. 4.b.) There is also a rapid increase in the truthful dislike rating (Fig. 4.c) and

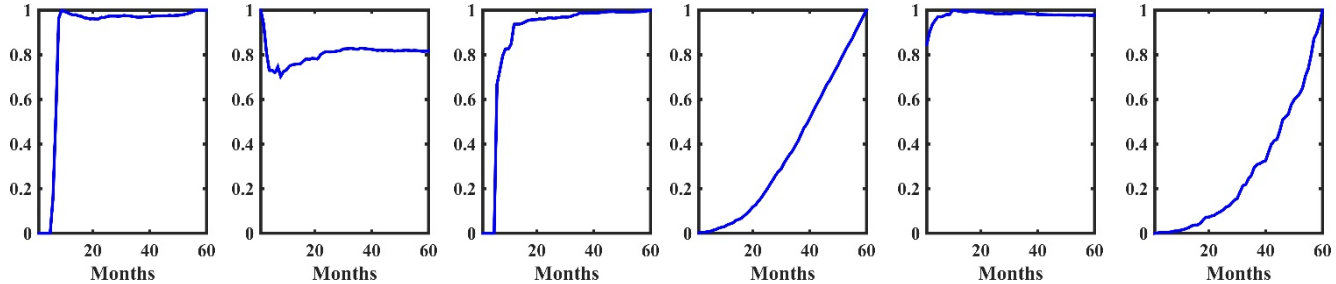
Table 2. Additional Behavioral Modalities for Evaluation

# of fake dislike reviews	Cumulative rating of fake dislike reviews
# of non-fake dislike reviews	Cumulative rating of non-fake dislike reviews
# of fake like reviews	Cumulative rating of fake like reviews
# of non-fake like reviews	Cumulative rating of non-fake like reviews
Cumulative rating of n-fake reviews	Cumulative rating of non-fake reviews

increase in the count of such dislike reviews (Fig. 4.d) till the fifth month. Though, the truthful like rating is constant as shown in Fig. 4.e, the restaurants inject more spam to check the influence of the truthful dislikes. This explains the sustained rate of deceptive like fake ratings (Fig. 4.a) with increase in the count of deceptive like reviews (Fig. 4.f). Thus, in early spamming, the influx of deceptive reviews starts early and maintains a steady promotion spamming rate to balance the truthful dislike influence.

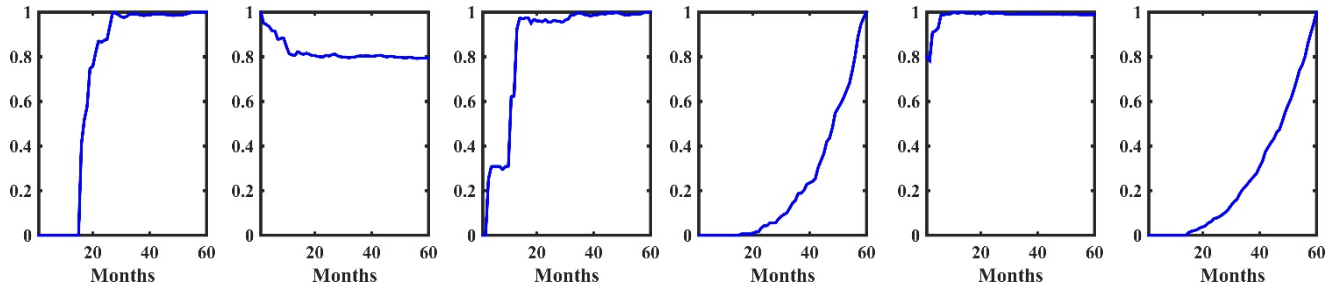
4.2 Mid Spamming

We find 17 restaurants employing mid spamming as shown by the reference centroid plot in Fig. 5.a. These restaurants don't exhibit spam injection until the 14th month. However, truthful average rating keeps on dropping rapidly till about 11th month (Fig. 5.b). It is worth noting that spamming picks up momentum after 14th month (Fig. 5.a). In the same time, truthful dislike rating rapidly increases from 10th month onward (Fig. 5.c) along with the increase in the number of truthful dislike reviews (Fig. 5.d). The truthful like rating



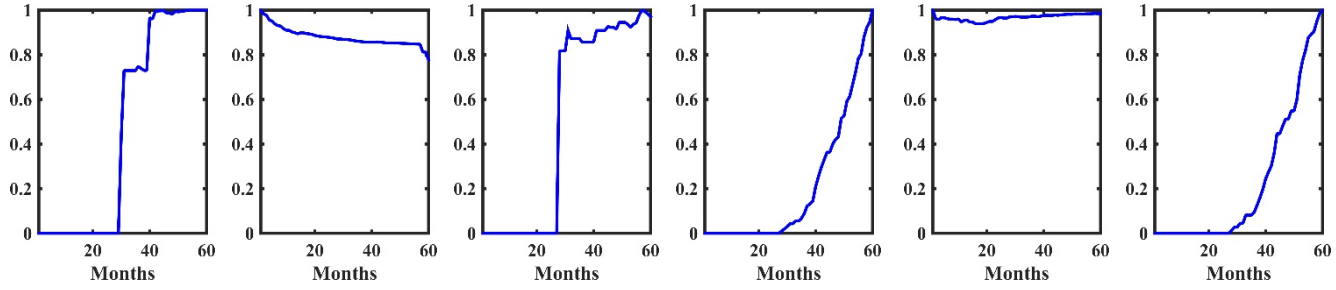
a) Early spamming b) Truthful average rating c) Truthful dislike rating d) # of truthful dislike e) Truthful like rating f)# of deceptive like

Figure 4. Normalized average cumulative rating and review count (#) of early spamming policy



a) Mid Spamming b) Truthful average rating c) Truthful dislike rating d) # of truthful dislike e) Truthful like rating f)# of deceptive like

Figure 5. Normalized average cumulative rating and review count (#) of mid spamming policy



a) Late spamming b) Truthful average rating c) Truthful dislike rating d) # of truthful dislike e) Truthful like rating f)# of deceptive like

Figure 6. Normalized average cumulative rating and review count (#) of late spamming policy

is not affected (Fig. 5.e). So, the number of deceptive like ratings increases after 14th month (Fig. 5.f) and as if counteracts the increase in the truthful dislike ratings (Fig. 5.c). This clearly shows that deceptive like rating (spam injection) is almost in tandem with truthful average and dislike ratings.

4.3 Late Spamming

The late spamming pattern was found in 4 restaurants only. Fig. 6.a shows the reference centroid plot. These restaurants start promotion spamming only after the 30th month (Fig 6.a). Interestingly, truthful average rating (Fig 6.b) is seen monotonically decreasing. After 28th month, there is rapid increase in the truthful dislike rating (Fig. 6.c) caused by the soaring of truthful dislike reviews (Fig. 6.d). Since, the truthful like rating rate does not differ much (Fig 6.e), promotion spamming seems to be carried out after 30th month (Fig 6.f) to check the influx of the truthful dislike reviews. We also note that after a slight decrease in truthful dislike rating, it increases again after 40th month (Fig. 6.c). Interestingly enough, we find the restaurants increase spamming after 40th month (Fig. 6.a) as if to bring an equilibrium with the dislike ratings spike. This phenomenon of increasing spam injection being tightly connected with the dynamics of dislike ratings across all policies can be inferred as *buffered spamming*.

5. Causal Modeling of Deceptive Ratings

In this section, we aim to characterize the plausible causes of spamming by comparing the time series of deceptive like ratings with the truthful like/dislike ratings. We use week as time interval for the time series in this section. We first explore the potential causes that are forerunners of deceptive ratings using cross correlation. Next, we encode the potential causal time-series in a vector autoregressive framework to forecast future deceptive ratings.

5.1 Time Series Causal Analysis Framework

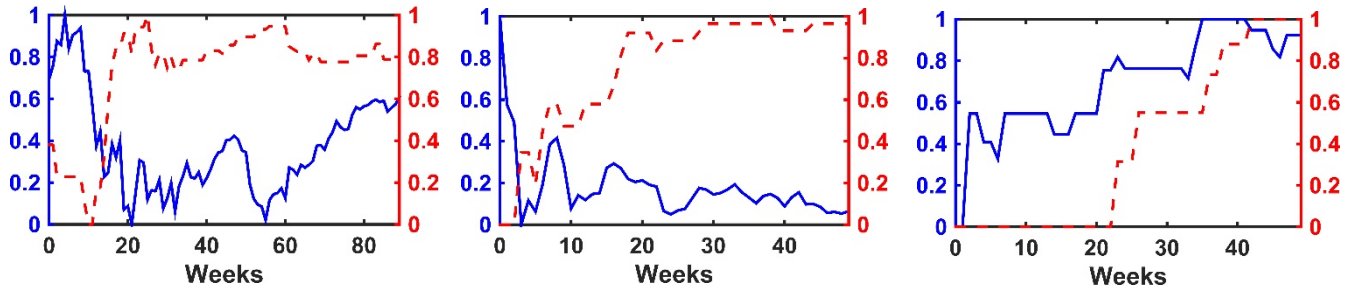
From the spamming dynamics explored in previous sections, it reflects the intuition that variations in truthful review ratings has a certain influence in the dynamics of deceptive like ratings. To discover the relationship between truthful ratings and deceptive like ratings (promotion spams) of restaurants, we consider their respective cumulative time-series. To validate the relationship, we extracted three segments of truthful ratings time-series (overall truthful average, truthful like, and truthful dislike) and compared them against the deceptive like rating time series of individual restaurants. To discover potential causality, we analyzed their cross-correlation (CCF) at different time lags. CCF at lag k estimates the relationship between a response $Y(t)$, and a covariate $X(t)$ time-series at different time-steps shifted by k time units and is given by:

$$CCF(k) = \frac{\sum_i((X(i)-\mu_X)(Y(i+k)-\mu_Y))}{\sqrt{\sum_i(X(i)-\mu_X)^2}\sqrt{\sum_i(Y(i+k)-\mu_Y)^2}} \qquad (2)$$

Correlation at a positive lags implies that X is a good predictor of Y and positive/negative CCF values indicate the changes in the series X and Y are in the same/opposite directions respectively.

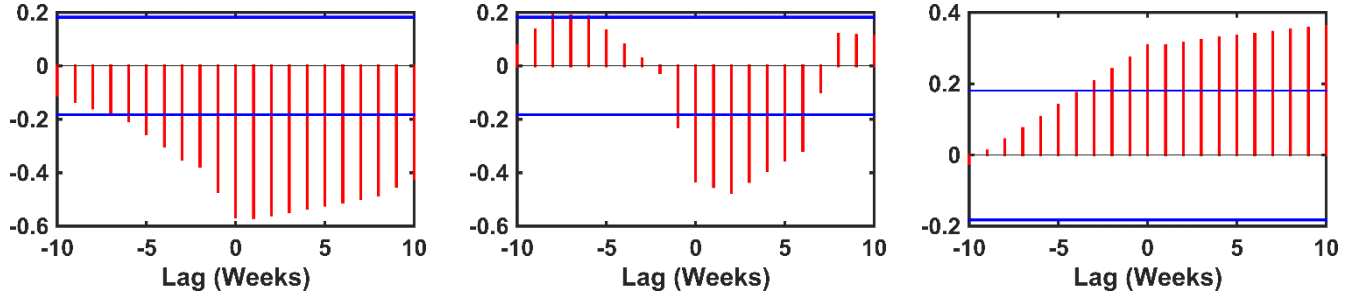
5.2 Buffered Spamming

How do restaurants deal with their weaning popularity and growth of dislike ratings? Do they proactively inject deceptive reviews to maintain threshold average rating/popularity or to lessen the impact of truthful dislike reviews? To analyze the impact of truthful ratings on the rate of deceptive like ratings, we compare the time-series of truthful average rating, truthful like rating and truthful dislike rating (as covariates) with the deceptive like time-series (as the response). From Fig. 7, we note that the time-series for deceptive like ratings is shifted in future and increases with decrease in truthful average rating (Fig. 7.a) and truthful like rating (Fig. 7.b). Further, the increase in truthful dislike ratings tends to also cause increase in deceptive like ratings (Fig. 7.c). It is also interesting to note that the changes in the response time-series ($Y\sim$ deceptive like ratings) are tightly connected with the covariate time-series (e.g., steep drops in truthful avg. ratings from week 9 follows a rise in deceptive like ratings from week 11 in Fig 7.a, gradual decrease of truthful like ratings from week 15 almost co-occur with steady increase in deceptive like ratings in Fig 7.b, increase in truthful dislike in the first 20 weeks follow an increase of deceptive like in Fig 7.c). These patterns tend to indicate a causality beyond mere coincidence. It is as if there is a "buffer" action at work which adjusts the spamming



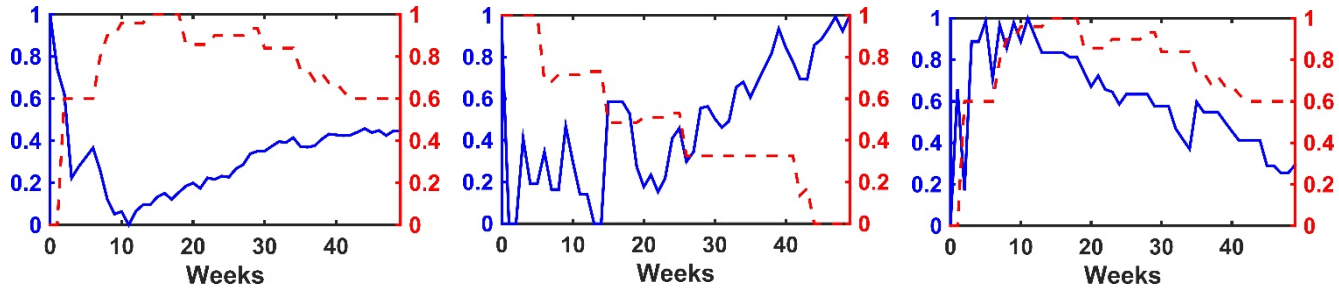
a) Truthful Average Rating b) Truthful Like Rating c) Truthful Dislike Rating

Figure 7. Buffered Spamming - Time series of truthful ratings (solid blue) vs deceptive like rating (dashed red) for different representative restaurants. Representative restaurants refers to the ones where the behavior was most prominent.



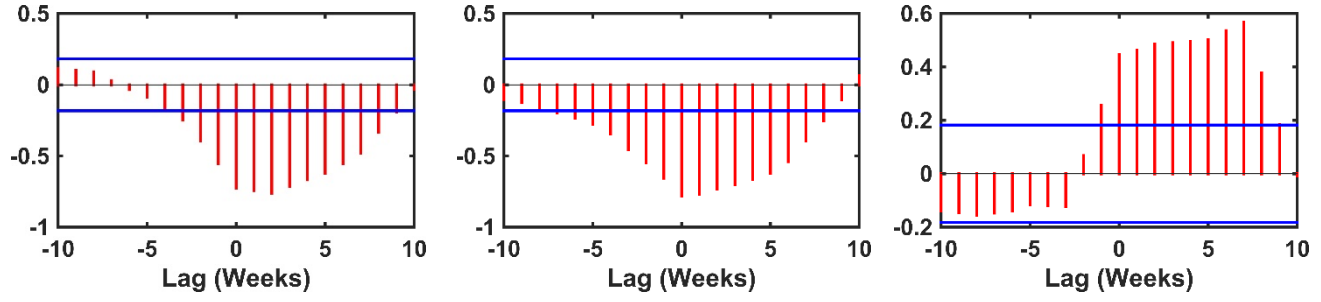
a) Truthful Average Rating b) Truthful Like Rating c) Truthful Dislike Rating

Figure 8. Buffered Spamming – *CCF* plots for respective time-series in Figure 7 for representative restaurants. Red lines indicate the *CCF* value and blue lines indicate the confidence interval bounds at 99% (p<0.01) confidence.



a) Truthful Average Rating b) Truthful Like Rating c) Truthful Dislike Rating

Figure 9. Reduced Spamming - Time series of three truthful ratings (solid blue) vs series of deceptive like rating (dashed red) for different representative restaurants

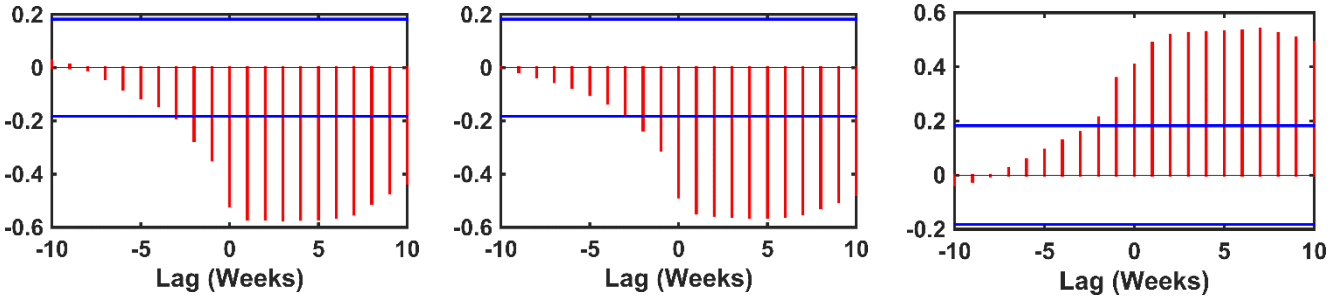


a) Truthful Average Rating b) Truthful Like Rating c) Truthful Dislike Rating

Figure 10. Reduced Spamming – CCF plots for respective time series in Figure 9 for representative restaurants. Red lines indicate the *CCF* value and blue lines indicate the confidence interval bounds at 99% (p<0.01) confidence.

rate by injecting deceptive like reviews as the truthful average and like ratings decrease or truthful dislike ratings increase. Hence, we refer to this spamming pattern as *buffered spamming*.

To further confirm and quantify the strength of the correlation, we computed the *CCF* plots (Fig. 8) of the covariate and response time-series corresponding to Fig 7. We note that for all potential causalities, the *CCF* values exceed the 99% confidence interval bounds indicating statistically significant correlations. Further, all the correlations exhibit positive lags (in the range of [0, 5] weeks) indicating that truthful rating influences the future rate of deceptive like ratings. Negative correlations in Fig. 8 a, b explain the fact that the spamming increases when the average truthful rating and like truthful rating decrease. Positive correlations in Fig. 8.c explain the increase in deceptive like ratings with increase in truthful dislikes.



a) Truthful Average Rating b) Truthful Like Rating c) Truthful Dislike Rating

Figure 11. Average *CCF* Plot. Red lines indicate the *CCF* value and blue lines indicate the confidence interval bounds obtained 99% confidence (p<0.01).

These results tend to confirm the buffered nature of spamming in Fig. 7 which is an attempt of self-promotion via deceptive like reviews when the truthful reviews are not in their favor.

5.3 Reduced Spamming

We now explore the case when restaurant maintain decent popularity and rating. Is there a reduction in the spam injection rate, as they have a better standing already? We again analyze the impact of truthful ratings on deceptive like ratings. In Fig. 9, we see that the time series for deceptive like rating is shifted in future and now decreases with the increase in truthful average rating (Fig. 9.a) and truthful like rating (Fig. 9.b). Moreover, the decrease in the truthful dislike rating causes the deceptive like rating to decrease (Fig 9.c). Interestingly, here also the changes in the response time series ($Y\sim$ deceptive like ratings) are tightly connected with the co-variate time series (e.g., gradual increase in truthful avg. rating from week 11 is followed by drop in deceptive like rating from week 18 in Fig. 9.a, rapid increase in truthful like rating from week 14 to 15 co-occur with rapid decrease in deceptive like rating in Fig. 9.b, gradual decrease in truthful dislike rating after week 11 is followed by gradual decrease after week 18 in deceptive like rating in Fig 9.c). These trends show a pattern where spam injection rates are reduced when the truthful reviews are favor. Thus, we refer this temporal dynamics as *reduced spamming*.

For significance testing, we computed the *CCF* plots (Fig. 10) of the covariate and response time series in Fig. 9. The *CCF* values exceed the 99% confidence interval bounds for all potential causalities indicating statistically significant correlations. Further, the positive lags (in the range of [0, 5] weeks) indicate that truthful ratings influence the future rate of deceptive like rating. Negative correlations in Fig. 10 a, b explain the fact that the spamming decreases when the average truthful rating and like truthful rating increase. Positive correlations in Fig. 10.c explain the decrease in deceptive like ratings with decrease in truthful dislikes.

5.4 Average Cross Correlation

The above results although establish a decent confidence in causality, they were based on individual restaurants. To ascertain whether these patterns are prevalent, we evaluated their trend in all restaurants. Time series of all the restaurants cannot be shown as average as different restaurants have the buffered and spamming trend at different instance of time. However, it is important to note that since the lags and directions of correlations of buffered and reduced spamming share the same trend (see Fig. 8, 10), it is sufficient to explore the average *CCF* values over all the restaurants. The average *CCF* plots have been shown in Fig. 11 which strengthen our conclusion that the three segments of truthful rating time series are good predictors of deceptive like rating time-series as there exist statistically significant correlations at positive lags. We also see that the average CCF over all restaurants have small yet significant correlation at lag of -1, -2 weeks. This may be due to prognostic behavior of the restaurants where they can sense an imminent consumer dissatisfaction and thus begin spamming beforehand to maintain their ratings.

Table 3. Mean Absolute Error for different training window and different spamming policies and trend for deceptive rating prediction.

	Early		Mid		Late		Buffered		Reduced	
Training	**lag, p**		**lag, p**		**lag, p**		**lag, p**		**lag, p**	
Window	**1**	**2**	**1**	**2**	**1**	**2**	**1**	**2**	**1**	**2**
15 weeks	0.55	0.42	0.28	0.22	0.22	.12	0.33	0.20	0.23	0.18
30 weeks	0.17	0.16	0.12	0.10	0.12	0.04	0.16	0.15	0.08	0.07
45 weeks	0.11	0.11	0.06	0.05	0.03	0.03	0.14	0.12	0.06	0.04
Avg.	0.28	0.23	0.15	0.12	0.12	0.06	0.21	0.16	0.12	0.10

6. Predicting Dynamics of Deceptive Ratings

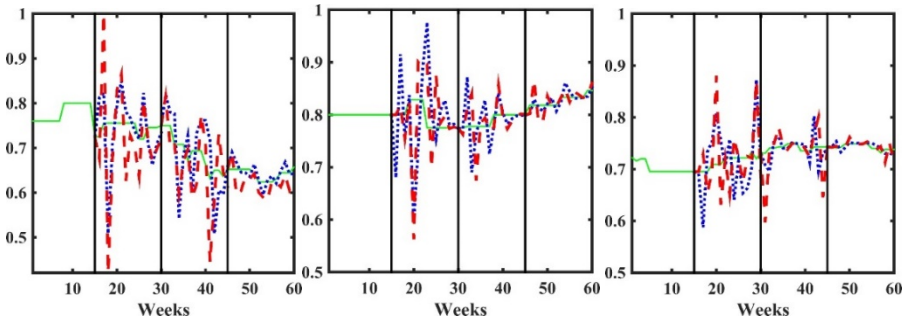
The preceding analysis shows that the dynamics of truthful ratings are harbingers of deceptive like rating. Naturally this raises the research question – Can we predict the dynamics of deceptive like ratings? This section employs vector auto regression to predict deceptive like ratings on restaurants using the time-series of the three truthful ratings.

Let y_t denote a $n \times 1$ vector of n time series variables. A p-lag vector auto regression $VAR(p)$ model takes the form:

$$y_t = a + \sum_{i=1}^{p} A_i y_{t-i} + \varepsilon_t \qquad (2)$$

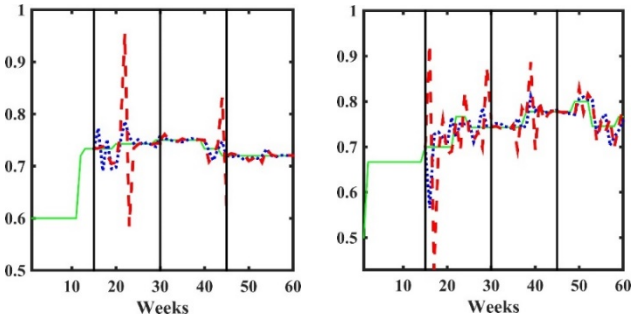
where a is a bias vector of offsets with n elements, A_i are $n \times n$ autoregressive matrices and ε_t is an $n \times 1$ vector of serially uncorrelated innovations (error terms). Training a VAR model entails fitting multiple time-series and parameter estimation via maximum likelihood estimators. Upon parameter learning, values in y_{t+1} are predicted using values of y_{t-W+1} where W is the width of the moving window.

We employed a 4 dimensional VAR with the deceptive like rating as the response time-series and truthful average rating, truthful like rating and truthful dislike rating as covariate time-series. For this analysis, we consider moving window based forecasting of deceptive like ratings for the first 60 weeks (~ 1 year). We trained the model for each restaurants at lags 1 and 2, predicted the next week's deceptive like rating. We experimented with three different training window widths $W = 15, 30, 45$ weeks. For the rating prediction from 16th week to 30th week, model with training window of 15 weeks was used. For the rating prediction (see Fig. 12) from 31st to 45th weeks, the model with training window of 30 weeks was used and for the rating prediction from 46th to 60th weeks, the model with 45 weeks training window was used. So, for example to predict the rating of 21st week, the training data of 6th to 20 weeks was used whereas to predict the rating of 41st week, the training data of 11th to 40th week was used. Thus, the window is moved each time to include the number of weeks specified by the window length. Fig. 12, 13 show the deceptive like time series forecast using p-lag VARs ($p = 1, 2$ weeks) across



a) Early Spamming b) Mid Spamming c) Late Spamming

Figure 12. Deceptive like rating prediction of the next week using VAR Model. Forecasting was done using two p-lag VARs: dotted blue refers to p=1 week lag model; red dashed refers to p=2 week lag predictor. Solid green line represents the actual rating.



a) Buffered Spamming b) Reduced Spamming

Figure 13. Deceptive like rating prediction of the next week using VAR Model. Forecasting was done using two p-lag VARs: dotted blue refers to p=1 week lag model; red dashed refers to p=2 week lag predictor. Solid green line represents the actual rating.

Table 5. MAE for truthful popularity regression

	Early	Mid	Late
NTF	3.94	2.02	1.52
NTF+OL	3.88	2.01	1.49
NTF+OL+NG	3.78	1.99	1.29
NTF+OL+NG+ASL	3.27	1.80	0.92

Table 6. MAE for truthful average rating regression

	Early	Mid	Late
NTF	0.47	0.38	0.16
NTF+OL	0.44	0.30	0.15
NTF+OL+NG	0.36	0.29	0.14
NTF+OL+NG+ASL	0.30	0.28	0.13

Table 4. Non-text features

Weekly rating of each week of the training period
Overall rating of the training period
Friend count of the top 10 reviewer
Total Friend count of all the reviewers
Average review per user
Average review length
Rating deviation in each week of the training period
Overall rating deviation in the training period
Funny count in the reviews
Cool count in the reviews
Parking type Boolean features (street, private lot, garage, valet, validated, on-site)
Attire type Boolean features (casual, dressy, formal)
Ambience Boolean features (casual, intimate, classy, touristy, trendy, upmarket, hipster, upscale, divey, romantic)
Restaurant specific Boolean features (good for kids, accepts credit cards, good for groups, price range, reservations, delivery, takeout, waiter service, outdoor seating, Wi-Fi available, , good for lunch, dinner, desert, late night or breakfast, alcohol and wine, bar, noise level, cuisine available and wheelchair accessibility)
Miscellaneous characters like (?, smileys, !)

three training windows for early; mid; late spamming policies, buffered spamming and reduced spamming for the representative restaurants in each policy (based on closeness to cluster centroid). Table 3 reports the respective MAEs averaged over all restaurants in each policy.

From Fig. 12, 13 and Table 3, we note the following observations. Across all policies, the forecasts are decent improve with longer training windows. Consequently, MAEs are higher for 15 week training windows than 45 week windows across all spamming policies as longer training windows helps learn the complexities of deceptive rating dynamics. Also, the VAR model with lag = 2 weeks performs better than lag 1 as it can leverage the context of an additional previous time unit.

Comparing the MAE across the three policies, we see that early spamming has the highest while late spamming has the lowest (see Table 3 and Fig. 12) indicating predicting deception in early spamming is more difficult compared to mid and late spamming. One reason for this could be that early spamming starts earlier, but their spams are also caught earlier. This can prompt spammers to devise newer and more complex ways of deception and altering their deception rate to avoid being filtered. This can result in a higher change rate in the deceptive ratings in early spamming (e.g., see Fig. 12.a) making it harder to predict.

We found that buffered/reduced spamming trends percolated across all early, mid and late spamming policies. So, the MAE reported in Table 3 for buffered and reduced spamming have been contributed by all types of restaurants. This is the reason why the average values of MAE for buffered (0.21, 0.16) and reduced spamming (0.12, 0.10) are those between the MAE of late (0.12, 0.06) and early (0.28, 0.23) spamming policies for both lag = 1, 2 VARs (see Table 3 last row). Upon VAR forecasting, we see that predicting buffered is harder than reduced (Fig. 13). This is because in buffered spamming, the reviews are injected via a counter buffer action (Fig. 7) making the changes in cumulative deceptive like rates higher (Fig. 13.a) thereby making forecasting harder as the spam injection rates are changing frequently. On the other hand, the deceptive reviews are decreasing in reduced spamming resulting in relatively easier forecasts.

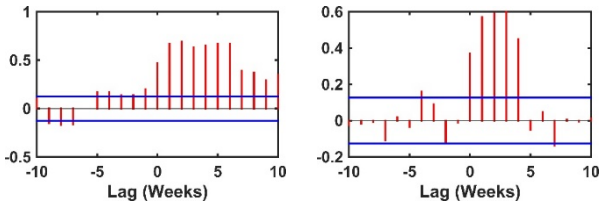
7. Predicting Truthful Popularity and Rating

Do deceptive reviews affect a restaurant's popularity and average ratings? In order to answer this, we need a prediction framework that can forecast the popularity and average rating of a restaurant gained from truthful reviews. Popularity here refers to the total number of reviews in a time period. We studied factors that govern the truthful average rating and popularity of the restaurants 6 months in future and trained lasso regression models for future popularity/rating predictors. The response variables for popularity and rating predictors used the truthful average review rating and total number of reviews beyond 6 months of the start date for model building. Regression models were trained on truthful reviews of the first 10 weeks from the start date and used the following four feature families:

NTF: Non text features shown in Table 4
OL: Opinion lexicon of positive/negative words
NG: Word n-grams ($n = 1, 2$)
ASL: Restaurant domain specific Aspect/Sentiment Lexicon obtained by fitting the model in [21] to our data.

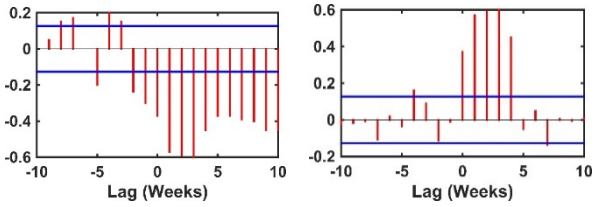
The start date for this experiment was set to the month where fake reviews started accumulating for each policy (see Fig. 4, 5, 6) for setting a comparison reference for the subsequent experiments. Table 5, 6 report the MAEs of prediction of popularity and average rating for restaurants in each policy using 10-fold cross-validation (over all restaurants in each policy).

We employed forward feature selection of four feature families incrementally adding each family in the order NTF, OL, NG and ASL. From Table 5, 6, we note that across all three policies, we that



a) Total # of friends of reviewers b) # of reviewers

Figure 14. Truthful Popularity CCF Plot. Red lines indicate the CCF value and blue lines indicate the confidence interval bounds obtained 99% confidence ($p < 0.01$).



a) Dislike Count b) # of reviewers with 5+ reviews

Figure 15. Truthful Average Rating CCF Plot. Red lines indicate the CCF value and blue lines indicate the confidence interval bounds obtained 99% confidence ($p < 0.01$).

the regression model performs better as we continue adding the features OL, NG and ASL respectively in both popularity and rating prediction showing that natural language signals were helpful. This could be argued by the fact that truthful review contents invariably leave sentiment signals that eventually contribute to the truthful popularity and average ratings of a restaurant. It is important to note that the MAEs for this task is higher in early spamming than mid/late spamming policies. One reason for this could be that the effect of early spamming altered the actual popularity/average rating response for restaurant in the policy that the regression model could not pick (as it was trained using truthful reviews).

7.1 How reliable are Yelp's filtered reviews?

Ideally, deceptive reviews are injected for spamming and are not grounded on true experience, thereby regarded as fake. Hence, the information contained in fake reviews should be detrimental in predicting the future popularity or average rating of a restaurant. In other words, if we have a hypothetical regression oracle (a perfect guesser/ideal solver) for predicting the future truthful popularity and average rating of a restaurant trained on truthful reviews alone, then upon adding fake reviews to the training data for the regression oracle, the error on predicting the response should increase because the oracle has a noise in its training (imparted by fake reviews). This result can be a basis to indirectly validate Yelp's filtering as follows.

Although we don't have a regression oracle for rating/popularity predictor, the regression models trained in Table 5, 6 are of high quality as the MAE of popularity is in the range of roughly $[0, 4]$ reviews (given median popularity as 65 reviews) and the MAE of the average rating lies in the range of $[0.13, 0.4]$ on a normalized average rating scale of [0, 1]. Hence, the regressors can be used as basis for testing the quality of Yelp's filtering. We again trained popularity/rating regressors (using the same settings as in §7) with the full feature set NTF+OL+NG+ASL but also added Yelp's filtered reviews along with truthful reviews in the training set (i.e., used all reviews in training). Table 7 and Table 8, report the MAEs for popularity/rating regressors across all three policies. We note a statistically significant ($p < 0.03$) increase in MAE upon adding review filtered by Yelp across all policies (see the "All reviews" row in Table 7, 8).

Statistically significant increase in the MAEs of the regression models upon changing the training set renders a high confidence that the altered training set imparts a considerable noise i.e., the result implies that the reviews filtered by Yelp imparted noise and were actually harmful to popularity/rating prediction. In other words, the reviews which imparted the noise in the popularity/rating models were not representative of truthful experiences (or potentially fake) and Yelp filtered those reviews. This indirectly raises confidence that Yelp's filter although may not be perfect is reasonably reliable.

Table 7. MAE comparison for popularity regressor

	Early	Mid	Late
Truthful only	3.27	1.80	0.92
All reviews	3.97	2.38	1.16

Table 8. MAE comparison for rating regressor

	Early	Mid	Late
Truthful only	0.30	0.28	0.13
All reviews	0.37	0.35	0.24

Table 9. Time series features for VAR model

Typed day index of time-step t : -1 for Mon/Tue, 0 for Wed/Thu, and +1 for Fri/Sat/Sun.
Total # of friends of all the reviewers in time-step t
of distinct reviewers in time-step t
of reviewers posting ≥ 5 reviews within $[-\infty, t]$
of reviewers outside of the Chicago area in time-step t
Standard deviation of the rating of the reviews in time-step t
of dislike reviews in time-step t
of like reviews in time-step t
of +ve lexicon words normalized by review length in time-step t
of -ve lexicon words normalized by review length in time-step t
of words (lol, !, ?, etc) normalized by review length in time-step t
of restaurant specific aspect sentiment words normalized by review length in time-step t

Table 10. Mean Absolute Error for next time-step's truthful popularity (# of reviews) prediction using VAR model

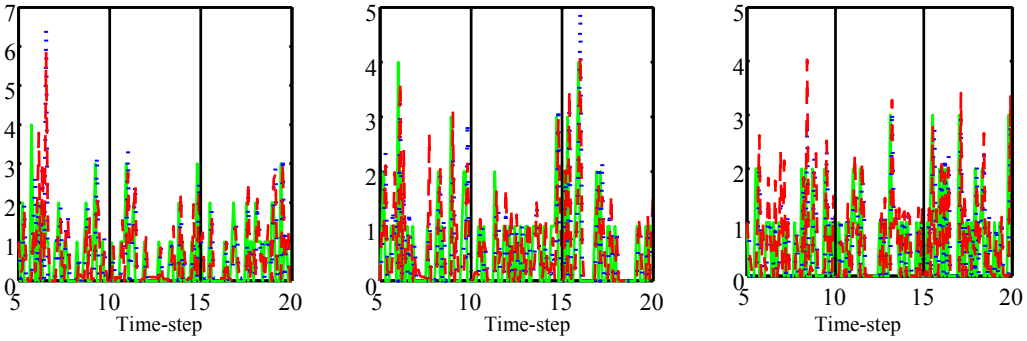
Training Window	Early		Mid		Late	
	lag 1	lag 2	lag 1	lag 2	lag 1	lag 2
50 time-steps	1.02	0.91	0.89	0.74	0.79	0.72
100 time-steps	0.80	0.78	0.75	0.62	0.62	0.60
150 time-steps	0.71	0.67	0.63	0.58	0.59	0.53
Average	0.84	0.79	0.76	0.65	0.67	0.62

Table 11. Mean Absolute Error for next time-step's truthful rating prediction using VAR model

Training Window	Early		Mid		Late	
	lag 1	lag 2	lag 1	lag 2	lag 1	lag 2
50 time-steps	0.161	0.149	0.143	0.140	0.108	0.102
100 time-steps	0.144	0.138	0.136	0.130	0.092	0.089
150 time-steps	0.130	0.124	0.122	0.120	0.088	0.082
Average	0.145	0.137	0.134	0.130	0.096	0.091

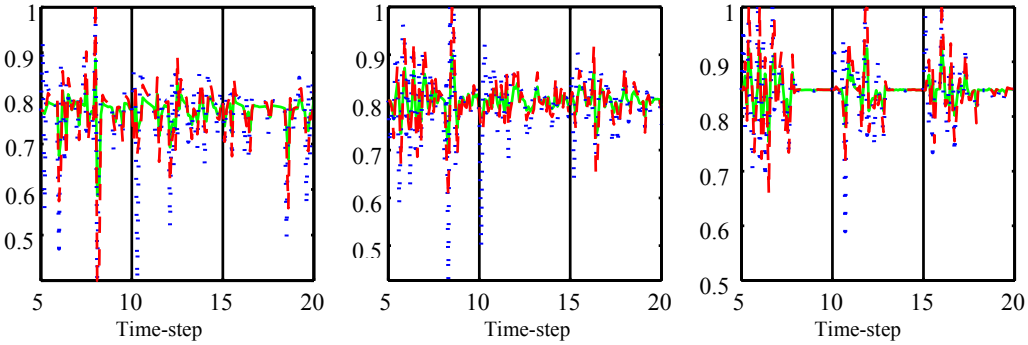
8. Predicting Imminent Future

The previous section provided us insights on long term (≥ 6 month) future prediction of a restaurant's popularity and rating. However, in restaurant business, imminent prediction (e.g., next week's popularity/rating) based on the review data till current time is more useful as it can help assess the recent impact of fake reviews or even help restaurants devise their plans easily. This section employs VARs on popularity and cumulative rating time-series to predict the imminent future performance of a restaurant. We found that the restaurant businesses have different dynamics for different days of the week. So, instead of using week or month as a time unit, we devised a novel time unit based on pooling multiple days of a week. For each week, Mon/Tue jointly formed the first time-step, Wed/Thu the second, and Fri/Sat/Sun the third followed by next week's Mon/Tue as the fourth time-step.



a) Early Spamming b) Mid Spamming c) Late Spamming

Figure 16. Imminent Truthful Popularity (# of reviews) Prediction using features on truthful reviews. Forecasting was done using two p-lag VARs: dotted blue refers to p=1 time-step lag model; red dashed refers to p=2 time-step lag predictor. Solid green line represents the actual popularity.



a) Early Spamming b) Mid Spamming c) Late Spamming

Figure 17. Imminent Truthful Average Rating Prediction using features on truthful reviews. Forecasting was done using two p-lag VARs: dotted blue refers to p=1 time-step lag model; red dashed refers to p=2 time-step lag predictor. Solid green line represents the actual rating.

8.1 Popularity/Rating Time Series VARs

To build good VAR predictors of time-series of popularity and cumulative average rating, we investigated various time-series features as covariates and generated their respective CCF plots with the time-series of truthful popularity and truthful average rating as response variables. Then for time-series feature selection, we sampled covariates that had significant CCF values, at positive lags. The time-series features (covariates, $X(t)$) for each restaurant are tabulated in Table 9.

We show the CCF plots for selected covariates in Fig. 14, 15. We see that the features: (i) total number of friends of reviewers, (ii) # of reviewers have significant positive correlation at positive lags, indicating that they are good predictors of restaurant popularity in next time step (Fig 14.a, b). Similarly, for the case of next time step's average rating, the time-series of dislike count of reviews (Fig. 15.a) shows a significant negative correlation which is arguable as having more dislike reviews in previous time-steps can impact the cumulative average rating in future time-steps. Other significantly correlated features include, number of reviewers with 5 or more reviews (Fig 15.b) which is quite intuitive. In fact all the 12 features listed above had a significant CCF value at lags 1 and 2 periods. So, all those have been used as time series in the VAR model.

We trained VARs with the time-series of the 12 covariates (Table 9) and 2 response variables (popularity/average rating) for each restaurant at lags 1 and 2. We predicted the next time-step truthful popularity and truthful average and experimented with 3 moving training window widths, $W = 50, 100, 150$ time-steps. Similar to §6, we use moving window based forecasting. Fig. 16, 17 show the time-series forecasting performance (degree of fit to the actual response) on representative restaurants in each policy for popularity and rating predictions respectively. Table 10, 11 show the corresponding MAEs averaged over all restaurants in each policy.

We note that across all policies, longer training windows improve imminent popularity and rating predictions as seen by the tightness of fit in Fig 16, 17 and respective MAEs in 10, 11. VAR models with lag 2 are better as they yield one more time step to regress on. Comparing the MAE across the three policies, we see that for early spamming, it is harder to predict the next time-steps popularity and rating than mid and late spamming. This is because early spamming having higher change rate of deceptive review injection (see §6) makes the prediction of imminent popularity and rating more difficult. It is also important to note that the trained models are good predictors of next time-step popularity and average rating as MAEs are quite low. The MAE of popularity is in the range of [0.53, 1.02] reviews (given median popularity per time-step is 19) and the MAE of rating lies in the range [0.082, 0.161] on a normalized scale [0, 1].

Table 12. Mean Absolute Error for next time-step's truthful popularity (# of reviews) prediction using VAR model with the deceptive review series as exogenous inputs

Training Window	Early		Mid		Late	
	lag 1	lag 2	lag 1	lag 2	lag 1	lag 2
50 time-steps	1.44	1.28	1.36	1.32	1.18	1.08
100 time-steps	1.32	1.14	1.28	1.27	1.09	1.07
150 time-steps	1.26	1.12	1.22	1.20	0.98	0.92
Average	1.34	1.18	1.29	1.26	1.08	1.02

Table 13. Mean Absolute Error for next time-step's truthful rating prediction using VAR model with the deceptive review series as exogenous inputs

Training Window	Early		Mid		Late	
	lag 1	lag 2	lag 1	lag 2	lag 1	lag 2
50 time-steps	0.344	0.325	0.312	0.302	0.288	0.253
100 time-steps	0.323	0.298	0.276	0.270	0.245	0.217
150 time-steps	0.283	0.212	0.244	0.200	0.196	0.194
Average	0.317	0.278	0.277	0.257	0.243	0.221

8.2 Modeling Deceptive Noise via Exogenous Variables

How do fake reviews filtered by Yelp affect the imminent future predictions of a restaurant's popularity and rating? Answering this can render insights into the robustness of Yelp's filtering on

Table 15. Time series features for Deception Detection

The standard deviation of the ratings of the truthful reviews in the previous week
The truthful average rating of the previous week reviews only
The truthful like rating of the previous week reviews only
The truthful dislike rating of the previous week reviews only
The truthful review count of the previous week
The truthful like review count of the previous week
The truthful dislike review count of the previous week
The truthful average rating till the review date
The truthful like rating till the review date
The truthful dislike rating till the review date
The standard deviation of the ratings of the previous week deceptive reviews
The deceptive average rating of the previous week reviews only
The deceptive like rating of the previous week reviews only
The deceptive dislike rating of the previous week reviews only
The deceptive review count of the previous week
The deceptive like review count of the previous week
The deceptive dislike review count of the previous week
The deceptive average rating till the review date
The deceptive like rating till the review date
The deceptive dislike rating till the review date

Table 14. SVM 5-fold CV classification results across time series features (TSF), behavioral features (BF), and n-gram features (NG), P: precision, R: recall, F1: F1-Score on fake class, A: Accuracy for classification

	Early Spamming				**Mid Spamming**				**Late Spamming**			
Feature Setting	**P**	**R**	**F1**	**A**	**P**	**R**	**F1**	**A**	**P**	**R**	**F1**	**A**
Ngrams (NG)	63.5	77.1	69.6	65.0	64.2	77.7	70.3	67.5	64.8	78.4	71.0	69.3
Behavior (BF)	82.1	85.3	83.7	84.4	83.3	86.5	84.9	84.7	83.9	87.2	85.5	86.2
TSF	65.2	92.7	76.5	73.1	67.6	93.1	78.3	75.1	68.5	93.9	79.2	76.4
NG+BF+TSF	**84.9**	**94.8**	**89.6**	**89.0**	**85.9**	**94.9**	**90.2**	**89.6**	**86.3**	**95.3**	**90.6**	**90.1**

affecting the imminent future popularity/rating of restaurant. It can also improve the confidence on the reliability of Yelp's filtering (strengthening the conclusions §7.1). Similar to the analysis in §7.1, we include the filtered reviews in the VARs trained in §8.1 in predicting the imminent truthful popularity and rating. However, since our response variable are time-series, we cannot directly add the filtered reviews in the training set. Hence, we modeled the filtered reviews as exogenous variables in our VAR models. A VAR with exogenous variables takes the following form:

$$y_t = a + X_t . b + \sum_{i=1}^{p} A_i y_{t-i} + \varepsilon_t \quad (3)$$

where X_t is an $n \times r$ matrix representing the r exogenous values for each of the n elements in y_t. The other terms are similar to the traditional VARs detailed in §6. Exogenous variables can be seen as "additional" signal for each time-series. In our setting, we only used $r = 1$ exogenous value for each time-series. Specifically, the exogenous variables took values of all the 14 features used in the preceding analysis (§8.1) with the exception that those features were calculated only on the filtered reviews.

We repeated the experiments in Table 10 and 11 with filtered reviews included in the training of VAR models as exogenous variables. The MAEs have been reported in Table 12 and 13. We note that across all policies and both rating and popularity predictors the MAE of VARs with filtered reviews as exogenous variables is higher. The increase in MAEs for all policies and across both popularity and rating predictors were statistically significant at 98% confidence levels (using a paired t-test). All other trends between the relative errors across policies and prediction performance based on lags remain the same as in Table 10, 11.

Thus, additional knowledge gained upon using the filtered reviews in VARs for popularity and rating predictions is actually harmful indicating the filtered reviews as being noisy and non-informative in predicting the imminent truthful popularity and ratings for a restaurant. In other words, those filtered reviews were likely to be untrue experiences as using them in the exogenous variables of 12 modalities in Table 9 increased the error against using the same 12 modalities on truthful reviews which had significant cross correlations with the target responses (see Fig. 14, 15). Yelp was able to detect those reviews as fake and filter them which not only shows that Yelp's filter is decent but also indicates its resilience that allowed it to pass our tests on imminent future predictions of rating and popularity.

9. Leveraging Temporal Dynamics

Having characterized the temporal dynamics of opinion spamming, can we improve deception prediction beyond the existing state-of-the-art approaches leveraging the knowledge of temporal dynamics? To answer this we used our Yelp data and its filtering labels (filtered: fake; non-filtered-truthful) to set up the fake review detection as a classification problem. We use two state-of-the-art approaches as our baselines: Ott et al., (2011) which employed linguistic n-grams (NG) and Mukherjee et al. (2013) which uses a set of 8 anomalous behavioral features (BF) (e.g., reviewer deviation, percentage of positive reviews, etc). Classification settings were same as in [26], except that we partitioned the Yelp data by the spamming policies. We trained linear kernel SVMs with 5-fold cross validation on balanced data (using under-sampling). The soft margin parameter was tuned using cross validation and set to $C = 1.5$. We derived a set of Time-Series Features (TSF) (Table 14) from various analyses in this work. We compare TSF and TSF+NG+BF against the baselines in Table 15. We note that TSF alone does significantly better than linguistic n-grams, but is weaker than the 8 anomalous behaviors proposed in [26]. However, upon combining TSF with NG and BF feature sets, we obtain the highest F-scores which are significantly better than both linguistic and behavioral features demonstrating that the discovered temporal dynamics have a value in improving deception detection beyond just characterization. We also performed feature ablation (not shown due to space constraints) and found deceptive review count, like and dislike ratings (see Table 14) to be among the most discriminative features.

10. Conclusion

This paper performed an in-depth analyses on the temporal dynamics of opinion spamming. It used a large-set of reviews from Yelp restaurants and its filtered reviews to characterize the way opinion spamming operates in a commercial setting. Experiments using time-series analyses showed that there exist three dominant spamming policies: early, mid, and late across various restaurant. Our analyses showed that the deception rating time-series for each restaurant had statistically significant correlations with the dynamics of truthful ratings time-series indicating that spam injection may potentially be coordinated by the restaurants/spammers to counter the effect of unfavorable ratings over time. Causal time-series analysis of deceptive like rating time-series as response with different covariates time-series (e.g., average truthful ratings, truthful like and truthful dislike ratings) established the presence to two additional trends of spam injection: *buffered* and *reduced* spamming. The covariate time-series along with various other features were then used to predict future deceptive ratings, long term/imminent future popularity and rating of a restaurant in the presence of deception using vector auto regression. The framework further allowed us to indirectly validate Yelp's filter which was shown to be reasonable. We also derived a novel suite of time-series features from our discovered temporal dynamics. Experiments on fake review detection showed the effectiveness of our features that significantly outperformed relevant baselines for the task of opinion spam detection establishing a value of the temporal dynamics in spam detection beyond characterization.

11. Acknowledgement

This work is supported in part by NSF 1527364. We also thank anonymous reviewers for their helpful feedbacks.

12. REFERENCES

[1] Chen, Y.-R. and Chen, H.-H. 2015. Opinion Spam Detection in Web Forum: A Real Case Study. *Proceedings of the 24th International Conference on World Wide Web* (2015), 173–183.

[2] Chirita, P.A., Diederich, J., and Nejdl, W. 2005. MailRank : Using Ranking for Spam Detection. *Conference on Information and Knowledge Management* (2005).

[3] Fei, G., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M. and Ghosh, R. 2013. Exploiting Burstiness in Reviews for Review Spammer Detection. *AAAI International Conference on Weblogs and Social Media*. (2013).

[4] Feng, S., Xing, L., Gogar, A. and Choi, Y. 2012. Distributional Footprints of Deceptive Product Reviews. *The International AAAI Conference on Weblogs and Social Media* (2012).

[5] Feng, S., Banerjee R., Choi, Y. 2012. Syntactic Stylometry for Deception Detection. *Association for Computational Linguistics* (2012).

[6] Jindal, N. and Liu, B. 2008. Opinion Spam and Analysis. *ACM International Conference on Web Search and Data Mining* (2008).

[7] Jindal, N., Liu, B. and Lim, E.-P. 2010. Finding Unusual Review Patterns Using Unexpected Rules. *Conference on Information and Knowledge Management* (2010).

[8] Kolari, P., Java, A., Finin, T., Oates, T. and Joshi, A. 2006. Detecting Spam Blogs : A Machine Learning Approach ∗. *American Association for Advancement of Artificial Intelligence* (2006).

[9] Li, F., Gao, Y., Zhou, S., Si, X. and Dai, D. 2013. Deceptive Answer Prediction with User Preference Graph. *ACL (1)* (2013), 1723–1732.

[10] Li, F., Huang, M., Yang, Y. and Zhu, X. 2011. Learning to Identify Review Spam. *International Joint Conference on AI* (2011), 2488–2493.

[11] Li, H., Chen, Z., Liu, B., Wei, X. and Shao, J. 2014. Spotting Fake Reviews via Collective Positive-Unlabeled Learning. *IEEE International Conference on Data Mining*. (2014).

[12] Li, H., Chen, Z., Mukherjee, A., Liu, B. and Shao, J. 2015. Analyzing and Detecting Opinion Spam on a Large-scale Dataset via Temporal and Spatial Patterns. *Proceedings of Ninth International AAAI Conference on Web and Social Media*. (2015).

[13] Li, H., Liu, B., Mukherjee, A. and Shao, J. 2014. Spotting Fake Reviews using Positive-Unlabeled Learning. *Computaci{ó}n y Sistemas*. 18, 3 (2014).

[14] Li, J., Cardie, C. and Li, S. 2013. TopicSpam: a Topic-Model-Based Approach for Spam Detection. *Annual Meeting of the Association for Computational Linguistics*. (2013).

[15] Li, J., Ott, M. and Cardie, C. 2013. Identifying Manipulated Offerings on Review Portals. *Empirical Methods in Natural Language Processing*. (2013).

[16] Li, J., Ott, M., Cardie, C. and Hovy, E. 2014. Towards a General Rule for Identifying Deceptive Opinion Spam. *Association for Computational Linguistics*. (2014).

[17] Lim, E.-P., Nguyen, V.-A., Jindal, N., Liu, B. and Lauw, H.W. 2010. Detecting product review spammers using rating behaviors. *Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10* (New York, New York, USA, 2010), 939.

[18] Luca, M. and Zervas, G. 2013. Fake it till you make it: Reputation, competition, and Yelp review fraud. *Harvard Business School NOM Unit Working Paper*. 14-006 (2013).

[19] Mukherjee, A. 2015. Detecting Deceptive Opinion Spam using Linguistics, Behavioral and Statistical Modeling. *Association for Computational Linguistics Tutorials* (Beijing, China, Jul. 2015), 21–22.

[20] Mukherjee, A., Kumar, A., Liu, B., Wang, J., Hsu, M., Castellanos, M. and Ghosh, R. 2013. Spotting Opinion Spammers using Behavioral Footprints. *Proceedings of the 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. (2013).

[21] Mukherjee, A. and Liu, B. 2012. Aspect Extraction through Semi-Supervised Modeling. *Association for Computational Linguistics*. (2012).

[22] Mukherjee, A., Liu, B. and Glance, N. 2012. Spotting Fake Reviewer Groups in Consumer Reviews. *Proceedings of the 21st international conference on World Wide Web - WWW '12* (2012).

[23] Mukherjee, A., Liu, B., Wang, J., Glance, N. and Jindal, N. 2011. Detecting Group Review Spam. *WWW* (2011).

[24] Mukherjee, A. and Venkataraman, V. 2014. Opinion Spam Detection: An Unsupervised Approach using Generative Models. *Techincal Report, UH*. (2014).

[25] Mukherjee, A., Venkataraman, V., Liu, B. and Glance, N. 2013. Fake Review Detection: Classification and Analysis of Real and Pseudo Reviews. *Techical Report UIC-CS-2013-03*. (2013).

[26] Mukherjee, A., Venkataraman, V., Liu, B. and Glance, N. 2013. What Yelp Fake Review Filter might be Doing? *AAAI International Conference on Weblogs and Social Media*. (2013).

[27] Newman, M.L., Pennebaker, J.W., Berry, D.S., Richards, J.M. 2003. Lying words: predicting deception from linguistic styles. *Personality and Social Psychology Bulletin*. (2003), 665–675.

[28] Ott, M., Cardie, C. and Hancock, J. 2012. Estimating the prevalence of deception in online review communities. *Proceedings of the 21st international conference on World Wide Web - WWW '12* (New York, New York, USA, 2012), 201.

[29] Ott, M., Choi, Y., Cardie, C. and Hancock, J.T. 2011. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. *Association for Computational Linguistics* (2011), 309–319.

[30] Qian, T. and Liu, B. 2013. Identifying Multiple Userids of the Same Author. *Empirical Methods in Natural Language Processing*. (2013).

[31] Qian, T., Liu, B., Chen, L. and Peng, Z. 2014. Tri-Training for Authorship Attribution with Limited Training Data. *Association for Computational Linguistics*. (2014).

[32] Shen, G., Gao, B., Liu, T.-Y., Feng, G., Song, S. and Li, H. 2006. Detecting link spam using temporal information. *Data Mining, 2006. ICDM'06. Sixth International Conference on* (2006), 1049–1053.

[33] Solorio, T., Hasan, R. and Mizan, M. 2013. A case study of sockpuppet detection in Wikipedia. *Workshop on Language Analysis in Social Media (LASM) at NAACL HLT 2013, ed Atlanta, GA: Association for Computational Linguistics*. (2013).

[34] Wang, Z. 2010. Anonymity, Social Image, and the Competition for Volunteers: A Case Study of the Online Market for Reviews. *The B.E. Journal of Economic Analysis & Policy*. 10, 1 (Jan. 2010), 1–34.

[35] Xie, S., Wang, G., Lin, S. and Yu, P.S. 2012. Review spam detection via temporal pattern discovery. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12*. (2012), 823.

[36] Yang, J. and Leskovec, J. 2011. Patterns of temporal variation in online media. *Proceedings of the fourth ACM international conference on Web search and data mining* (2011), 177–186.