TREND DETECTION IN SOCIAL DATA

SCOTT HENDRICKSON   JEFF KOLB   BRIAN LEHMAN   JOSHUA MONTAGUE

1   Introduction

User interactions in social data tell us a great deal about the real world, and these insights are not limited to any particular segments of time. By understanding the time-dependent behavior of groups of social media users, we can identify and even predict important real-world trends.

What sort of real-world activities, events, and trends might be reflected in social data? And what properties of those events might we want to know?

Suppose that an influential financial analyst Tweets a strong opinion about a particular stock and that Tweet goes viral. Or suppose a large number of customers use Twitter to complain about a brands new product. In both cases, good first questions are: when did the event happen, or when did the trend start? As a follow-up, we can also ask: how significant is the change? How large is the increase or decrease? More importantly, how large is this change relative to typical changes on Twitter? The quantification not only allows an analyst to distinguish the atypical from the typical, but it also allows them to compare one atypical event to another. Are there characteristics of atypical events that allow them to be separated into groups that can then be assigned real-world meaning (e.g. seasonal trends, holiday events)? If so, do these assignments point toward a particular choice of quantitative model for the event? If the identification of an atypical period of time can be quantified, can it be automated? And can it be used to predict future behavior?

After reading this paper, you will be well on your way to building a system for discovering, measuring, comparing and discussing changes in time series data that arise from online social interactions.

2   Trends and Events

We measure time-dependent user behavior with bucketed counts of mentions, hashtags, followers, friends, links, or any quantity that can be counted over time. If that quantity is defined by the presence of a word or phrase, that word or phrase is called the *topic*. When discussing changes in a social data time series, we must speak concretely about what kinds of change are interesting. For example, growth over time of an audience is a simple but important measure of change. Similarly, we may want to know about seasonal cycles of change. How does June compare to December? Against the backdrop of steady growth and cyclic variations, we can ask about emerging changes, in which a count rises from something negligible or unimportant to something significant. We are also interesting in structural changes, where a time series abruptly shifts from one state to another.

2.1   Difficulties

Identifying growth, cycles, and – especially – emerging and structural changes, is hard. Why? A primary difficulty is the fact that we often don't know in advance

the scale or size of the change. The time interval over which a change occurs can range from fractions of seconds to years, a difference of 10 orders of magnitude! Additionally, the size of the change can range from counts of 10s through counts of billions. The community of users generating or associated with the change can range from a single person through a group of 100 million people. It is difficult to construct algorithms that function evenly over such broad ranges of data. Finally, we know that changes of many sorts is happening all the time, and we want to take care to identify single, rather than composite, changes.

The corpus of social data is enormous, and this size brings about other difficulties. Most signals of interest are relatively small. Data that match a particular topical filter are usually contaiminated by other signals, and changes in the data reflect the cumulative result of all underlying effects. The size of the data also implies the existence of many atypical patterns that are entirely due to statistical variation, rather than reflecting real-world events. Despite this knowledge, humans prefer to associate any changes with meaningful and nameable events. Even the distinction between the real world and the online social interactions is complicated, and it can be difficult to establish causality. Do the social data simply reflect the offline world? How do online social interactions affect the rest of the world? The first step in unravelling this tangled feedback network is to quantify the social data trends.

2.2   Analysis Trade-offs

Attempts to quantify changes in social data are subject to trade-offs. At times, random fluctuations in the data will be identified as a trend. At other times, real trends will not be identified. We identify three particular measures of performance that account for these types of mistakes. First is the *time-to-detection*, or the time between the real-world event and the detection in the social data. Second is the *precision*, or the fraction of identified trends that are not statistical flukes. Last is the *recall*, or the fraction of real trends that are identified by the trend detection scheme. These performance metrics can not be simultaneously optimized. For example, if we wish to quickly identify an emerging change, and we wish to do so with high confidence that were not detecting random fluctuations, we will necessarily have low recall for real trends, and be able to only identify very statistically significant patterns.

2.3   Classification

Once the detection scheme is defined, humans have to interpret and act on anomalous events as they are observed. These actions can take the following forms:

- alerting - start paying attention to something new and urgent
- informing - note the relative state of things available when someone checks

- discovery - do iterative refinement for novelty detections or root cause analysis

- model building - enable downstream consumption of the signal for other modeling purposes

Given these challenges and considerations, lets organize the analysis around three classes of anomalies, as seen in Figure 1. While anomolous decreases in time series can be interesting, we will limit ourselves for the duration of the paper to the specific case of atypical increases.

Ramp-up: from a well-understood steady state (negligible, constant, or periodic), the time series exhibits a continuing increase that is sustained over many instances of the time resolution.

Mean shift: from a well-understood steady state, the mean of the time series shifts abruptly to a significantly different value and maintains that value over a time span much longer than the time resolution.

Pulse: from a well-understood steady state, the value of a time series increases significantly, then returns to previously-typical values. Pulses with widths similar to the time resolution capture the briefest events that can be observed. Those with widths much larger than the time resolution represent extended events that can be further characterized by the area under the pulse.
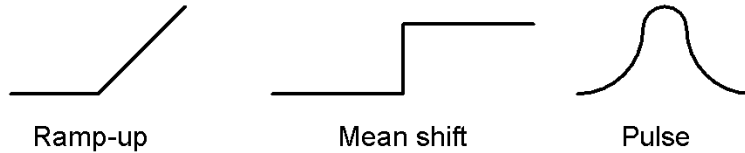

Ramp-up   Mean shift   Pulse

Figure 1: Three basic types of anomalies.

There is some interrelation between these basic anomaly types. For example, a pulse can be thought of as a pair of mean shift or ramp-up/ramp-down anomalies. A higher-level feature like a cycle can also be thought of as a sequence of these anomalies.

A final challenge is the mapping between anomalies and real-world events. The word "event" can refer to a nameable change (e.g. Superbowl mentions), but it can also refer to any interval in a time series that is sufficiently atypical, with no meaning attached. In the remainder of this paper, we use the word event to refer to specific, nameable happenings in the either the online or the offline world.

We encourage the reader to systematically think about what sorts of change are important to the problem they're trying to solve, and what sorts of action are to be taken upon detection. Identifying and characterizing atypical behavior in

social data time series can be difficult, but it provides us with brand-new insights into group behavior and the interplay between the online and the offline world.

3   Brief Survey of Analytical Approaches

In the section, we will work through the details of a simple technique for trend detection in a time series defined by a topic on Twitter. We will then briefly review a set of other techniques that extend, generalize, or improve on the simple case. See the Appendix for a link to implementations of these techniques.

Many techniques for identifying trending behavior define a background model, which can be thought to represent the null hypothesis, or the case of no trend. Deviations from the background model are described by a figure-of-merit called η, and large values of η can be said to disprove the null hypothesis. In other techniques, the model includes both a background component and and trend-like component. In these cases, the η value quantifies the extent to which the data look more like a trend than a non-trend. Whatever the model, we say that the topic is trending at the time the η exceeds a pre-determined value.

To calculate η, we typically must choose some model parameter values. If we have access to historical data that is labeled with truth (trend or no-trend) and the true trend start time, we can measure the performance of a choice of model and parameter values, in terms of the precision, the recall, and time-to-detection.

3.1   Point-by-point Poisson Model

The Poisson distribution describes the probability of observing a particular count of some quantity, when many sources have individually low probabilities of contributing to the count. This sounds applicable to the case of counting in social data, because each individual has a small chance of Tweeting about a given topic, but the large Twitter user base leads to significant counts. We can do a rather simple form of trend detection by assuming that the counts in a social data time series are Poisson-distributed around some average value, and then looking for unlikely counts according to the Poisson model. Consider, for example, the number of Tweets in some time interval that contain the hash-tagged phrase "#scotus" (referencing the Supreme Court of the United States). If we ignore variations in the overall rate of Tweeting, we might expect the counts of "#scotus"-containing Tweets to vary, but the distributions of counts will generally follow the Poisson distribution,

$$P(c_i; \nu) = \nu^{c_i} \cdot e^{-\nu}/c_i! \tag{1}$$

where P is the probability of observing c_i "#scotus" Tweets in the given time window, when the expected number of such Tweets is ν. While we have no way of knowing the true value of ν, a good source is the time interval *previous* to the one being tested, c_{i-1}. We identify trends by counts c_i that are particularly unlikely, given the previous count, c_{i-1}, and the assumption of Poisson distributed data.
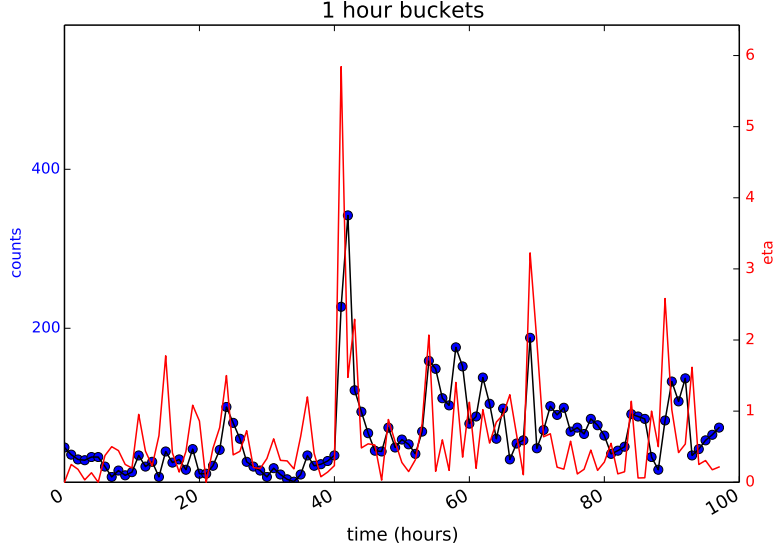
Figure 2: Mentions of "#scotus" per hour. The time series data are shown in the blue dots with black lines. For each point, η is calculated based on the previous point, and plotted in red. In this case, $\alpha = 0.99$.

In this Poisson model, the unlikeliness of a particular count c_i can be quantified by the distance from the mean (ν) in multiples of the confidence interval (CI) with confidence level α. Confidence intervals for a Poisson mean ν and confidence level α can be found in [1]. The parameter η described the unlikeliness of a particular point:

$$c_i = \eta \cdot \text{CI}(\alpha, \nu) + \nu, \text{where } \nu = c_{i-1}. \tag{2}$$

In other words, a count c_i is defined to reject the null hypothesis when

$$c_i >= \eta_c \cdot \text{CI}(\alpha, c_{i-1}) + c_{i-1}, \tag{3}$$

for predetermined values of η_c and α. Together, these two parameters control the performance of the algorithm. Poisson confidence intervals.

In Figure 2, we plot the hourly counts for a data set defined by the "#scotus" hashtag. While there may be minor events driving mentions of "#scotus", this time series does not contain any major real-world events, and the values of η are relatively low.

The point-by-point Poisson model is an attempt to simplify the problem of background description by assuming a very simple model. Yet this simplicity can
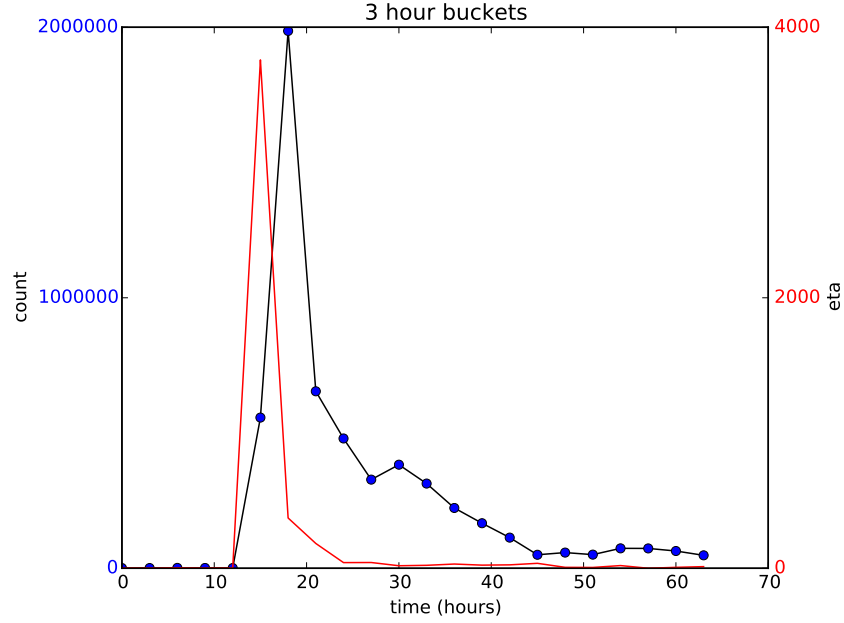
6

Figure 3: Mentions per 3-hour intervals of Steve Jobs, around the time of his death. The time series data are shown in the blue dots with black lines. For each point, η is calculated based on the previous point, and plotted in red. In this case, $\alpha = 0.99$.

be a source of challenges. First, we see that the data are generally not Poisson-distributed around the previous data point. For example, given a choice of $\alpha = 0.99$, we should see values of $\eta > 1$ only about 1% of the time. Nevertheless, the parameter η is indicative of atypical counts, just not with the usual probability interpretation. For example, Figure 3 shows a time series with a very distinctive, large spike, and the corresponding values of η are very large.

Once a value of α is chosen, the definition and identification of a trend is still dependent on the choice of two parameters values: η_c and the time interval for a single data point. As η_c is increased, the precision is increased, but more real trends are missed (decreased recall). A similar trade-off exists for the bin width, as shown in Figure 4: small bins provide faster identification of trends, but lead to worse precision.

Despite the challenge of choosing appropriate parameter values, the point-by-point Poisson model can be very appealing. It's fast, in part because it requires a single data point for the background model. It's also easy to implement, and its single measure of atypicality, η, is fairly easy to interpret.
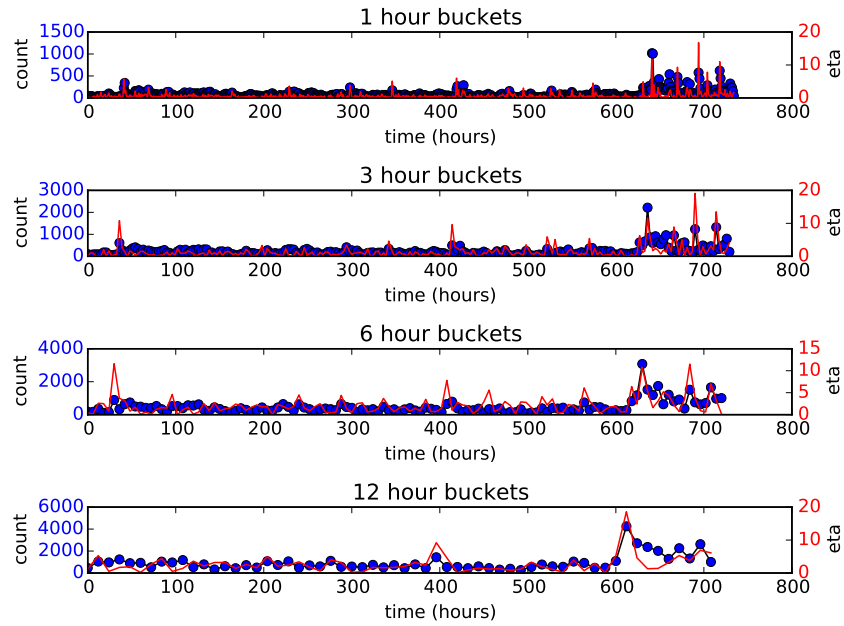
Figure 4: Counts of "#scotus" mentions, in variously-sized time buckets. The time series data are shown in the blue dots with black lines. For each point, η is calculated based on the previous point, and plotted in red. In this case, $\alpha = 0.99$.

3.2 Cycle-corrected Poisson Model

Most social data time series exhibit cyclic patterns that reflect genuine human cycles of activity. For example, if the majority of users that generate a particular body of Tweets live in a narrow band of time zones, we would naturally expect to see fewer Tweets during night hours for those time zones. Thus, the patterns of hours, days, weeks, and even months can be reflected in changes in rates of social media use.

To reduce the rate of false trend identification due to expected, cyclic human activity, the cycle-corrected Poisson model builds on the foundation of the point-by-point Poisson model, and uses a background model derived from data similar to the point being tested. For example, if a data point represents 3 hours of data from a Friday night in the Eastern US, it would not make a good model to use the previous three hours as the Poisson mean. People Tweet about different topics at 2-5PM than they do at 5-8PM, leading to topical time series with large variations simply due to the progression of the day. A better background model for the data from 5-8PM on a particular Friday is an average over the data from the 5-8PM interval on other days of the week. We can build an even better model by taking the average over the same time interval, but only from previous Fridays. If monthly cycles of activity are important, we might even build our background estimate from only Fridays around the same time of the month.

The primary drawback of this technique, relative to the point-by-point model, is the need to sample and retain enough data to calculate the background estimates. If anomalous events have previously occurred in the time series, this will contribute to the rolling averages and artificially increase the rate of false positives. Figures 5 and 6 shows a comparison of the two Poisson-based background model discussed in this and in the previous section. The cycle-corrected model shows generally reduced η values (fewer false positives), but actually produces a greater η value at the initial spike around hour 650.

Continuing to expand on the basic Poisson model, there is a variety of further improvements that can be made. Any value chosen for the Poisson mean can be further stabilized by calculating an average over a rolling window of adjacent data points. If the long-term overall growth rate for the data is known, this baseline can be subtracted from the data. Ihler, Hutchins, Smyth [2] present a framework for removing the effects of previously-occurring anomalies from the Poisson background model.

3.3 A Data-driven Method

There is a two-fold drawback to the Poisson models: first, it is impossible to choose values for α and η_c that are a good choice for trends of all shapes and sizes. Moreover, our decision to use the Poisson distribution as a model for the variations in the data is not necessarily a good choice. In fact, we know that many social data time series are not Poisson-distributed. What if we were to avoid these problems by simply comparing the data to real examples of trending
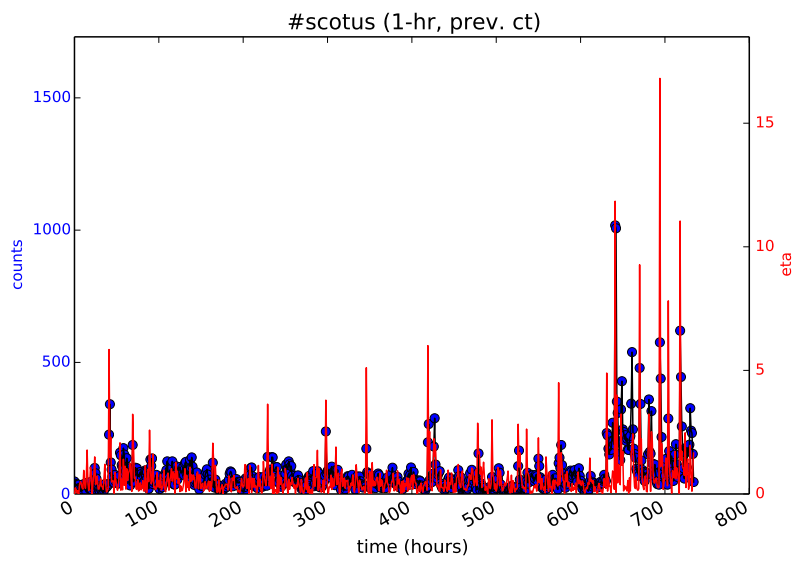
Figure 5: Counts of "#scotus" mentions, in 1-hour time buckets. The time series data are shown in the blue dots with black lines. For each point, η is calculated based on the previous point, and plotted in red. In this case, $\alpha = 0.99$.
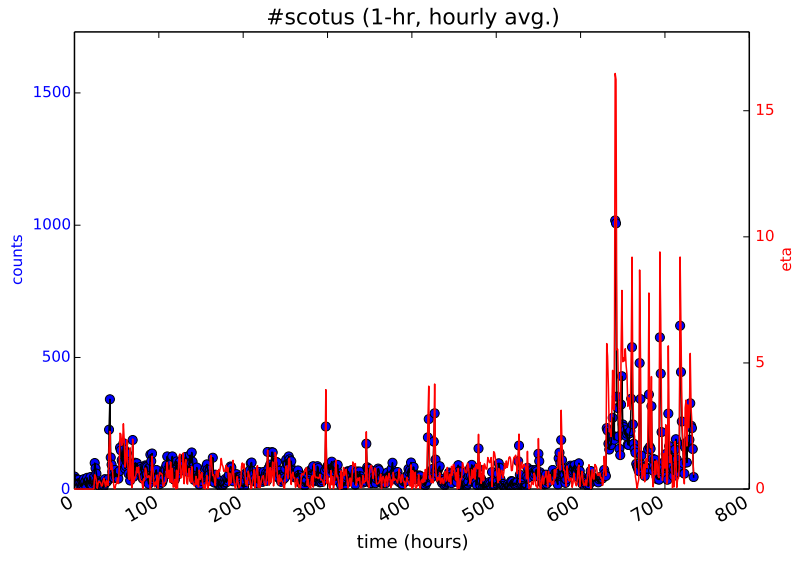
Figure 6: Counts of "#scotus" mentions, in 1-hour time buckets. The time series data are shown in the blue dots with black lines. For each point, η is calculated based on the average value from the same hour on previous days in the time series, and plotted in red. In this case, $\alpha = 0.99$.

and non-trending data?

Nikolov suggests we do just this in a non-parametric method [3]. We begin by compiling a library of labeled time series, identifying each as trending or non-trending. We then define a weight that is a function of the distance between a labeled time series and the data in question. The final result is given by the ratio of the total weight for the trending time series divided by the total weight for the non-trending time series.

We start by collecting *reference* time series from historical data. Based on their shape and the details of real-life events associated with them, we label them $+$ (trending) or $-$ (non-trending). The sets of references time series are named $R+$ and $R-$ (and together comprise R), and typically haves sizes in the 100s. In general, the elements of $R+$ and $R-$ are much longer than the time series that they are compared to.

We next define a distance between two same-length time series: $d(r, s)$, where r is in $R+$ or $R-$ and s is the time series that were evaluating for trending behavior. To facilitate comparison, both time series are unit-normalized. We use the Euclidean distance:

$$d(r, s) = \Sigma_i^N (r_i - s_i)^2, \tag{4}$$

where r_i and s_i are the i-th points in the N-length time series r and s. Other choices of distance functions emphasize different properties of the time series, and lead to different value of the trend detection metrics discussed below. If r is longer than s, we define the distance to be the smallest of all distances $d(r_s, s)$, where r_s is any s-length sub-series of r. Given a distance function, we then define a weight in terms of a scaling parameter λ.

$$W(r, s) = e^{-\lambda \cdot d(r,s)} \tag{5}$$

The parameter λ controls the relative importance of very similar vs. very different reference series. For example, a large value of lambda generates very small weights for elements of R very different from s.

We then sum up the weights from the trending and non-trending comparisons and produce a final metric from their ratio:

$$\eta(s) = \frac{\Sigma_{r \in R+} W(r, s)}{\Sigma_{r \in R-} W(r, s)} \tag{6}$$

To demonstrate the performance of this technique on a known trend, Figure 7 shows a plot of a single element of R+, along with η as calculated for this time series. The η curve rises dramatically soon after the real spike in the data, with the lagtime demonstrating the effect of the data-smoothing.

The primary difficulty with this method is the need for a labeled set of reference time series. To obtain similar detection performance over a broad range of trend shapes and sizes, it is also important to apply a series of transformations to all r and s. In our implementation, these transformation include the previously-mentioned unit normalization, a smoothing with an average taken
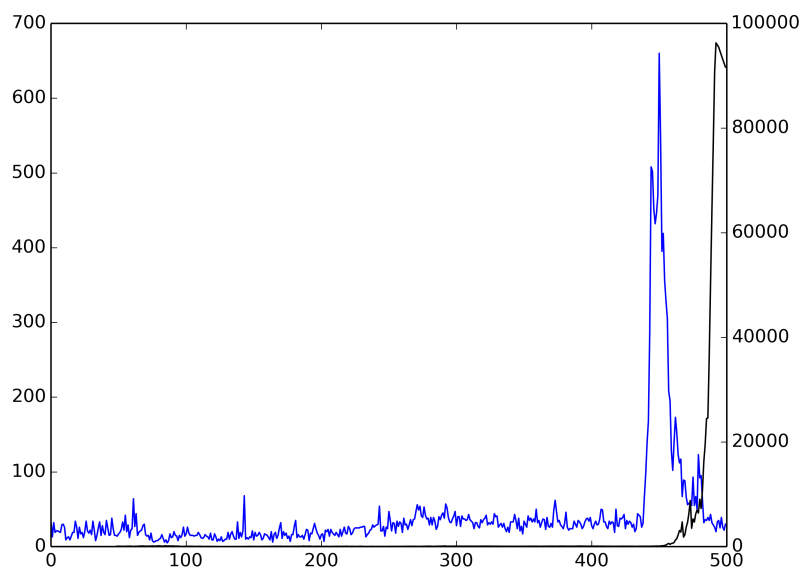
Figure 7: Data from a single reference time series in $R+$ are plotted in blue, for 2-minute time intervals. Based on a library of 500 reference trends in $R+$ (minus the one being tested) and 500 reference non-trends in $R-$, the figure of merit η is calculated for each point and plotted in black. The length of elements of R is 200 minutes, while the length of the tested series (s) is 100 minutes.
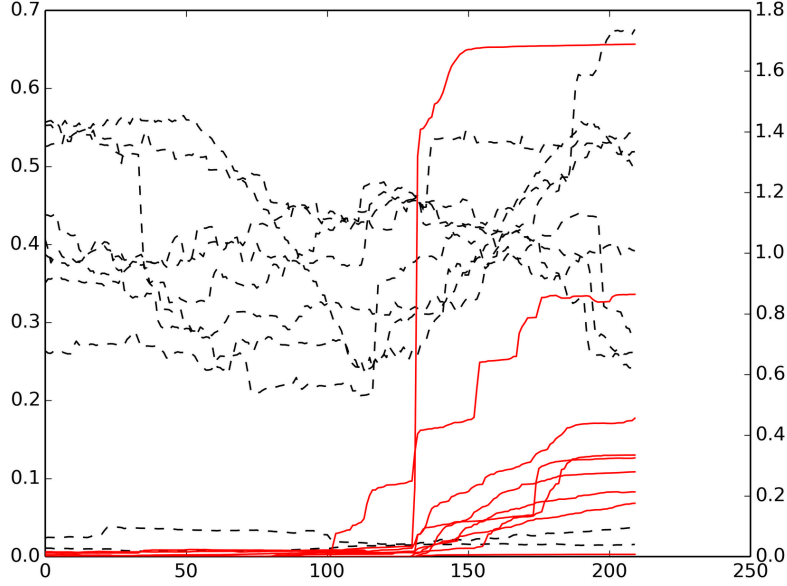
13

Figure 8: A plot of elements of $R+$ (red lines) and $R-$ (black dashed lines), after the smoothing and scaling described in [3]. The trending series in $R+$ rise sharply at the right side of the plot, while changes in series in $R-$ are more evenly distributed.

over a sliding window, and a logarithmic scaling (see [3] or the code referenced in the Appendix for details of the transformations). Examples of the transformed reference time series are shown in Figure 8.

Even though the shapes of the labeled time series provide the model for trending and non-trending time series, the analyst still controls the performance of the algorithm by setting parameter values. The values chosen for the scaling parameter, the lengths of s and r, the time series precision, and any other transformation parameters lead directly to the precision and recall metrics. With the labeled reference series in hand, we can easily calculate these metrics by removing a single element from $R+$ or $R-$ and testing it against the ensemble.

4 Conclusion

Trends in social data tell us about what is important to users of social media. Trends not only reflect real-world events, but also drive offline behavior. By identifying trending behavior, we can be informed of current events, we can discover emerging events, and we can model future events. But reliable, precise,

and fast trend detection is made difficult by the size and diversity of the social data corpus, along with the large variations in the time and volume scales of social data sets.

We have overviewed three techniques of trend detection that strike various balances between simplicity, speed, accuracy, and precision. If simplicity is extremely important, or for a pilot model, we recommend the point-by-point Poisson technique. This technique is most appropriate to small sets of time series, in which typical behavior can be manually observed and correlated with the atypicality parameter (η). If a sufficient history of data is available, we recommend enhancing the technique to account for cyclic behavior, as in the cycle-corrected Poisson technique. This is a relatively small step up in complexity, and provides a significantly decreased rate of false positive signals.

When optimal coverage and false positive rates are worth extra model complexity and technical commitment, the data-driven method is worth investigating. While it is potentially difficult to collect and label a sufficient number of comparison time series, the technique provides stable results across a wide variety of trend detection problem.

A   Appendix

The latest version of this document and implementations of the trend detection models can be found at: `https://github.com/jeffakolb/Gnip-Trend-Detection`.

References

[1] F. George B. Golam Kibria, *Confidence Intervals for Signal to Noise Ratio of a Poisson Distribution*, `http://thescipub.com/abstract/10.3844/amjbsp.2011.44.55` 2013.

[2] Ihler, Hutchins, Smyth, *Adaptive Event Detection with TimeVarying Poisson Processes*, `http://www.datalab.uci.edu/papers/event_detection_kdd06.pdf` 2006.

[3] S. Nikolov, *Trend or No Trend: A Novel Nonparametric Method for Classifying Time Series*, `http://dspace.mit.edu/bitstream/handle/1721.1/85399/870304955.pdf` 2011.

ABOUT THE AUTHORS

Scott Hendrickson, PhD
Data Scientist

Scott currently leads the Gnip Data Science team at Twitter. He has a deep interest in discovering patterns, creating useful models and sharing a deeper understanding of how people can leverage social data from publishers like Twitter. Prior to Gnip, Scott worked with startups focused on data analysis, machine learning, data visualization and data-centric strategy projects.

Josh Montague, PhD
Data Scientist

Josh is a Gnip Data Scientist at Twitter where he conducts research in support of enterprise customers. He is passionate about designing, building, and sharing data-driven projects that solve problems. Before starting his career in software, Josh spent many years as an experimental condensed matter physicist. He spends his free time enjoying the outdoors in Colorado.

Jeff Kolb, PhD
Data Scientist

Jeff is a Gnip Data Scientist at Twitter. He guides people through the process of deriving genuinely useful conclusions from large-scale Twitter data analysis. In a previous career, he studied the origins of mass at CERN.

Brian Lehman
Data Scientist

Brian Lehman is a Gnip Data Scientist at Twitter. Counting his quiver of race bikes usually results in a value near the number of frisbees in his dog's arsenal of retrieval toys. He previously taught mathematics at the Colorado School of Mines.

GNIP

303.997.7488 | info@gnip.com | gnip.com | @gnip