A Machine Learning Approach to Discovering Interesting Moments in Social Media Streams

Cody Buntain
Dept. of Computer Science
University of Maryland
College Park, Maryland 20742
cbuntain@cs.umd.edu

Jimmy Lin
College of Information Studies
University of Maryland
College Park, Maryland 20742
jimmylin@cs.umd.edu

Jen Golbeck
College of Information Studies
University of Maryland
College Park, Maryland 20742
golbeck@cs.umd.edu

ABSTRACT

This paper introduces a general technique for identifying interesting and compelling moments in social media streams without the need for any domain-specific information or seed keyword lists. We leverage machine learning to model temporal patterns around bursts in Twitter's public sample stream and build a classifier to identify tokens experiencing these bursts. Since many events are also interesting moments, we are also able to subsume some event detection tasks and achieve a measure of language agnosticism and general flexibility current techniques lack. To demonstrate our approach's potential, we compare a baseline event-detection system with our language-agnostic algorithm in detecting a number of key moments across three major sporting competitions: the 2013 World Series, 2014 Super Bowl, and 2014 World Cup. Results from this comparison show our method demonstrates similar performance in this sports domain without the need for pre-specified tokens. We then go further by applying our sports-based models to the task of identifying earthquakes in Japan and show our method detects large spikes in earthquake-related tokens within two minutes of the actual event.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining;
H.3.3 [**Information Search and Retrieval**]: Information Filtering

Keywords

event detection, twitter, social networks, temporal features

1. INTRODUCTION

With few exceptions, current social media-based event detection systems rely on tracking occurrences of manually defined keywords to detect interesting events. For instance, Cipriani followed frequencies of "goal" on Twitter during the World Cup to detect when players score goals [6]. Though straightforward and capable, these approaches cannot detect events for which they lack defined keywords, so penalties and red/yellow cards would go undetected. One might respond to this weakness by extending the seed keywords to include penalty-related tokens, but this approach would still be unable to identify *unexpected* moments like Luis Suarez's biting Giorgio Chiellini during the Uruguay-Italy World Cup match; who would have thought to include "bite" as a relevant token during that event? Furthermore, relying on predefined keywords constrains these systems only to languages represented in the seed keyword set, which is a significant issue for events of international interest like the World Cup or in areas that do not speak the target languages. Given the shear volume of social media data (hundreds of thousands of comments, statuses, and photos generated per minute on Facebook alone as of 2012 [18]), we think machine learning could be an ideal solution for addressing these shortcomings.

To explore these questions, we propose a general method for learning temporal patterns of increased token usage (called bursts) and tracking the frequency of tokens experiencing these bursts to identify interesting moments in social media streams. In contrast to existing work, our approach operates without prior knowledge of the target events or any domain-dependent stream filtering. We demonstrate this additional flexibility through experiments on Twitter's public sample stream surrounding key moments in large sporting competitions. Sporting events are both highly followed and occur regularly, which make them ideal for collecting data and exercising event detection systems. Such events are also complex in that they include multiple types and unpredictable patterns of events around scores, fouls, and other compelling moments of action. High-impact sports competitions like the World Cup also have extensive international viewership, so one must also account for multiple languages and different vocabularies. Given the overlap in interesting moments and events, we compare our language-agnostic, burst-centric technique to a baseline keyword-centric method for detecting events in several such sporting competitions. Finally, we transfer the models learned in the sporting domain to earthquake detection and show performance commensurate with existing socially enabled earthquake detection research.

More concisely, this work makes the following contributions:

- We propose a feature set and classifier for the language-agnostic discovery of moments of high interest in the Twitter public sample stream. Importantly, our approach does *not* require a list of manually-defined keywords as input.

- We demonstrate our approach is competitive with respect to a baseline the relies on manually-defined keywords.

- It is possible to adapt our sports-related models to other domains like earthquake detection and still maintain comparable levels of performance to domain-specific systems.

2. RELATED WORK

Though we are focusing on the slightly different problem of identifying "interesting moments" in social media streams, we still build on much of the same work as classical event detection, which has fascinated researchers for over twenty years. This subfield has evolved to integrate the latest available techniques and data sources, starting from early digital newsprint to blogs and now social media. Early stages of this research began in the mid-nineties with the Topic Detection and Tracking (TDT) initiative. These programs demonstrated feasibility in detecting new topics from traditional media sources, but as Allan, Papka, and Lavrenko discussed in 1998, these approaches required additional work to see real success [1]. Even in that early work, Allan et al. were already discussing the tradeoffs of using pre-defined keyword sets and event classes when detecting new events.

Though this early research focused primarily on topic detection from newsprint and traditional media sources, Kleinberg's 2002 paper altered the landscape by applying topic detection to non-traditional data sources like his personal email archive and by introducing what appears to be the first real treatment of burst analysis in "document streams that arrive continuously over time" rather than static collections [9]. Despite introducing the streaming context, Kleinberg cast topic detection as a retrospective, state-based optimization problem. He then leveraged existing work on hidden Markov models (HMMs) to find sequences of high usage keywords, or bursts, from which he could detect events (and construct complex nested states to develop event hierarchies). Kleinberg's examination here laid much of the foundation for the focus on topic bursts that characterized much of the proceeding research in this area.

Following from Kleinberg's work and the increasing size of digital content on the Internet, several new approaches to topic detection emerged. Notably, topic detection divided into two distinct tasks: identifying topics in data via algorithms like Latent Dirichlet Allocation (LDA) [4] and detecting events from text. Our research focuses on events rather than topics, so early event detection work like that from Fung et al. in 2005 is especially interesting [8]. Fung et al. extended Kleinberg's burst detection scheme by identifying bursty keywords from digital newspapers and clustering these keywords into groups to identify bursty events, which displayed success in identifying trending events in an English-language newspaper from Hong Kong.

Soon after, the research community began experimenting with alternative media sources like blogs, but real gains came when microblogging platforms began their rise in popularity. These microblogging platforms include Twitter and Sina Weibo and are characterized by constrained post sizes (e.g., Twitter constrains user posts to 140 characters) and broadcasting publicly consumable information. Since their rise, a great deal of research has explored how data posted to these networks can be leveraged for social good projects like event detection. Much of this work, however, is retrospective in nature and focuses on detecting events after the fact rather than utilizing one of the real advantages in these social networks: unlike newspapers and blogs, these networks produce huge volumes of information that can be processed in *real time*.

One of the most well-known works in detecting events from microblog streams is Sakaki, Okazaki, and Matsuo's 2010 paper on detecting earthquakes in Japan using Twitter [21]. Sakaki et al. show that not only can one detect earthquakes on Twitter but also that it can be done simply by tracking frequencies of earthquake-related tokens. Surprisingly, this approach can outperform geological earthquake detection tools since digital data propagates faster than tremor waves in the Earth's crust. Though this research is limited in that it requires pre-specified tokens and is highly domain- and location-specific (Japan has a high density of Twitter users, so earthquake detection may perform less well in areas with fewer Twitter users), it demonstrates a significant use case and the potential of such applications.

Along with Sakaki et al., 2010 saw two other relevant papers: Lin et al.'s construction of a probabilistic popular event tracker [12] and Petrović, Osborne, and Lavrenko's application of locality-sensitive hashing (LSH) for detecting first-story tweets from Twitter streams [15]. Lin's work is interesting for a number of reasons: first, it circumvents the need for language model-based stop word lists by using probabilistic models to discriminate between common and informational tokens. Secondly, Lin's integration of social and structural features into the event detection task demonstrated that real performance enhancements can be gained through non-textual features. Thirdly, their paper relates well to Kleinberg's initial work on bursty topic detection by illustrating how his state machine approach is a degenerate case of the PET model. Like the majority of its contemporary systems, however, PET also requires seeding with a pre-specified list of tokens to guide its event detection and concentrates on retrospective per-day topics and events.

In contrast, Petrović and his colleagues' clustering research in Twitter avoids the need for seed keywords and retrospective analysis by instead focusing on the practical considerations of clustering large streams of data quickly. That is, rather than construct a probabilistic mixture model for each token, Petrović focuses on methods for clustering tweets that contain similar tokens into topical clusters. While typical clustering algorithms require distance calculations for all pairwise messages, LSH facilitates rapid clustering at the scale necessary to support event detection in Twitter streams by restricting the number of tweets compared to only those within some threshold of similarity. Once these clusters are generated, Petrović was able to track their growth over time to determine impact for a given event. This research was originally unique in that it was one of the early methods that did not require pre-specified seed tokens for detecting events and has been very influential in the field, resulting in a number of additional publications to demonstrate its utility in breaking news and for high-impact crisis events [13, 17, 20]. That being said, a key weakness in Petrović's work is its reliance on semantic similarity between tweets, which limits its ability to operate in mixed-language environments.

Similar to Petrović, Weng and Lee's 2011 paper on EDCoW, short for Event Detection with Clustering of Wavelet-based Signals, is also able to identify events from Twitter without seed keywords [24]. After stringent filtering (removing stop words, common words, and non-English tokens), EDCoW uses wavelet analysis to isolate and identify bursts in token usage as a sliding window advances along the social media stream. Highly significant bursts are then cast as a cross-correlation matrix against which a graph partitioning algorithm is run to construct topical clusters from bursty tokens. Besides the heavy filtering of the input data, this approach exhibits notable similarities with the language-agnostic method we describe herein with its reliance on bursts to detect event-related tokens; the methods described in Weng and Lee's paper, however, operate in a more retrospective manner, focusing on the level of daily news rather than breaking event detection on which our research focuses. Becker, Naaman, and Gravano's 2011 paper on identifying events in Twitter also fall under this label of retrospective analysis, but their findings also demonstrate reasonable performance in identifying events in Twitter by leveraging classification tasks to separate tweets into those on "real-world events" versus non-event messages [3, 2]. Similarly, Diao et al. also employ a retrospective technique to separate tweets into global, event-related topics and personal topics [7].

Beyond these works, several domain-specific research efforts have targeted sporting events specifically[23, 26, 10]. Lanagan and Smeaton's work is of particular interest because it relies almost solely on Twitter's per-second message volume [10]. Though naive, this frequency approach is able to detect large bursts on Twitter in high-impact events without reliance on complex linguist analysis and performs well in streaming contexts as little information must be kept in memory. Unfortunately, two main disadvantages exist with this technique: first, detecting a burst would provide evidence of an event, but it would be difficult to gain insight into what that event actually was without additional processing. Secondly, as described in their paper, a pure volumetric approach is hampered by limitations on Twitter's public stream, which has an upper limit on the number of messages per minute one can capture.

Finally, the most recent work relevant to our research is the 2013 paper by Xie et al. on TopicSketch [25]. Like us, TopicSketch's authors seek to perform real-time event detection from Twitter streams "without pre-defined topical keywords" by maintaining acceleration features across three levels of granularity: individual token, bigram, and total stream. As with Petrović's use of LSH, Xie et al. leverage "sketches" and dimensionality reduction to facilitate event detection and also relies on language-specific similarities. Furthermore, Xie et al. focus only on tweets from Singapore rather than the worldwide stream. In contrast, our approach is differentiated primarily in its language-agnosticism and its use of the unfiltered stream from Twitter's global network.

Despite this extensive body of research, it is worth asking how event detection on Twitter streams differs from Twitter's own offerings on "Trending Topics," which they make available to all their users. When a user visit's Twitter's website, she is immediately greeted with her personal feed as well as a listing of trending topics for her city, country, worldwide, or nearly any location she chooses. These topics offer insight into the current popular topics on Twitter, but the main differentiating factor is that these popular topics are not necessarily connected to specific events. Rather, popular memetic content like "#MyLovelifeInMoveTitles" often appear on the list of trending topics. Additionally, Twitter monetizes these trending topics as a form of advertising [22]. These trending topics also can be more high-level than the interesting moments we seek to identify: for instance, during the World Cup, particular matches or the tournament in general were identified as trending topics by Twitter, but individual events like goals or penalty cards in those matches were not. It should be clear then that Twitter's trending topics serves a different purpose than the streaming event detection described herein.

3. STREAMING EVENT DETECTION

This paper's primary goal is to demonstrate the feasibility of detecting highly interesting moments from social media streams by identifying bursting tokens and analyzing their frequencies (we refer to this language-agnostic, burst-centric technique as LABurst). Since compelling moments often align with significant events, we compare LABurst's performance with that of a baseline seed-token technique in detecting specific events within a variety of major sporting competitions. This comparison shows our more general approach can still perform on par with existing domain-specific methods. Then, we show applications outside the sports domain by applying LABurst to the task of detecting earthquakes using social media similar to Sakaki et al. This section defines these detection tasks, presents our experimental framework, details the data sources used, and outlines both the baseline technique and our LABurst method.

3.1 Problem Definition and Evaluation

Given the ambiguities in the current literature's definition of an "event" (varying definitions from existing works like Allan et al. to an entire ontology devoted to describing the term [19, 1]), we instead focus on identifying interesting moments in time. This task is difficult because one may not know tokens relevant to these brief instances until the moment occurs, and by the time one can react, the moment has passed. Such impactful moments are often accompanied with significant and sudden increases, or bursts, in message volume and bursts in usage of specific tokens (we refer to such tokens as "bursty"). Where the typical method relies on tracking predefined tokens and identifying points in which those tokens experience bursts, we instead identify these bursty signals across all tokens via machine learning and use their frequency as the primary indicator that a moment of high interest is occurring. As mentioned, since many events experience such bursts, we subsume a number of event detection tasks with this generalization.

More formally, if we let E denote the set of all minutes t in which an interesting moment occurs, then the indicator function $\mathbb{1}_E(t)$ returns a 1 for all times t in which such a moment occurs, and 0 for all other values of t. We can then define the moment detection task as constructing a function that approximates this indicator function $\mathbb{1}_E(t)$. To account for possible lag in experiencing the event, typing out a message about the event, and the message actually posting to a social media server, we include a parameter τ to control for tolerance of delay. This parameter relaxes the task slightly by constructing the set E' where, for all $t \in E$, $t, t+1, t+2, ..., t+\tau \in E'$. Since our evaluation is a comparison between two methods that share the same ground truth, and controlling τ affects the ground truth consistently for both methods, comparative results should be unaffected. In this paper, we use $\tau = 2$.

False positives/negatives and true positives/negatives follow in the normal way for some candidate function $\widehat{\mathbb{1}_{E'}}(t)$: a false positive is any time t such that $\widehat{\mathbb{1}_{E'}}(t) = 1$ and $\mathbb{1}_{E'}(t) = 0$; likewise, a false negative is any t such that $\widehat{\mathbb{1}_{E'}}(t) = 0$ and $\mathbb{1}_{E'}(t) = 1$. True positives/negatives then follow as expected.

With these definitions, we compare LABurst and the baseline by constructing a series of receiver operating characteristic (ROC) curves across several individual test sets and a composite ROC curve for all testing data. We then evaluate relative performance between the two approaches by comparing their respective areas under the curves (AUCs) by varying threshold parameters for each method. In the baseline approach, our threshold parameter controls the minimum difference between current frequency and average frequency over the sliding window. For our LABurst method, the ROC curve is generated by varying the minimum number of tweets a window must contain for an event to be detected. The AUC of the ROC curve is useful in this instance because it is robust against imbalanced classes, which we expect to see in the event detection task.

We then follow up with an additional experiment of whether LABurst can detect moments in which an earthquake strikes as inspired by Sakaki et al. Rather than generating ROC curves for this comparison, we take a more straightforward approach and compare lag between the actual earthquake event and the point in time in which the two methods detect the earthquake.

3.2 Experimental Framework

Having established the vocabulary and tasks, we can now turn our attention to the actual mechanics of comparing our LABurst technique with the baseline. Sporting competitions are full of interesting and unpredictable moments, such as scores, fouls, ejections, or other dramatic instances of play. Additionally, sporting events with large followings occur fairly often and with a good degree of regularity, greatly simplifying data collection.

Our main experiment is a comparative study in detecting a collection of events in three major sporting competitions: the 2013 Major League Baseball (MLB) World Series, 2014 National Football League (NFL) Super Bowl, and 2014 Fédération Internationale de Football Association (FIFA) World Cup. Specifically, for the 2013 World Series between the Boston Red Sox and the St. Louis Cardinals, we explore the final two games on 28 October and 30 October of 2013. Similarly for the 2014 World Cup, our analysis covers the final two matches of tournament: the 12 July match between the Netherlands and Brazil for third place, and the final match on 13 July between Germany and Argentina for first place. Since the 2014 Super Bowl was a single-day competition, we covered the entire game between the Seattle Seahawks and the Denver Broncos on 2 February 2014.

In each competition, we extract the times of four basic types of events: beginning of the game, end of the game, scores, and penalties. We capture these events and times from sport journalism articles, game highlights, box scores, blog posts, and social media messages. Table 1 provides some statistics on the events we identified.

Table 1: Sporting Competition Event Counts

Sport	Start	End	Score	Penalty	Sum
World Series	2	2	10	0	14
Super Bowl	1	1	10	0	12
World Cup	2	2	4	9	17
				Total	43

These events then comprise the ground truth against which we compare LABurst and the baseline. That is, positive instances of events are those minutes in which one of these events occurs, and negative instances are those minutes in which no known sports-related event occurs. This data was also excluded from development such that neither the baseline nor LABurst are trained on or evaluated against this data prior to experimentation.

For the earthquake detection task, we compare LABurst with the frequency of the keyword "earthquake" by applying LABurst's models learned in the first experiment to social media captured during two large earthquakes: the 7.1-magnitude quake off the coast of Honshu, Japan on 25 October 2013, and a 6.5-magnitude quake off the coast of Iwaki, Japan on 11 July 2014.

3.2.1 Data Collection and Training Events

Our data on the main sporting events used for evaluation (the World Series, Super Bowl, and World Cup) come from Twitter's 1% public sample stream using code from Jimmy Lin's twitter-tools library[1]. For training LABurst, however, we require additional data to model bursty events, so we leverage two other Twitter data sources as well: an excerpt from the Edinburgh Twitter Corpus [16] and a selection of tweets from the Twitter Firehose covering Argentina in November of 2011. From these sources, we are able to generate a series of 164 timestamped events and related tokens for several additional sporting events as well:

- The 2010 NFL National Football Championship game,
- Four premier league soccer games in November of 2012,
- The National Hockey League's (NHL) 2014 playoffs,
- The National Basketball Assoc. (NBA) 2014 playoffs,
- The 2014 Kentucky Derby and Belmont Stakes races,
- And six early days in the 2014 FIFA World Cup.

3.3 Baseline Detector

Inspired by Twitter's blog post on detecting goals during the World Cup, we constructed a simple detector that uses per-minute frequencies for a small set of seed tokens [6]. These seed tokens are likely to exhibit bursts in usage during events, such as "goal" for goals in soccer/football or hockey or "run" for runs scored in baseball. Our baseline implementation also includes some rudimentary normalization to collapse modified words to their originals (e.g., "gooaalll" down to "goal"). We know such a technique is effective because, as seen in the related work, many existing stream-based event detection systems use just such an approach to track specific types of events.

[1] https://github.com/lintool/twitter-tools

At a high level, we compare the most recent count of these target tokens against the average count over the past few minutes, and if the difference is above some threshold, we claim an event occurred. Formally, we define the following: a time series T segmented into m minutes, a set of event-related seed tokens S such that $s_i \in S$ is one of these event-related tokens, and a function $\texttt{count}(s_i, t_j)$ that returns the frequency of token s_i in minute t_j. The frequency for a given minute t_j is then defined by the function $\texttt{freq}(t_j)$ shown in Eq. 1. We also construct a sliding window w of size $|w|$ such that w_k includes minutes t_k to $t_{k+|w|-1}$ and define an average over this window as $\texttt{avg}(w_k)$, shown in Eq. 2.

$$\mathrm{freq}(t_j) = \sum_{i=0}^{|S|} \mathrm{count}(s_i, t_j) \tag{1}$$

$$\mathrm{avg}(w_k) = \frac{\sum_{j=k}^{k+|w|-1} \mathrm{freq}(t_j)}{|w|} \tag{2}$$

Given these functions, we take the difference Δ_k between the frequency at time $t_{k+|w|-1}$ and the average for window w_k such that $\Delta_k = \texttt{freq}(t_{k+|w|-1}) - \texttt{avg}(w_k)$. If this difference exceeds some threshold Z such that $\Delta_k > Z$, we say an event was detected at time $t_{k+|w|-1}$.

Since our analysis covers three separate types of sporting competitions, the seed keyword list for this method should include tokens from the vocabulary of each. We avoid separate keyword lists for each sport to provide an even comparison to the general nature of our language-agnostic technique. The keywords for which we searched are shown in Table 2, and we took the union of these token sets. Additionally, the following regular expressions collapsed deliberately misspelled tokens down to their normal counterparts: "g+o+a+l+" →"goal", "g+o+l+" →"gol", "g+o+l+a+z+o+" →"golazo", "sco+red?" →"score".

Table 2: Predefined Seed Tokens

Sport	**Tokens**
World Series	"run", "home", "homerun"
Super Bowl	"score", "touchdown", "td", "fieldgoal", "points"
World Cup	"goal", "gol", "golazo", "score", "foul", "penalty", "card", "red", "yellow", "points"

3.4 Language-Agnostic Moment Discovery

As mentioned, existing research achieves acceptable performance by tracking frequencies for a small set of relevant keywords when a real-world event occurs, but this approach has several disadvantages. We extend such approaches and obviate the need for pre-specification by automatically identifying bursty tokens and detecting interesting moments based on their frequencies. To this end, we constructed a set of temporal features to model the characteristics of a bursty token. Like other approaches, these temporal features were built around a sliding window of several minutes with each window further divided into overlapping slices. In this manner, we construct a windowed time series containing the most recent frequencies for each token, from which our features are generated.

From these features, we use AdaBoost to combine support vector machines (SVMs) and random forests (RFs) into an ensemble classifier to discriminate between bursty and non-bursty tokens. To cast this task as a machine learning problem, however, we require positive and negative token samples on which our classifier can be trained. While obtaining positive samples of bursty tokens is straightforward (we can at least use keywords from the baseline as well as event-specific tokens like names of scoring players or locations of events), determining negative samples is difficult since one cannot know all events that may burst on Twitter at a given moment. We circumvent this difficulty by automatically classifying all stop words as negative, non-bursty tokens, and Python's NLTK[2] library provides convenient lists of stop words for several languages.

It is important to note that we perform this classification without any semantic or language-based filtering or normalization, and the only data we discard in this analysis are retweets and hashtags.

3.4.1 Temporal Features

While previous work already covers burst detection, only a few are extensible to the streaming case, and we integrate those into our system. We also develop a number of features that should yield higher weights for tokens that deviate significantly from normal posting frequencies:

- **Frequency Regression** Given the log of a token's frequency at each slice in the current window, take the slope of the best-fitting line.
- **Message Frequency Regression** Given the log of the number of tweets in which a token appears for each slice in the current window, take the slope of the best-fitting line.
- **User Frequency Regression** Given the log of the number of users using a token at each slice in the current window, take the slope of the best-fitting line.
- **Average Frequency Difference** The difference between the token's frequency in the most recent slice and the average frequency across the current window.
- **Message Average Frequency Difference** The difference between the number of messages in which a token appears in the most recent slice and the average number of messages containing that token across the current window.
- **User Average Frequency Difference** The difference between the number of users who use a token in the most recent slice and the average number of users across the current window.
- **Inter-Arrival Time** The average number of seconds between token occurrences in the given window.
- **Entropy** The entropy of the set of tweets containing a given token.
- **TF-IDF** The term frequency, inverse document frequency for a each token.

[2] http://www.nltk.org

- **TF-PDF** A modified version of TF-IDF called term frequency, proportional document frequency [5].

- **BursT** Weight using a combination of a given token's actual frequency and expected token frequency [11].

We normalize each window's feature vectors into the range $[0, 1]$ to avoid biases from scale during classification by taking the maximum and minimum values for each feature in the current window.

3.4.2 *Training the Ensemble Classifier*

Discriminating between bursty tokens and stop words necessitates the use of a classification algorithm, and many different such algorithms exist. In particular, the Scikit-learn Python package provides implementations for SVMs and RFs as well as an implementation of the ensemble classifier AdaBoost [14]. Both SVMs and RFs have tunable parameters to select before integrating into AdaBoost, however, so we employed a grid search strategy to select parameters based on the F1 scores on our training and testing data.

For SVMs, we first had to decide whether to use a traditional linear model or employ a kernel. Initial experiments showed linear SVMs performed quite poorly, and after using principal component analysis to reduce the dimension and looking at the labeled data, it fit well in a sphere rather than a clear linear plane. As a result, we use the radial basis kernel, which has two parameters: cost c and kernel coefficient γ. In searching the space of c and γ, the grid covers powers of two such that $c = 2^x$, $x \in [-2, 10]$ and $\gamma = 2^y$, $y \in [-2, 5]$. For each pair of parameter values, we train thirty different classifiers using repeated random subsampling, take the average F1 score, and select the parameter set with the highest F1 score. Selecting parameter values for RFs is similar for the number of estimators n and feature count c' such that $n = 2^x$, $x \in [0, 10]$ and $c' = 2^y$, $y \in [1, 11]$. This training procedure yielded the results shown in Table 3.

Table 3: Classifier Parameter Scores

Classifier	Params	F1-Score
SVM	$c = 64$, $\gamma = 4$	0.588410

RF	trees = 128,
	features = 9
		0.575301

These two classifiers are then combined using the Scikit-learn's AdaBoost implementation with four estimators. We apply the resulting classifier to all training data and expand our set of known bursty tokens with those tokens that had a greater than 90% likelihood of being part of the bursty class in one round of self-training (Scikit's AdaBoost implementation provides label likelihoods). Regarding sliding window and slice size, preliminary investigations suggest a window size of ten minutes with a slice size of three minutes (each slice overlapped the next by two minutes) exhibited acceptable results.

4. EXPERIMENTAL RESULTS

To restate, the research question posed in this work is to determine whether a language-agnostic scheme for detecting moments of high interest can perform as well as a domain-specific frequency-based method in detecting events in sporting competitions. We answer this question across three separate sporting events: the final two games of the 2013 MLB World Series, the 2014 NFL Super Bowl, and the final two matches of the 2014 FIFA World Cup. These events were completely new data sets for both the baseline and LABurst and were not included in training, so any events detected by either algorithm were previously unseen events.

For each sporting competition, we generated ROC curves by varying the threshold parameter and calculating the true and false positive rates on detection corresponding to the indicator function $\mathbb{1}_E(t)$ described above. That is, the true positive rate equals the number of minutes in which an event both was detected *and* occurred versus the number of minutes in which an event actually occurs. Similarly, the false positive rate is the number of minutes in which the algorithm detects an event when no event actually occurred versus the total number of minutes in which no events occurred.

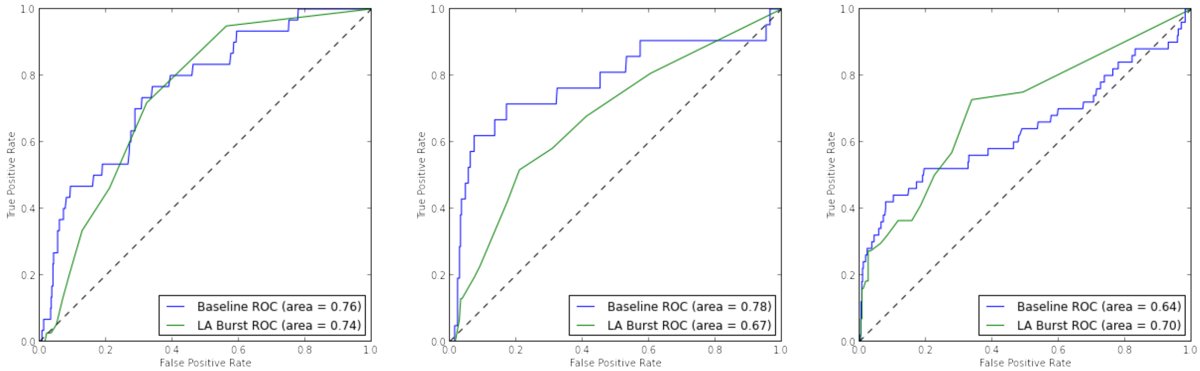
Prior to presenting comprehensive results, we first present performance curves for each event type. For the 2013 World Series, LABurst and the baseline techniques exhibited similar performance, with a difference of only 0.02 (the baseline with 0.76 and the language-agnostic bursty method with 0.74). Figure 1a shows the how these two curves compare graphically, and we can see that neither curve completely dominates the other, and both perform better than random guessing. In the Super Bowl, the difference between the two mechanisms (shown in Figure 1b) is more pronounced with a difference in AUC near 0.1, with the baseline performing better. LABurst exhibited a much higher false-positive rate in comparison to the baseline, which may be explained later in 5.2. Unlike the World Series and Super Bowl, however, the difference between the baseline and LABurst during the 2014 World Cup as seen in Figure 1c shows our LABurst method actually outperforms the baseline in this instance with a difference in AUC of approximately 0.05.

4.1 Composite Results

To compare comprehensive performance, we look to Figure 2, which shows ROC curves for both methods across all three event types. From this figure, we see the two methods perform nearly identically with AUC values of 0.7187 for the baseline and 0.7052 for our language-agnostic technique. Assuming equal cost for false positives and false negatives and optimizing for the largest difference between true positive rate (TPR) and false positive rate (FPR), the baseline method shows a TPR of 0.5581 and FPR of 0.1408 with a difference of 0.4174 at a threshold value of 13.2. Our language-agnostic method, on the other hand, has a TPR of 0.7105 and FPR of 0.3518 with a difference of 0.3587 at a threshold value of 2. From these values, we see our approach achieves a higher true positive rate but at a cost of a higher false positive rate as a result.

4.2 Earthquake Detection

Detecting interesting moments within sporting competitions as described above is a useful task for areas like advertising or automated highlight generation, but a more compelling and worthwhile task would be to detect higher-



(a) 2013 World Series

(b) 2014 Super Bowl

(c) 2014 World Cup

Figure 1: Per-Sport ROC Curve Performance

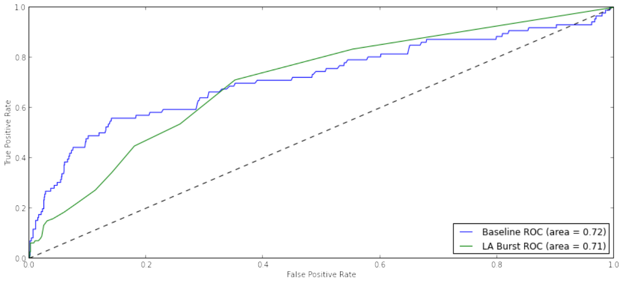


Figure 2: Composite ROC Curves

impact events like natural disasters. The typical frequency-based approach is difficult here as it is impossible to know what events are about to happen where, and a list of target keywords to detect all such events would be long, leading to false positives. Our method could be highly beneficial here as one need not know the target language or other pre-specified information. Since Sakaki showed the feasibility of detecting earthquakes using Twitter, we pulled Twitter data for two earthquakes in Japan: a 7.1-magnitude quake off the coast of Honshu on 25 October 2013, and a 6.5-magnitude quake off the coast of Iwaki on 11 July 2014.

To determine whether our approach could detect these earthquake events, we applied the classifier trained and tested for the sporting domain to these Twitter sets and tracked the frequency of the term "earthquake" simultaneously. Figures 3a and 3b show the frequencies for both methods for the 2013 and 2014 earthquakes respectively; the red dots indicate the earthquake times as reported by the United States Geological Survey (USGS). From these figures, one can see the token "earthquake" sees a significant increase in usage when the earthquake occurs, and our language-agnostic method experiences a similar increase at the same moment for both events. That is, both techniques identify the earthquake simultaneously.

Given our method's success here, one can now ask what tokens we identified as bursting when the earthquakes occurred. Many of the tokens are in Japanese, and tokens at the peak of the earthquake events are shown in Table 4. We also extracted several tweets that contain the highest number of these tokens for the given time period, a selection of which include, "地震だあああああああああああああああああああああ," "今回はチト使ってないから地震わからなかった," and "地震だ." Google Translate[3] translates these tweets as "Ah ah ah ah ah ah ah ah ah Aa's earthquake," "I did not know earthquake because not using cheat this time," and "Over's earthquake" respectively.

Table 4: Tokens Classified as Busting During Events

Match	Bursty Tokens
Honshu, Japan – 25 October 2013	çdostum, 丈, 地, 夫, 怖, 波, 注, 津, 源, 福, 震
Iwaki, Japan – 11 July 2014	antojo, comida, sammy, び, ゆ, ビビ, 地, 怖, 急, 福, 警, 速, 震

5. ANALYSIS

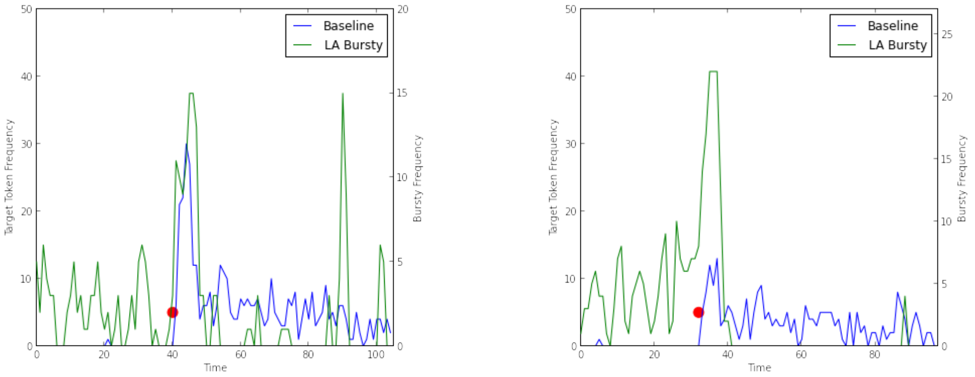
In comparing the baseline and language-agnostic techniques, it is important to understand the baseline provides little in the way of discovering previously unknown tokens or significant events that do not conform to a priori knowledge. Our language-agnostic method's real advantage is that it not only directly supports this notion of discovery but also provides direct insight into these moments of high interest by tagging relevant bursting tokens.

5.1 Identifying Event-Related Tokens

As mentioned, where the baseline requires the user to specify interesting or event-related tokens prior to any data processing or analysis, our approach identifies these event-related tokens automatically. These tokens may include misspellings, colloquialisms, and cross language boundaries, which makes them hard to know before hand. The 2014 World Cup presents an interesting case for finding these otherwise unexpected tokens because the event has enormous international viewership; as such, many Twitter users of many different languages are likely tweeting about the same event.

To explore the tokens generated during these high-profile events, we look to those tokens identified as bursting during

[3] http://translate.google.com



(a) Honshu, Japan Earthquake - 25 October 2013

(b) Iwaki, Japan Earthquake - 11 July 2014

Figure 3: Japanese Earthquake Detection

several events in the final two World Cup matches. Table 5 shows a selection of events from these matches and a subset of those tokens classified as bursting during the events (one should note the list is not exhaustive owing to formatting and space constraints).

Table 5: Tokens Classified as Busting During Events

Match	Event	Bursty Tokens
Brazil v. Netherlands, 12 July 2014	Netherlands' Van Persie scores a goal on a penalty at 3', 1-0	0-1, 1-0, 1:0, 1x0, card, goaaaaaaal, goal, gol, goool, holandaaaa, kırmızı, pen, penal, penalti, pênalti, persie, red
Brazil v. Netherlands, 12 July 2014	Brazil's Oscar get's a yellow card at 68'	dive, juiz, penalty, ref
Germany v. Argentina, 13 July 2014	Germany's Götze scores a goal at 113', 1-0	goaaaaaallllllll, goalllll, godammit, goetze, gollllll, gooooool, gotze, gotzeeee, götze, nooo, yessss, ドイツ

Several interesting artifacts emerge from this table, first of which is that one can get an immediate sense of the detected event from tokens our algorithm presents. For instance, the prevalence of the token "goal" and its variations clearly indicate a team scored in the first and third events in Table 5; similarly, bursting tokens associated with the middle event regarding Oscar's yellow card reflect his penalty for diving. Beyond the pseudo event description put forth by the identified tokens, this reference to diving and to specific player and teams names in the first and third events are also of significant interest. In the first event, one can infer that the Netherlands scored since "holandaaaa" is flagged along with "persie" from the Netherlands' player Van Persie, and likewise for Germany's Götze in the third event (and the accompanying variations of his name). These terms would be difficult to capture beforehand as would be required in the baseline and would likely not be related to every event or every type of sporting event.

Finally, the last artifact of note is that the set of bursty tokens displayed includes tokens from several different languages: English for "goal" and "penalty," Spanish for "gol" and "penal," Brazilian Portuguese for "juiz" (meaning "referee"), as well as the Arabic for "goal" and Japanese for "Germany." Since these words are semantically similar but syntactically dissimilar, typical normalization schemes could not capture these connections. Instead, capturing these words in the baseline would require a pre-specific keyword list in all possible languages or the inclusion of an expensive machine translation system that was also capable of normalizing within different languages (to collapse "goool" down to "gol" for example).

5.2 Discovering Unanticipated Moments

One particular weakness present in the baseline is that it is unable to capture unexpected events or events that do not conform to the keyword list. This deficiency means analysts might miss significant events within these competitions, especially if they are not directly related to scores or penalties, such as Uruguay's Luis Suarez's biting the Italy's Giorgio Chiellini during a World Cup match on 24 June since no foul was called at the time. Other instances of particularly dramatic play or events that happen at the larger sporting event but not necessarily on the field might be missed as well.

We can see instances of such omissions in the last game of World Cup. Figure 4 shows the frequencies for target tokens for the baseline in blue and bursty tokens for our method in green. From this graph, we can see the first, obvious incidence in Peak #1 where our bursty method exhibits a peak that is missed by the baseline in the first few points of data. The primary tokens appearing in this peak are "puyol," "gisele," and "bundchen," which correspond to former Spanish player Carles Puyol and model Gisele Bundchen, who presented the World Cup trophy prior to the match. At peak #2, slightly more than eighty minutes into the data (which is sixty minutes into the match), our burst analysis sees another peak that is otherwise relatively minor in the baseline graph. Upon further exploration, tokens present in this peak refer to Argentina's substituting in Agüero for Lavezzi at the beginning of the match's second half.

Given our detection of these moments of interest that do not related to score or penalty events, it is perhaps unsurprising that our burst detection technique exhibits a higher

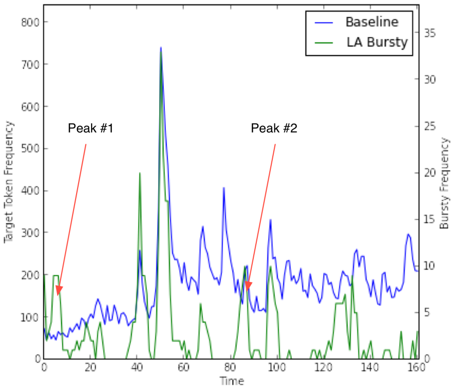


Figure 4: Baseline and LA Bursty Frequencies

false-positive rate compared to the baseline. Since our approach is both language- and domain-agnostic, it makes sense that it would detect additional events beyond the game start/end, score, and penalty events our data counts as ground truth. A better comparison between the accuracies of the two approaches may be to identify only those peaks in which the keywords used in the baseline are identified as bursty in our method, but such a test disregards our approach's additional power.

5.3 Real-Time Usage and Event Persistence

In comparing these two event detection methods, we must also address their abilities to handle streaming data and the lag between an event and its detection by either of these mechanisms. The baseline technique processes data minute by minute and therefore has at most a minute of lag between input and discovery. Our technique, on the other hand, exhibits a lag of at most the slice size, which in the case of this paper is three minutes.

Another important aspect to consider is the length of time in which an event's peak persists. For the baseline, event detection achieves its highest accuracy when events are flagged in the minute they occur and the minute immediately following. Our approach logically follows the slice length such that events persist for approximately three minutes.

6. FUTURE WORK

While the experiments outlined herein establish the utility of language-agnostic event detection, several avenues of research can follow up on this foundation. First, though this work is applied in the streaming context, little effort was made to enforce near-real-time computation constraints; with the growing popularity of stream-centric processing frameworks like Apache Storm (as used by Petrović et al.) or Apache Spark Streaming, one has considerable latitude in exploring ways to enforce such constraints. Secondly, our selection of classifiers used in our ensemble was based primarily on ease of use given our features; deep learning systems are more complex but could provide enhanced capabilities in detecting bursting patterns in social media streams. Thirdly, we specifically targeted moments of high interest, which often correspond to instantaneous events, but it is possible that these same techniques could be applied at different levels of granularity (per hour or per day for instance) to detect events of larger scales; one could then apply Kleinberg's notion of event hierarchies across multiple temporal granularities and track different aspects of the same event. Additionally, an implicit assumption made in this work is that tokens that experience bursts at the same time are related in some way, which allows us to detect events using token frequency; this assumption has interesting consequences with regarding to language-agnostic topic detection. That is, one could cluster tokens with similar temporal signatures to identify topics across languages, which would otherwise be impossible if one were to rely solely on semantic similarity or would require applying machine translation to reduce all text in the stream to the same language. Finally, though this paper exclusively leverages data from Twitter's public stream, our techniques should be applicable to streams from other social networks as well, and how events burst on more media-centered networks like Flickr or Pinterest might reveal interesting photographic representations of an event.

7. CONCLUSIONS

To revisit our motivations, the goal for this experiment was to demonstrate the feasibility of detecting highly compelling or interesting moments through analyzing temporal characteristics from unfiltered Twitter data streams. While many social media-based event detection systems require some form of prespecified seed list and/or language model processing, our approach is more flexible, lighter weight, and easily adaptable to different domains. Our results show that by leveraging temporal characteristics to identify bursty tokens and using frequency of these bursty tokens, we can detect significant events across a collection of disparate sporting competitions with a level of performance nearly equivalent to an existing, domain-specific baseline.

Similar performance to the baseline is only part of the story, however, as our approach offers notable flexibility in identifying bursting tokens without normalization and across language boundaries. With this versatility also comes support for event description since we no longer rely on predetermined keywords; that is, we can get a sense of the occurring event by inspecting the bursty tokens. Finally, these advantages culminate in powerful tool for event *discovery* in that it can unanticipated instances of high interest that we did not expect, regardless of the source language, which makes this technique particularly useful for journalists and newswire sources who have a need to know about events on the ground, as they happen but cannot know a priori what the event may be about in all cases.

8. ACKNOWLEDGMENTS

This work made use of the Open Science Data Cloud (OSDC), which is an Open Cloud Consortium (OCC)-sponsored project. The OSDC is supported in part by grants from Gordon and Betty Moore Foundation and the National Science Foundation and major contributions from OCC members like the University of Chicago.

9. REFERENCES

[1] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on*

Research and development in information retrieval, pages 37–45. ACM, 1998.

[2] H. Becker, M. Naaman, and L. Gravano. Beyond Trending Topics: Real-World Event Identification on Twitter. *ICWSM*, 11:438–441, 2011.

[3] H. Becker, M. Naaman, and L. Gravano. Beyond Trending Topics: Real-World Event Identification on Twitter - Technical Report. 2011.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[5] K. K. Bun and M. Ishizuka. Topic Extraction from News Archive Using TF*PDF Algorithm. In *Proceedings of the 3rd International Conference on Web Information Systems Engineering*, WISE '02, pages 73–82, Washington, DC, USA, 2002. IEEE Computer Society.

[6] L. Cipriani. Goal! Detecting the most important World Cup moments. Technical report, Twitter, 2014.

[7] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 536–544. Association for Computational Linguistics, 2012.

[8] G. P. C. Fung, J. X. Yu, P. S. Yu, and H. Lu. Parameter free bursty events detection in text streams. In *Proceedings of the 31st international conference on Very large data bases*, VLDB '05, pages 181–192. VLDB Endowment, 2005.

[9] J. Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 91–101, New York, NY, USA, 2002. ACM.

[10] J. Lanagan and A. F. Smeaton. Using twitter to detect and tag important events in live sports. *Artificial Intelligence*, pages 542–545, 2011.

[11] C.-H. Lee, C.-H. Wu, and T.-F. Chien. BursT: a dynamic term weighting scheme for mining microblogging messages. In *Proceedings of the 8th international conference on Advances in neural networks - Volume Part III*, ISNN'11, pages 548–557, Berlin, Heidelberg, 2011. Springer-Verlag.

[12] C. X. Lin, B. Zhao, Q. Mei, and J. Han. PET: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 929–938, New York, NY, USA, 2010. ACM.

[13] M. Osborne, S. Moran, R. McCreadie, A. Von Lunen, M. Sykora, E. Cano, N. Ireson, C. Macdonald, I. Ounis, Y. He, and Others. Real-Time Detection, Tracking, and Monitoring of Automatically Discovered Events in Social Media. *Association for Computational Linguistics*, 2014.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[15] S. Petrović, M. Osborne, and V. Lavrenko. Streaming First Story Detection with Application to Twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 181–189, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[16] S. Petrović, M. Osborne, and V. Lavrenko. The Edinburgh Twitter Corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pages 25–26, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[17] S. Petrovic, M. Osborne, R. McCreadie, C. Macdonald, I. Ounis, and L. Shrimpton. Can Twitter replace Newswire for breaking news? In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, volume 2011, 2013.

[18] C. Pring. 100 social media statistics for 2012. *TheSocialSkinny.com*, Jan. 2012.

[19] Y. Raimond and S. Abdallah. The Event Ontology, 2007.

[20] J. Rogstadius, M. Vukovic, C. A. Teixeira, V. Kostakos, E. Karapanos, and J. A. Laredo. CrisisTracker: Crowdsourced social media curation for disaster awareness. *IBM Journal of Research and Development*, 57(5):4:1–4:13, Sept. 2013.

[21] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM.

[22] L. Sydell. How Twitter's Trending Algorithm Picks Its Topics, Dec. 2011.

[23] V. Vasudevan, J. Wickramasuriya, S. Zhao, and L. Zhong. Is Twitter a good enough social sensor for sports TV? In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pages 181–186. IEEE, 2013.

[24] J. Weng and B.-S. Lee. Event Detection in Twitter. In *ICWSM*, 2011.

[25] W. Xie, F. Zhu, J. Jiang, E.-p. Lim, and K. Wang. TopicSketch: Real-time Bursty Topic Detection from Twitter. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 837–846. IEEE, 2013.

[26] S. Zhao, L. Zhong, J. Wickramasuriya, and V. Vasudevan. Human as Real-Time Sensors of Social and Physical Events: A Case Study of Twitter and Sports Games. *CoRR*, abs/1106.4, 2011.