Discovering Key Moments in Social Media Streams

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

Jimmy Lin
Cheriton School of Computer Science
University of Waterloo
Waterloo, Ontario
Email: jimmylin@uwaterloo.ca

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

Abstract—**This paper introduces a general technique, called LABurst, for identifying key moments, or moments of high impact, in social media streams without the need for domain-specific information or seed keywords. We leverage machine learning to model temporal patterns around bursts in Twitter's unfiltered public sample stream and build a classifier to identify tokens experiencing these bursts. We show LABurst performs competitively with existing burst detection techniques while simultaneously providing insight into and detection of unanticipated moments. To demonstrate our approach's potential, we compare two baseline event-detection algorithms with our language-agnostic algorithm to detect key moments across three major sporting competitions: 2013 World Series, 2014 Super Bowl, and 2014 World Cup. Our results show LABurst outperforms a time series analysis baseline and is competitive with a domain-specific baseline even though we operate without any domain knowledge. We then go further by transferring LABurst's models learned in the sports domain 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.**

I. Introduction

Though researchers have presented many adaptations of social media into sources for journalists or first responders, many current approaches rely on prior knowledge and keyword engineering to detect events of interest. While straightforward and capable, such approaches are often constrained to events one can easily anticipate or describe in general terms, potentially missing impactful but *unexpected* key moments. For instance, one can follow the frequency of "goal" on Twitter during the 2014 World Cup to detect when goals are scored [1], but interesting occurrences like penalties or missed goals would be missed. One might respond to this weakness by tracking additional penalty-related tokens, but one cannot continually enlarge the keyword set for all cases. Furthermore, one would still be unable to identify an unexpected moment 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? Relying on predefined keywords also restricts these systems to languages represented by the seed keywords, a significant issue for international events like the World Cup.

Given social media's high volume, one could forgo seed keywords and leverage time series analysis to track bursts in message frequency (as with Vasudevan et al. [2]). Such methods gain flexibility by sacrificing semantic information about detected events (one would need to extract keywords causing such bursts manually). In this paper, we propose leveraging machine learning to combine both techniques.

To explore this integration, we introduce LABurst (for language-agnostic burst detection), a general method to model bursts in token usage in social media streams. Burst volume indicates the presence of a high-impact or key moment, such that as more tokens experience a simultaneous burst, the higher the impact of that moment. Contrasting with existing work, our approach is a streaming algorithm for unfiltered social media streams that discovers high-impact moments without prior knowledge of the target event and yields a description of the discovered moment. We illustrate this flexibility with experiments on Twitter's sample stream surrounding key moments in large sporting competitions and natural disasters. These experiments compare LABurst to two existing methods: a time series-based burst detection technique, and a domain-specific technique with a pre-determined set of relevant keywords. Results from these experiments demonstrate LABurst's competitiveness with existing methods. This work makes the following contributions:

- Presents a streaming algorithm and feature set for the discovery and description of impactful and unexpected key moments in Twitter's public sample stream without requiring manually defined keywords,
- Demonstrates LABurst's performance is both competitive and flexible, and
- Transfers sports-trained models to the higher-impact domain of disaster response.

II. Related Work

Though LABurst focuses on the alternative problem of discovering interesting moments in social media streams, our work shares foundations with classical event detection research. Extracting events from digital media has fascinated researchers for over twenty years, from digital newsprint to blogs and now social media [3]. Early research followed that of Fung et al., whose 2005 work built on burst detection schemes presented by Kleinberg to identify and cluster bursty keywords from newspapers to detect bursty or trending events [4], [5]. Recognizing this newsprint differed substantially from social media both in content and velocity, the research community began experimenting with new social media sources like blogs, but real gains came when microblogging platforms rose in popularity.

One of the most well-known works in detecting events from microblogs is Sakaki, Okazaki, and Matsuo's 2010 paper on detecting earthquakes in Japan [6]. Sakaki et al. showed one can detect earthquakes on Twitter simply by tracking

978-1-4673-9292-1/16/$31.00 ©2016 IEEE

frequencies of earthquake-related tokens. Surprisingly, this approach outperformed some geological earthquake alert tools since digital data propagates faster than tremor waves in the Earth's crust. Though this research was 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 demonstrated a significant use case and the potential of such applications.

Along with Sakaki et al., 2010 saw two other relevant papers: a probabilistic popular event tracker by Lin et al. [7] and Petrović, Osborne, and Lavrenko's application of locality-sensitive hashing (LSH) for detecting first-story tweets from Twitter streams [8]. Lin's work integrated non-textual social and structural features into event detection to enhance performance. Like many contemporary systems, these models required pre-specified tokens to guide event detection and concentrated on retrospective per-day topics and events. In contrast, Petrović et al.'s clustering research avoided the need for seed keywords and retrospective analysis by addressing the practical considerations of clustering large streams at scale. Once these clusters are generated, Petrović tracked their growth over time to determine a given event's impact. This research was one of the early methods that did not require seed tokens for detecting events and has been very influential, resulting in a number of additional publications [9], [10]. Petrović's work and related semantic clustering approaches rely on textual similarity between tweets, however, which limits operation in mixed-language environments, thereby differentiating LABurst and its language agnosticism.

Similar to Petrović, Weng and Lee's 2011 paper on Event Detection with Clustering of Wavelet-based Signals (EDCoW), also identified events from Twitter without seed keywords [11]. After stringent filtering (removing stop words, common words, and non-English tokens), EDCoW used wavelet analysis in a sliding window to isolate and identify bursts in token usage. This method operated retrospectively, focusing on 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 falls under retrospective analysis; their findings demonstrated success in identifying events in Twitter by leveraging classification tasks to separate tweets into those about real-world events versus endogenous events [12]. Similarly, Diao et al. employed a retrospective technique to separate tweets into global, event-related topics and personal topics [13]. While each effort gave us inspiration in moving from the retrospective paradigm to real-time processing, EDCoW's approach (despite heavily filtering the input data) is the most similar to LABurst with its reliance on bursts to detect event-related tokens.

Many researchers have explored platforms like Twitter and have shown interesting dynamics in user behavior around events with broad impact. For instance, Lehmann et al.'s 2012 work on collective attention on Twitter explored hashtags and the different classes of activity around their use [14]. Their work included activity surrounding unexpected, exogenous events, characterized by a peak in hashtag usage with little activity leading up to the event, which lends credence to our use of burst detection for identifying such events. Additionally, this interest in burst detection led to several domain-specific research efforts that also target sporting events specifically [2], [15], [16]. Lanagan and Smeaton's work is of particular interest because it relied almost solely on detecting bursts in Twitter's per-second message volume, which we use as inspiration for one of our baseline methods. Though naive, this frequency approach detected large bursts on Twitter in certain high-impact sporting events without complex linguist analysis and performed well in streaming contexts. Detecting such bursts provided evidence of an event, but it was difficult to gain insight into that event without additional processing. LABurst addresses this need by identifying both the overall burst and keywords related to that burst.

More recently, Xie et al.'s 2013 paper on TopicSketch performed real-time event detection from Twitter streams "without pre-defined topical keywords" by maintaining acceleration features across three levels of granularity: individual tokens, bigrams, and total stream [17]. As with Petrović's use of LSH, Xie et al. leveraged "sketches" and dimensionality reduction to facilitate event detection and also relies on language-specific similarities. Furthermore, Xie et al. focused 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 Twitter's unfiltered global stream.

Along with this extensive body of research, Twitter maintains its own offerings on "Trending Topics," which they make available to all their users. When a user visits 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 [18]. 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. Twitter's trending topics serves a different purpose than the streaming event detection described herein.

III. Moment Discovery Defined

This paper demonstrates LABurst's ability to discover and describe impactful moments from social media streams without prior knowledge of event types or domains. To that end, we first lay LABurst's foundations by defining the problem LABurst seeks to solve and presenting the model around which LABurst is built.

A. *Problem Definition*

Given an unfiltered (though potentially down-sampled) stream S of messages m consisting of various tokens w (where a "token" is defined as a space-delimited string)[1], our objective is to determine whether each time slice t contains an impactful moment and, if so, extract tokens that describe the moment.

[1] Our use of "token" is more general than a "keyword" as it includes numbers, emoticons, hashtags, or web links

Identifying and describing such moments separately is difficult because, by the time one can react to a key moment with a separate analysis, the moment may have passed. We define a "key moment" here as a brief instant in time, lasting on the order of seconds, that a journalist would label as "breaking news." Key moments might comprise the highlights of a sporting competition, the moment an earthquake strikes, the moment a terrorist attack occurs, or similar. Such moments often generate significant popular interest, affect large populations, or represent an otherwise instrumental moment in larger event (e.g., the World Cup). By focusing on these instantaneous moments of activity, we also avoid the complexities of defining "events" and the hierarchies among them.

Formally, we let E denote the set of all time slices t in which a key moment occurs. The indicator function $\mathbb{1}_E(S_t, t)$ takes the stream S up to time t and returns a 1 for all times in which an impactful moment occurs, and 0 for all other values of t. We then define the moment discovery task as approximating this indicator function $\mathbb{1}_E(S_t, t)$. We also include a function $B_E(S_t, t)$ that returns a set of words w that describe the discovered moment at time t if $t \in E$ and an empty set otherwise. To account for possible lag in reporting the event, typing out a message about the event, and the message actually posting to a social media server, we include a delay parameter τ. This parameter relaxes the task by constructing the set E' where, for all $t \in E$, $t, t+1, t+2, ..., t+\tau \in E'$. Since our evaluation compares methods that share the same ground truth, and controlling τ affects the ground truth consistently, 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 $\widehat{\mathbb{1}_{E'}}(S_t, t)$: a false positive is any time t such that $\widehat{\mathbb{1}_{E'}}(S_t, t) = 1$ and $\mathbb{1}_{E'}(S_t, t) = 0$; likewise, a false negative is any t such that $\widehat{\mathbb{1}_{E'}}(S_t, t) = 0$ and $\mathbb{1}_{E'}(S_t, t) = 1$. True positives/negatives follow as expected.

B. The LABurst Model

In LABurst, we combine the language-agnostic flexibility of burst detection techniques with the specificity of domain-specific keyword burst detectors. This integration results from ingesting a social media stream, maintaining a sliding window of frequencies for each token contained within the stream, and using the number of bursty tokens in a given minute as an indicator of the moment's impact. Critically, these tokens can be of any language and are neither stemmed, normalized, or otherwise modified. Most other approaches use language models to collapse these various token forms, whereas LABurst leverages this information as a predictor.

In more detail, LABurst runs a sliding window over the incoming data stream S and divides it into slices of a fixed number of seconds δ such that time $t_i - t_{i-1} = \delta$. LABurst then combines a set number ω of these slices into a single window (with an overlap of $\omega - 1$ slices), splits each message in that window into a set of tokens, and tabulates each token's frequency. By maintaining a list of frequency tables from the past k windows up to time t, we construct features describing a token's changes in frequency. From these features, we separate tokens into two classes: bursty tokens $\mathbf{B}_t$, and non-bursty tokens $\mathbf{B}'_t$. Following this classification, if the number of bursty tokens exceeds some threshold $|\mathbf{B}_t| \geq \rho$, LABurst flags this window at time t as containing a high-impact moment. In this manner, LABurst approximates the target indicator function with $\widehat{\mathbb{1}_{E'}}(S_t, t) = |\mathbf{B}_t| \geq \rho$ and yields $\mathbf{B}_t$ as the set of descriptive tokens for the given moment.

TABLE I: Features

Feature	Description
Frequency Regression	Given the logarithm of a token's frequency at each window, take the slope of the line that best fits this data. This feature is also duplicated for message frequency and user frequency as well.
Average Frequency Difference	The difference between the token's frequency in the most recent window and the average frequency across the previous $k - 1$ windows. As with the regression feature, this feature was also calculated for message frequency and user frequency.
Inter-Arrival Time	The average number of seconds between token occurrences in the previous k windows.
Entropy	The entropy of the set of messages containing a given token.
Density	The density of the @-mention network of users who use a given token.
TF-IDF	The term frequency, inverse document frequency for a each token.
TF-PDF	A modified version of TF-IDF called term frequency, proportional document frequency [20].
BursT	Weighted combination of token's actual and expected frequencies [21].

To avoid spurious bursts, retweets are discarded since existing literature shows retweets propagate extremely rapidly, leading to possible false bursts [19].

1) Temporal Features: To capture token burst dynamics, we constructed a set of temporal and graphical features to model these effects, shown in Table I. These features were calculated per token and normalized into the range $[0, 1]$ to avoid scaling issues. Each feature's relative importance was then examined through an ablation study described later.

2) LABurst's Bursty Token Classification: LABurst's primary capability differentiating between bursty and non-bursty tokens. To make this determination, LABurst integrates these temporal features into feature vectors for each token and processes them using an ensemble of known classification algorithms. Specifically, we use ensembles of support vector machines (SVMs) [22] and random forests (RFs) [23] integrated using AdaBoost [24].

Training these burst detection classifiers requires both positive and negative samples of bursty tokens. While obtaining positive samples of bursty tokens is relatively straightforward, negative samples are problematic. For positive samples, we can identify high-impact, real-world events and construct a set of seed tokens that *should* experience bursts along with the event (as done in typical seed-based event detection approaches). Negative samples, however, are difficult to identify since one cannot know all events occurring around the world at a given moment. To address this difficulty, we rely on a trick of linguistics and use stop words as negative samples, our justification being that stop words are in general highly used but used consistently (i.e., stop words are intrinsically non-bursty). Therefore, in our experiments, we train LABurst on a set of events with known bursty tokens and stop words in both English and Spanish. As this task is semi-supervised, we also include a self-training phase to expand our list of bursty tokens.

IV. Evaluation Framework

Having established the details of our model, we now turn to frameworks for evaluating LABurst compared to existing methods. We first look to similar methods for detecting interesting events from social media streams and compare their performance. We then include a second experiment to demonstrate LABurst's domain independence and utility in the disaster response context.

A. Accuracy in Event Discovery

Our first research question is **RQ1**: is LABurst able to identify key moments as well as existing systems? To answer this question, we constructed an experiment for enumerating key moments during major sporting competitions. Such competitions are interesting given their large followings (many fans to post on social media), thorough coverage by sports journalists (high-quality ground truth), and regular occurrence (large volume of data), making them ideal for both data collection and evaluation. Such events are also complex in that they include multiple types of events and unpredictable patterns of events around scores, fouls, and other compelling moments of play.

Our first step here was to collect data from a number of popular sporting events and identify key moments in each competition. We captured these moments and their times from sports journalism articles, game highlights, box scores, blog posts, and social media messages. These moments then comprise our ground truth.

We then introduced a pair of baseline methods: first, a time-series algorithm using raw message frequency following the approaches of Vasudevan et al. and the "activity peak detection" method set forth by Lehmann et al. [2], [14], and second, a seed keyword-based algorithm in the pattern of Cipriani and Zhao et al. [1], [15]. We then evaluate the relative performance for LABurst and both baselines as described below.

1) Sporting Competitions: To minimize bias, these competitions covered several different sporting types, from horse racing to the National Football League (NFL), to Fédération Internationale de Football Association (FIFA) premier league soccer, to the National Hockey League (NHL), National Basketball Assoc. (NBA), and Major League Baseball (MLB). Each competition also contained four basic types of events: beginning of the competition, its end, scores, and penalties. Table II lists the events we identified and the number of key moments in each.

In 2012, we tracked four Premier League games in November. For the 2013 World Series between the Boston Red Sox and the St. Louis Cardinals, we covered the final two games on 28 October and 30 October of 2013. Likewise, we tracked a subset of playoff games during the 2014 NHL Stanley Cup and NBA playoffs. For the 2014 World Cup, our analysis included a number of early matches during stages 1 and 2 and the 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.

TABLE II: Sporting Competition Data

Sport	Key Moments
Training Data	
2010 NFL Division Championship	13
2012 Premier League Soccer Games	21
2014 NHL Stanley Cup Playoffs	24
2014 NBA Playoffs	3
2014 Kentucky Derby Horse Race	3
2014 Belmont Stakes Horse Race	3
2014 FIFA World Cup Stages A+B	80
Testing Data	
2013 MLB World Series Game 5	7
2013 MLB World Series Game 6	8
2014 NFL Super Bowl	13
2014 FIFA World Cup Third Place	11
2014 FIFA World Cup Final	7
Total	193

These events were split into training and testing sets; training data covered the 2010 NFL championship, 2012 premier league soccer games, NHL/NBA playoffs, the Kentucky Derby/Belmont Stakes horse races, and several days of World Cup matches in June of 2014. The testing data covered the 2013 MLB World Series, 2014 NFL Super Bowl, and the final two matches of the 2014 FIFA World Cup.

2) Burst Detection Baselines: LABurst straddles the line between time-series analysis and token-centric burst detection. Therefore, to evaluate LABurst properly, we implemented two baselines for comparison. The first baseline, to which we refer as RawBurst, uses a known method for detecting bursts by taking the difference between the number of messages seen in the current time slice and the average number of messages seen over the past k time slices [2], [14].

Formally, we define a series of time slices $t \in T$ segmented into δ seconds and a social media stream S containing messages m such that S_t contains all messages in the stream between $t-1$ and t. We then define the frequency of a given time slice t as $\text{freq}(t,S) = |S_t|$ and the average over the past k time slices as $\text{avg}(k,t,S)$, shown in Eq. 1.

$$\text{avg}(k,t,S) = \frac{\sum_{j=t-k}^{t} \text{freq}(j,S)}{k} \tag{1}$$

Given these functions, we take the difference $\Delta_{t,k}$ between the frequency at time t and the average over the past k slices such that $\Delta_{t,k} = \text{freq}(t,S) - \text{avg}(k,t,S)$. If this difference exceeds some threshold ρ such that $\Delta_{t,k} \geq \rho$, we say an event was detected at time t.

Following Cipriani from Twitter's Developer Blog and others, we then modify the RawBurst algorithm to detect events using frequencies of a small set of seed tokens $w \in W$, to which we will refer as TokenBurst [1]. To convert RawBurst into TokenBurst, we modify the $\text{freq}(t,S)$ function to return the summed frequency of all seed tokens, as shown in Eq. 2 where $\text{count}(w,S_t)$ returns the frequency of token w in the stream S during time slice t. These seed tokens are chosen such that they likely exhibit bursts in usage during the key moments of our sporting event data, such as "goal" for goals in soccer/football or hockey or "run" for runs scored in baseball. This TokenBurst implementation also includes some rudimentary normalization to collapse modified words to their originals (e.g., “gooaallll” down to “goal”). Many existing stream-based event detection systems use just such an approach to track specific types of events.

$$\text{freq}(t, S) = \sum_{w \in W} \text{count}(w, S_t) \tag{2}$$

Since our analysis covers three separate types of sporting competitions, seed keywords should include tokens from vocabularies 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 tokens for which we searched are shown in Table III. We also used regular expressions to collapse deliberately misspelled tokens to their normal counterparts.

TABLE III: 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) Performance Evaluation: Having defined LABurst, RawBurst, and TokenBurst, we evaluate these algorithms by constructing a series of receiver operating characteristic (ROC) curves across test sets of our sports data. We then evaluate relative performance between the approaches by comparing their respective areas under the curves (AUCs) by varying the threshold parameters for each method. In RawBurst and TokenBurst, this threshold parameter refers to ρ in $\Delta_{t,k} \geq \rho$. For our LABurst method, the ROC curve is generated by varying the minimum ρ in $\widehat{\mathbb{1}_{E'}}(S_t, t) = |\mathbf{B}_t| \geq \rho$. The ROC AUC is useful because it is robust against imbalanced classes, which we expect to see in such an event detection task. By comparing AUC, we can answer to **RQ1**.

B. Evaluating Domain Independence

Beyond LABurst's ability to discover and describe interesting moments, we also claim it to be domain independent. To justify this claim, we must answer our second research question **RQ2**: can LABurst transfer models learned in one context to another one separate from its training domain and remain competitive?

Detecting key moments within sporting competitions is useful for advertising or automated highlight generation, but a more compelling task is to detect higher-impact events like natural disasters. The typical seed-token-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 and lead to false positives. LABurst could be beneficial here as one need not know details like event location, language, or type. This context presents an opportunity to evaluate LABurst in a new domain and compare it to existing work by Sakaki, Okazaki, and Matsuo [6]. Thus, to answer **RQ2**, we can take the LABurst model as trained on sporting events presented for **RQ1** and apply them directly to this context.

For this earthquake detection task, we compare LABurst with the TokenBurst baseline using the keyword “earthquake,” as in Sakaki, Okazaki, and Matsuo. We target earthquakes in Japan over the past two years and select two of the most severe: 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. 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. If the lag between TokenBurst and LABurst is sufficiently small, we will have good evidence for an affirmative answer to **RQ2**.

V. Data Collection

While the algorithms described herein are general and can be applied to any sufficiently active social media stream, the ease with which one can access and collect Twitter data makes it an attractive target for our research. To this end, we leveraged two existing Twitter corpora and created our own corpus of tweets from Twitter's 1% public sample stream[2]. This new corpus was created using the twitter-tools library[3] developed for evaluations at the NIST Text Retrieval Conferences (TRECs). In collecting from Twitter's public sample stream, we connect to the Twitter API endpoint (provide **no filters**), and retrieve a sampling of 1% of all public tweets, which yields approximately 4,000 tweets per minute.

The two existing corpora we used were the Edinburgh Corpus [25], which covered the 2010 NFL division championship game, and an existing set of tweets pulled from Twitter's firehose source targeted at Argentina during November of 2012, which covered the four Premier League soccer games. All remaining data sets were extracted from Twitter's sample stream over the course of October 2013 to July 2014.

Where possible, for each event (both sporting and earthquake), we recorded all tweets from the 1% stream starting an hour before the target event and ending an hour after the event, yielding over 15 million tweets. Table IV shows the breakdown of tweets collected per event. From these tweets, we extracted $1,109$ positive (i.e., known bursty) samples and $43,037$ negative samples for a total of $44,146$ data points.

VI. Experimental Results

A. Setting Model Parameters

Prior to carrying out the experiments described above, we needed appropriate parameters for window sizes and LABurst's classifiers. For LABurst's slice size δ, window size ω, and k previous window parameters, preliminary experimentation yielded acceptable results with the following: $\delta = 60$ seconds, $\omega = 180$ seconds, and $k = 10$. We used these δ and k parameters in both RawBurst and TokenBurst as well.

Regarding LABurst's classifier implementations, we used the Scikit-learn[4] Python package for SVMs and RFs as well as an implementation of the ensemble classifier AdaBoost, each of which provided a number of hyperparameters. For

[2] https://dev.twitter.com/streaming/reference/get/statuses/sample

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

[4] http://scikit-learn.org/

TABLE IV: Per-Event Tweet Counts

Event	Tweet Count
Training Data	
2010 NFL Division Championship	109,809
2012 Premier League Soccer Games	1,064,040
2014 NHL Stanley Cup Playoffs	2,421,065
2014 NBA Playoffs	500,170
2014 Kentucky Derby Horse Race	233,172
2014 Belmont Stakes Horse Race	226,160
2014 FIFA World Cup Stages A+B	5,867,783
Testing Data	
2013 MLB World Series Game 5	1,052,852
2013 MLB World Series Game 6	1,026,848
2013 Honshu Earthquake	444,018
2014 NFL Super Bowl	1,024,367
2014 FIFA World Cup Third Place	809,426
2014 FIFA World Cup Final	1,166,767
2014 Iwaki Earthquake	358,966
Total	16,305,443

SVMs, the primary hyperparameter is the type of kernel to use, and initial experiments showed SVMs with linear kernels performed poorly. We then applied principal component analysis to reduce the training data's dimensionality to a three-dimensional space. The resulting visualization showed a decision boundary more consistent with a sphere rather than a clear linear plane, motivating our choice of the radial basis kernel (RBF).

For the remaining hyperparameters, we constructed separate parameter grids for SVMs and RFs and performed a distributed grid search. The grid for SVM's two parameters, cost c and kernel coefficient γ, covered powers of two such that $c, \gamma = 2^x$, $x \in [-2, 10]$. RF parameters were similar for the number of estimators n and feature count c' such that $n = 2^x$, $x \in [0, 10]$ and $c' = 2^y$, $y \in [1, 12]$.

Each parameter set was scored using the AUC metric across a randomly split 10-fold cross-validation set, with the best scores determining the parameters used in our ensemble. We then combined the two classifiers using Scikit-learn's AdaBoost implementation, yielding the results shown in Table V. These grid search results show RFs perform better than SVMs, and the AdaBoost ensemble outperforms each individual classifier.

TABLE V: Per-Classifier Hyperparameter Scores

Classifier	Params	ROC-AUC
SVM	kernel = RBF, $c = 64$, $\gamma = 0.0625$	87.48%

RF	trees = 1024,
	features = 2
		88.35%

AdaBoost	estimators = 2	89.84%

B. Ablation Study

Given the various features from both our own development and related works, we address the relative values or importance of each feature to our task. To answer this question, we performed an ablation study with a series of classifiers, each excluding a single feature set. Each degenerate classifier was then compared with the full AdaBoost classifier using the same 10-fold cross-validation strategy as above. Table VI shows each model's AUC and its difference with that of the full model. These results suggest the regression and entropy features contribute the most while the average difference features seem to hinder performance.

TABLE VI: Ablation Study Results

Feature Sets	ROC-AUC	Difference
AdaBoost, All Features	89.84%	–
Without Regression	87.79%	-2.05
Without Entropy	87.94%	-1.90
Without TF-IDF	88.85%	-0.99
Without TF-PDF	89.00%	-0.84
Without Density	89.07%	-0.77
Without InterArrival	89.46%	-0.38
Without BursT	89.52%	-0.31
Without Average Difference	90.56%	0.72

C. Event Discovery Results

Our first research question (**RQ1**) is whether LABurst performs as well as existing methods in detecting key moments. Prior to presenting comprehensive results, we first examine performance for each sporting competition for RawBurst, TokenBurst, and LABurst. We also include a restricted version of LABurst, called LABurst*, that is trained using the best features from the ablation study (i.e., all features but average difference).

For the 2013 World Series, RawBurst's AUC is 0.62, TokenBurst's is 0.76, LABurst achieves 0.73, and LABurst* yields 0.76. The two LABurst models clearly dominate RawBurst and exhibit performance on par with TokenBurst. During the Super Bowl, RawBurst and TokenBurst achieve an AUC of 0.68 and 0.78 respectively, while LABurst and LABurst* perform worse with an AUC of 0.63 and 0.64. During the 2014 World Cup, both LABurst and LABurst* (AUC = 0.72 and 0.73) outperformed both RawBurst (AUC = 0.66) and TokenBurst (AUC = 0.64).

D. Composite Results

Figure 1 shows comprehensive performance, which shows ROC curves for all three methods across all three testing events. The blue and green lines showing the ROC curves for RawBurst and TokenBurst respectively. The red line shows the ROC curve for the LABurst model trained using all features, and the black line shows LABurst*. From this figure, we see LABurst (AUC=0.7) and LABurst* (AUC=0.71) both outperform RawBurst (AUC=0.65) and perform nearly as well as TokenBurst (AUC=0.72). Given these results, one can answer **RQ1** in that, yes, LABurst is competitive with existing methods.

More interestingly, assuming equal cost for false positives and negatives and optimizing for the largest difference between true positive rate (TPR) and false positive rate (FPR), TokenBurst shows a TPR of 0.56 and FPR of 0.14 with a difference of 0.42 at a threshold value of 13.2. LABurst, on the other hand, has a TPR of 0.64 and FPR of 0.28 with a difference of 0.36 at a threshold value of 2. From these values, we see LABurst achieves a higher true positive rate at the cost of a higher false positive rate. This effect is possibly explained

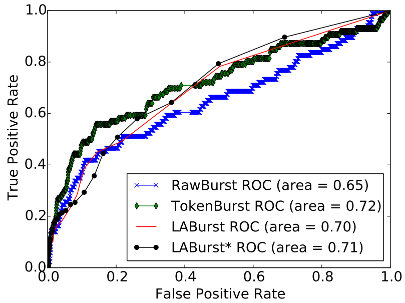

Fig. 1: Composite ROC Curves

TABLE VII: 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	goaaaaalllllllll, goalllll, godammit, goetze, gollllll, goooooool, gotze, gotzeeee, götze, nooo, yessss, ドイツ

by the domain-specific nature of our test set and TokenBurst implementation, as discussed in more detail in Section VII-C.

E. Earthquake Detection

Our final research question (**RQ2**) determines if adapting LABurst's models, as trained on using sporting events listed in Tables II and IV, can compete with existing techniques in a different domain. We explored this adaptation by applying the sports-trained LABurst classifier to Twitter data surrounding known earthquake events in Japan in 2013 and 2014. As described, we compare LABurst with frequency of the token "earthquake." Comparing the two algorithms, we see zero lag between LABurst and TokenBurst's detection, both experiencing bursts simultaneously. Given this result, LABurst seems effective in cross-domain event discovery (**RQ2**).

One can now ask what tokens we identified as bursting when the earthquakes occurred. Many of the tokens are in Japanese, and we 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[5] 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.

VII. Analysis

In comparing LABurst with the baseline techniques, it is important to note the strengths and weaknesses of each baseline: RawBurst requires no prior information but provides little in the way of semantic information regarding detected events, while TokenBurst provides this semantic information at the cost of missing unknown tokens or significant events that do not conform to its prior knowledge. LABurst attempts to combine these two approaches by supporting undirected event discovery while yielding insight into these moments by tagging relevant bursting tokens.

A. Identifying Event-Related Tokens

As mentioned, where the baselines sacrifice either insight or flexibility, LABurst jointly attacks these problems and yields event-related tokens automatically. These tokens may include misspellings, colloquialisms, and language-crossing tokens, which makes them hard to know a priori. The 2014 World Cup provides an illustrative case for such unexpected tokens given its enormous viewership: many Twitter users of many different languages are likely tweeting about the same event. Table VII shows a selection of events from the final two World Cup 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).

Several interesting artifacts emerge from this table, first of which is that one can get an immediate sense of what happened in the detected moment 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 VII; 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, references to diving and specific player/team 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" for the Netherlands' player, Van Persie, and likewise for Germany's Götze in the third event (and the accompanying variations of his name). These tokens would be difficult to capture beforehand as TokenBurst would require, and such tokens 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 distinct, typical normalization schemes could not capture these connections. Instead, capturing these words in the baseline would require a pre-specified keyword list in all possible languages or a machine translation system capable of normalizing within different languages.

B. Discovering Unanticipated Moments

Results show LABurst is competitive with the domain-specific TokenBurst, but TokenBurst's specificity makes it unable to detect unanticipated moments, and we can see instances of such omissions in the last game of World Cup. Figure 2 shows target token frequencies for TokenBurst in

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

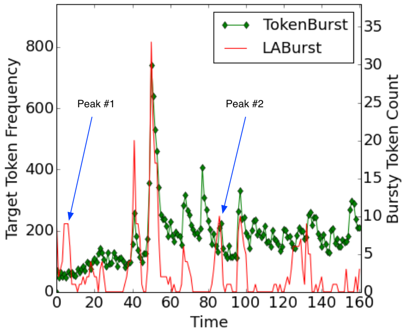

Fig. 2: Baseline and LABurst Frequencies

green and LABurst's volume of bursty tokens in red. From this graph, we can see the first instance in Peak #1 where LABurst exhibits a peak missed by TokenBurst. Tokens appearing in this peak include "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. While not necessarily a sports-related event, many viewers were interested in the trophy reveal, making it a key moment. At peak #2, slightly more than eighty minutes into the data (which is sixty minutes into the match), LABurst sees another peak otherwise inconspicuous in the TokenBurst curve. Upon further exploration, tokens present in this peak refer to Argentina's substituting Agüero for Lavezzi at the beginning of the match's second half.

C. Addressing the Super Bowl

While LABurst performs as well as the domain-specific TokenBurst algorithm in both the World Series and World Cup events, one cannot ignore its poor performance during the Super Bowl. Since LABurst is both language agnostic and domain independent, it likely detects additional high-impact events outside of the game start/end, score, and penalty events present in our ground truth. For instance, during the Super Bowl, spectators tweet about moments beyond sports plays: they tweet about the half-time show, commercials, and massive power outages. Since our ground truth disregards such moments, LABurst's higher false-positive rate is less surprising, and TokenBurst's superior performance might result from its specificity in domain knowledge with respect to the ground truth (i.e., both include only sports data). Hence, LABurst's ability to detect unanticipated moments potentially penalizes it in domain-specific tasks.

LABurst's propensity towards more organic moments of interest becomes obvious when we inspect the tokens LABurst identified when it detected a large burst early on that TokenBurst missed. Approximately four minutes before the game started (and therefore before when TokenBurst would detect any event), LABurst saw a large burst with tokens like "joe", "namath", "fur", "coat", "pimp", "jacket", "coin", and "toss". As it turns out, Joe Namath, an ex football star, garnered significant attention from fans when he tossed the coin to decide which team would get first possession. Since neither our ground truth data nor TokenBurst's domain knowledge captured this moment, LABurst's detection is counted as a false positive much like the trophy presentation during the World Cup.

VIII. Limitations and Extensions

The approach adopted herein is fundamentally limited regarding tracking potentially interesting events that do not garner mass awareness on social media. Since the LABurst presupposes significant bursts in activity during key moments, if only a few people are participating in or following an event, LABurst will likely be unable to detect moments in that event. This effect is clear in applying LABurst to regular season baseball games: since Major League Baseball sees over 2,400 games in a season, experiments showed too few viewers were posting messages to Twitter during these games to generate any significant burst. As a result, many key moments in these games are exceedingly difficult to capture via burst detection.

This deficiency leads to a potential opportunity, however, in combining domain knowledge with LABurst's domain-agnostic foundations. For example, one could apply domain-specific filters to the Twitter stream prior to LABurst in the detection pipeline. Since LABurst uses relative frequencies to identify bursts, this pre-filtering step should amplify the signal of potentially bursty tokens in the stream and increase LABurst's likelihood of detecting them. Returning to the baseball example, one could use domain information to filter the Twitter stream to contain only relevant tweets, and the baseball-specific key moments should become more apparent.

In a more interesting case, this domain knowledge could be applied as events are discovered and allow LABurst to provide more insight into those events as they unfold. Examples where such an approach could be used include hurricanes, where one can know the name of the hurricane and its approximate area of landfall, filter the Twitter stream accordingly, and then use LABurst to track the unanticipated moments that occur once the storm hits. One could apply a similar approach in the early hours of political protests or mass unrest to track events that may not be covered by mainstream news outlets (e.g., in oppressive regimes where media is controlled). Additional knowledge such as geolocation data could also be integrated into these stream filters to increase LABurst's moment discovery capabilities further.

IX. Conclusions

Revisiting motivations, this research sought to demonstrate whether LABurst, a streaming, language-agnostic, burst-centric algorithm, can discover key moments from unfiltered social media streams (specifically Twitter's public sample stream). Our results show temporal features can identify bursty tokens and, using the volume of these tokens as an indicator, we can discover key moments across a collection of disparate sporting competitions. This approach's performance is competitive with existing baselines. Furthermore, these sports-trained models are adaptable to other domains with a level of performance exceeding a simple time series baseline and rivaling a domain-specific method. LABurst's performance relative to the domain-specific baseline shows this method's potential given its omission of manual keyword selection and prior knowledge.

Beyond this comparison, our approach also offers notable flexibility in identifying bursting tokens across language boundaries and in supporting event description; that is, we can get a sense of the occurring event by inspecting bursty tokens returned by LABurst. These features combine to form a capable tool for discovering unanticipated moments of high interest, regardless of language. This technique is particularly useful for journalists and first responders, who have a vested interest in rapidly identifying and understanding high-impact moments, even if a journalist or aid worker is not physically present to observe the event. Possibilities also exist to combine LABurst with other domain-specific solutions and yield insight into unanticipated events, events missed by existing approaches, or events that might otherwise be lost in the noise.

Acknowledgment

This work was supported in part by the National Science Foundation under IIS-1218043 and CNS-1405688. Any opinions, findings, conclusions, or recommendations expressed are those of the authors and do not necessarily reflect the views of the sponsors. This work also 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.

References

[1] L. Cipriani, "Goal! Detecting the most important World Cup moments," Twitter, Tech. Rep., 2014. [Online]. Available: https://blog.twitter.com/2014/goal-detecting-the-most-important-world-cup-moments

[2] 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*. IEEE, 2013, pp. 181–186.

[3] 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*. ACM, 1998, pp. 37–45.

[4] J. Kleinberg, "Bursty and hierarchical structure in streams," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 91–101. [Online]. Available: http://doi.acm.org/10.1145/775047.775061

[5] 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*, ser. VLDB '05. VLDB Endowment, 2005, pp. 181–192. [Online]. Available: http://dl.acm.org/citation.cfm?id=1083592.1083616

[6] 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*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 851–860. [Online]. Available: http://doi.acm.org/10.1145/1772690.1772777

[7] 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*, ser. KDD '10. New York, NY, USA: ACM, 2010, pp. 929–938. [Online]. Available: http://doi.acm.org/10.1145/1835804.1835922

[8] 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*, ser. HLT '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 181–189. [Online]. Available: http://dl.acm.org/citation.cfm?id=1857999.1858020

[9] 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.

[10] 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*, vol. 2011, 2013.

[11] J. Weng and B.-S. Lee, "Event detection in Twitter," in *ICWSM*, 2011. [Online]. Available: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2767

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

[13] 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*. Association for Computational Linguistics, 2012, pp. 536–544.

[14] J. Lehmann, B. Gonçalves, J. J. Ramasco, and C. Cattuto, "Dynamical classes of collective attention in Twitter," in *Proceedings of the 21st International Conference on World Wide Web*, ser. WWW '12. New York, NY, USA: ACM, 2012, pp. 251–260. [Online]. Available: http://doi.acm.org/10.1145/2187836.2187871

[15] 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*, vol. abs/1106.4, 2011. [Online]. Available: http://arxiv.org/abs/1106.4300

[16] J. Lanagan and A. F. Smeaton, "Using Twitter to detect and tag important events in live sports," *Artificial Intelligence*, pp. 542–545, 2011.

[17] 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*. IEEE, 2013, pp. 837–846.

[18] L. Sydell, "How Twitter's trending algorithm picks its topics," Dec. 2011.

[19] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 591–600.

[20] 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*, ser. WISE '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 73–82. [Online]. Available: http://dl.acm.org/citation.cfm?id=645962.674082

[21] 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*, ser. ISNN'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 548–557. [Online]. Available: http://dl.acm.org/citation.cfm?id=2009463.2009531

[22] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. [Online]. Available: http://dx.doi.org/10.1007/BF00994018

[23] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[24] Y. Freund and R. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," *Computational learning theory*, vol. 55, no. 1, pp. 119–139, 1995. [Online]. Available: http://link.springer.com/chapter/10.1007/3-540-59119-2_166

[25] 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*, ser. WSA '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 25–26. [Online]. Available: http://dl.acm.org/citation.cfm?id=1860667.1860680