Change-point methods in disturbance identification and probabilistic labeling of events

Emmanuel Yashchin[1], Nianjun Zhou[1], Anuradha Bhamidipaty[1]

[1]IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598

Abstract

Environmental disturbances often cause failure or malfunction of assets and related outage events. However, it is quite common that the failures cannot be identified as being caused by a disturbance based on the data due to the limited information available at the time of data compilation, time constraints, or personnel's insufficient training. The ability to label an outage event reliably as one caused by a disturbance is a key prerequisite for analytic activities such as risk modeling, outage detection, prediction, and management. Change-point methods play an important role in this process, enabling efficient identification of disturbances and establishment of temporal boundaries. We introduce a methodology for disturbance identification and illustrate its use in conjunction with complex processes governing weather-related outages, including handling spatio-temporal effects and outliers. We also discuss the use of this methodology for probabilistic labeling of outage service tickets.

Key Words: Control Charts, Detection, Disturbance, Machine Learning, Monitoring, Segmentation, Run Length.

1. Introduction

This work is motivated by the problem of probabilistic labeling of the data records. Many Machine Learning applications depend critically on the availability of a labeled data set, e.g., see Goodfellow et al. (2016). With such data set, one can address several problems, including modeling, classification, regression, and prediction. In many practical situations, however, a data set including properly labeled records is not attainable. Thus, construction of the typical modeling framework, including training, validation, and test data sets, becomes impossible, preventing the use of well-developed machine learning and statistical analysis tools. Difficulties in obtaining a labeled subset of records can have several reasons, one of the main ones being the cost of labeling. For many important machine learning applications, this cost is prohibitively high. Sometimes, obtaining a definitive label is, in principle, impossible: for example, in records describing certain failures of computing systems, it might not be feasible to classify them definitively as hardware-related or software-related because of a complex interaction between the respective factors.

When cost is the major factor in the record labeling effort, automation can be of help. For example, documents can often be classified using Natural Language Processing (NLP) techniques, see Minaee et al. (2021). However, in cases where the main objective is a construction of the training – validation - testing data set, automated search for relevant records and their labeling can sometimes produce data sets that are inherently biased. In some cases, this bias can be rectified using a modeling technique that accurately reflects the properties of the data collection process, see Yashchin (2007).

In many cases, however, it is beneficial to use probabilistic (instead of deterministic) labels for the data records. Such a type of labeling reflects better the inherent uncertainty present in the labeling process, and it enables one to construct helpful data sets for modeling and inference, see Zhou et al. (2021).

The process of probabilistic record labeling can be quite complex and dependent on the specific nature of the data. In this paper, we consider a particular setting where change-point theory plays a key role in this process. Consider the situation where we observe the process of events (e.g., requests for power service restoration) that is influenced by occasional disturbances (e.g., storm affecting the service area). Given a particular outage event, we would like to classify (label) it as (a) *related* to a disturbance or (b) *not related* to a disturbance. As circumstances causing the event could materialize under both scenarios, the task of classifying the event can be quite challenging. To fulfil this task, it is essential to establish the boundaries of the disturbance period - and this is where change-point techniques are of value. Limiting the occurrence of disturbance-related events to well-defined disturbance periods greatly simplifies the classification problem, since it helps one evaluate and validate the *baseline* rate of events that are not disturbance-related. Detection of disturbance periods is then reduced to the analysis of change-points of the observed overall rate of outage events relative to the stochastic baseline process.

Several procedures of this type are described in the literature. For example, *scan statistics* can be quite helpful, see Glaz and Koutras (2019), Kulldorf et al. (2007). Another approach involves signal segmentation, see Lu et al. (2010), Cho and Fryzlewicz (2015). The non-restarting Cusum was introduced in Gandy and Lau (2013); in this approach, the *Cusum* procedure is supplemented by an upper reflecting boundary that regulates the "return to normal" process. Extensions of this technique were discussed in Dassanayake and French (2016), Hall (2019), and Hall and French (2019). In conjunction with the continuous inspection sampling schemes, procedures involving a restarting mechanism with a reflective upper threshold were discussed in Beattie (1962) and Wasserman and Wadsworth (1989). Niu et al. (2016) presented a selected overview of multiple change-point detection methods, and Truong et al. (2020) give a selected review of offline detection of change-points.

In this paper, we introduce new procedures for detecting disturbances and establishing their boundaries. They offer several advantages in terms of operational convenience, analytic properties, and visualization. The procedures use *Cusum* as the primary mechanism of information handling. However, the re-starting mechanism built into them enables considerable flexibility in terms of the declaration of "return to normal" conditions and finalizing the temporal endpoints of the disturbance. The main advantage of our method compared with the scan statistics approach is that our procedures are not associated with pre-specified scan windows. We also use the fact that our objectives are less stringent than the complete process segmentation: we strive to isolate the disturbance windows and are not assuming that the process mean is stable, either within or outside these windows.

In Section 2, we present the basic approach to the problem of disturbance identification. In Section 3, we discuss the use of the change-point methodology for identifying disturbances and estimating their boundaries. In Section 4, we discuss an application related to the management of outage tickets and their probabilistic labeling. In Sec. 5, we give several concluding remarks.

2. The Basic Approach

Let us assume, for simplicity, that the observed events belong to one of two classes: those not caused by a disturbance (the baseline class 1) and those caused by a disturbance (class 2). We will assume that the stochastic processes generating the events are of *Cox* type with the intensity functions $\Lambda_1(t)$ and $\Lambda_2(t)$, respectively. The intensity functions are themselves random processes, and their realizations corresponding to the data set at hand are denoted by $\lambda_1(t)$ and $\lambda_2(t)$. In practical situations, the intensities can also depend on random covariates (e.g., corresponding to seasonality or spatial factors associated with the events) – however, we will not explicitly include them in the notation. In our application, we observe the events corresponding to the superposition of the underlying intensities, $\Lambda(t) = \Lambda_1(t) + \Lambda_2(t)$; the corresponding realization is $\lambda(t) = \lambda_1(t) + \lambda_2(t)$.

Our objective is to label the outage events as belonging to one of the two classes. It is known that for the *Cox* processes, the probability of the event that arrived at time t to belong to the first class, is

$$\pi_1(t) = \frac{\lambda_1(t)}{\lambda_1(t)+\lambda_2(t)} = \frac{\lambda_1(t)}{\lambda(t)}, \tag{2.1}$$

see Ross 2014, Zhou et al. (2021). One of the challenges in obtaining the probabilistic labels is obtaining the estimated intensity realizations $\{\hat{\lambda}_1(t), \hat{\lambda}_2(t)\}$. Since the intensity of the superposition of two processes is estimable, we can obtain $\hat{\lambda}(t)$ using methods from the theory of point processes, see Snyder and Miller (2012). Obtaining the estimate of the baseline, $\hat{\lambda}_1(t)$, is a much more complex problem. We are, however, making a simplifying assumption that the events of the second category occur only during disturbance periods. We also assume that these periods do not dominate the data set. Under these assumptions, estimation of the baseline becomes feasible, as an application of the *robust estimation* methods enables us to represent the time periods (e.g., days) with high disturbance-related counts as outliers and neutralize their effect through imputation techniques. The robust methods for baseline derivation are described in Zhou et al. (2021). For the purposes of this paper, we will assume that the estimate $\hat{\lambda}_1(t)$ is available, and thus the main problem in establishing the probabilistic labels $\hat{\pi}_1(t), \hat{\pi}_2(t)$ reduces to isolating the disturbance periods, estimating $\hat{\lambda}(t)$ within these periods and applying (2.1). Furthermore, we will assume that the accompanying scaling process $\hat{\sigma}_1(t)$ is also available; this process characterizes the commonly observed variability of the process of events.

In what follows, we will also assume that the time t is discretized, and so we will use instead the index i of the corresponding bin. For simplicity, we will refer to these bins as "days" (in line with the application to the electric utility service tickets considered below); thus, the event rates $\lambda_1(t), \lambda_2(t)$ correspond to daily counts.

The overall strategy for obtaining the probabilistic labels can be summarized as follows:

1. Estimate the baseline process characteristics using robust methods;
2. Use imputation to ensure that baseline covers the complete time range;
3. Specify and parametrize a measure of deviation between the process characteristics and the baseline;
4. Establish acceptable/unacceptable levels for the parameters specified in Step 2;
5. Define and set performance characteristics (false alarm rate, sensitivity) for the control scheme responsible for detecting disturbances;
6. Apply control scheme and identify disturbances, and corresponding endpoints;
7. Obtain Probabilistic Labels;
8. Validate the methodology against any partial labeling, if available; validate relevance against other objectives (e.g., prediction, classification).

As noted above, the focus of our paper is on steps 3 – 8, with a special emphasis on steps 5 – 6.

3. Change-point methodology for disturbance identification

In this section, we introduce the methods for the detection of disturbances in the stream of events and establishing the start and end points of every disturbance. Since the baseline and the accompanying scaling process $\{\hat{\lambda}_1(t), \hat{\sigma}_1(t)\}$ are available, one natural way is to characterize the intensity of events in the vicinity of time t in terms of a standardized score. The parameter of the measure of the deviation between the actual event process characteristics and the baseline (see point #3 of the strategy presented at the end of Sec. 2) is then the mean of the standardized score. In line with point #4, we could set the acceptable and unacceptable levels of this mean score, which will enable us to regulate the operating characteristics of the disturbance detection scheme.

3.1 Standardization

In setting up the control scheme for detecting disturbances, it is essential to specify the acceptable and unacceptable deviations between the intensities of the actual process of events and the baseline intensity. In terms of the discretized time, this amounts to specifying the acceptable and unacceptable deviations between the actual daily counts of events (we will denote them by $\{X_i\}$) and the baseline daily counts, $\{\hat{\lambda}_{1,i}\}$, for i = 1, 2, …. In general, when establishing the acceptable/unacceptable levels, one needs to consider several factors, including business requirements, scheme performance under various scenarios, and the process history, e.g., see Hawkins and Olwell (2012), Yashchin et al. (2021). In this paper, we utilize the scale process $\{\hat{\sigma}_{1,i}\}$ provided jointly with the baseline estimate. We transform the *count* process into the *score* process:

$$Y_i = \left\{\frac{X_i - \hat{\lambda}_{1,i}}{\hat{\sigma}_{1,i}}\right\}, \quad i = 1, 2, \ldots. \tag{3.1}$$

Note that one possible version of the scaling process is $\hat{\sigma}_{1,i} = \sqrt{\hat{\lambda}_{1,i}}$. This is in line with the Poisson distribution properties – however, one will need to take into consideration that the process of daily rates $\{X_i\}$ could deviate from the Poisson assumption, and it is desirable for the disturbance-detecting procedures to show reasonable performance under these conditions. One would need to be prepared for the possibility that the marginal distribution of the counts deviates from the Poisson model (e.g., due to over-dispersion). Furthermore, some autocorrelation in the processes $\{X_i\}$, $\{\hat{\lambda}_{1,i}\}$, $\{\hat{\sigma}_{1,i}\}$ and cross-correlation between them could also need to be addressed.

Of course, it would be desirable for the process of scores $\{Y_i\}$ to conform to an independent and identically distributed (*iid*) standard Gaussian pattern, as this would enable deployment of the standard detection procedures developed for this pattern. However, this assumption does not hold in practical situations involving low levels of the underlying baseline rates, $\{\lambda_{1,i}\}$. Given the positive skew that is typically present in scores when the daily rates are low, it is crucial to ensure that the false alarm rate of the score-based disturbance detection scheme is acceptably low for all processes of daily rates that are compatible with the baseline process.

3.2 Disturbance detection

In this section, we will assume that the acceptable and unacceptable levels of the mean scores $\{Y_i\}$ are both fixed (i.e., they do not depend on the underlying baseline rate); denote them by μ_a and μ_u, respectively ($\mu_a < \mu_u$). For disturbance detection, we transform the scores to the *Cusum* process $\{S_i\}$:

$$S_0 = s_0,\ S_i = \max[S_{i-1} + Y_i - k, 0]\ ,\ I = 1,2,\ldots \tag{3.2}$$

and trigger a signal at the first time i for which $S_i > h$, where $h > 0$ is a suitably chosen signal level. In the above formula, k is the so-called *reference value*; it is typically chosen via formula $k = (\mu_a + \mu_u)/2$.
s_0 is called the *headstart* of the procedure; this value is selected inside the interval $[0, h]$.

The value of h regulates the trade-off between the rate of false alarms and sensitivity requirements. It is typically selected based on the equation:

$$Average\ Run\ Length\{\mu = \mu_a, h\} = ARL_0, \tag{3.3}$$

where ARL_0 is some large value, see Hawkins and Olwell (2012). Once h is selected, we typically validate that the sensitivity of the procedure with respect to the unacceptable values $\mu > \mu_u$, as measured in terms of the *ARL* or other measures of Run Length (RL) performance, are satisfactory. One of the appealing features of the *Cusum*

approach is that the point #5 of the strategy in Section 2 can often be implemented by selecting a suitable value of the signal level *h*, providing that the data volume can accommodate *both* the targeted protection against false alarms and sensitivity requirements.

The *Cusum* procedure was originally introduced as an online detection procedure. However, in this paper, we adapt it to the offline environment to process the stream of the scores $\{Y_i\}$ retrospectively and identify the disturbance periods in a single forward sweep of the procedure. For the offline application, we introduce the concept of an *episode*. The start of an episode is declared once the control scheme exceeds the threshold *h*, and the end of the episode is declared when there is a required degree of certainty that the process of daily counts has returned to the baseline condition. Only at the end of the episode will we be in a position to determine the endpoints of the disturbance the episode is related to.

A simplified version of the disturbance detection scheme is presented below.

Procedure 0. This procedure calls for auto-restarting the scheme (2.1) to the headstart s_0 after each threshold violation. It starts from s_0 and proceeds as follows:

$$S_i = \max[\, S_{i-1} + Y_i - k, 0] \text{ when } S_{i-1} \leq \text{h} \qquad (3.4)$$
$$S_i = \max[\, \text{s}_0 + Y_i - k, 0] \quad \text{when } S_{i-1} > \text{h}.$$

In (3.4), we consider the first exceedance of the threshold (*h*) as the signal that the disturbance has started (beginning of the episode). Subsequent threshold violations (when they occur reasonably soon after the preceding ones) will indicate the continuing disturbance. Eventually, the temporal cluster of threshold violations will stop, providing a basis for declaring the end of the episode. At this point, we will determine the disturbance boundaries. Below is a more detailed description of the process.

We depict the disturbance detection and disturbance period identification process in
Figure **1**1. In this illustration, the headstart value is $s_0 = 0$. The *Cusum*'s process first exceedance over *h* occurred at time *T*, and this is when the episode was declared. After the re-start of the scheme at *T*, we are monitoring the cycles of the control scheme $\{S_i\}$ until the first cycle where the scheme reaches the value 0 before exceeding *h* once again. At that time (*T*+*k*), the *end of the episode* is declared. The index (*T*+*n*) of the maximal value achieved by $\{S_i\}$ in the last cycle $[T, T + k]$ is declared as the endpoint of the disturbance. The index $(T - m)$ of the point at which the scheme trajectory started its elevation towards the first exceedance of *h* in the episode is declared as the starting point of the disturbance. The overall disturbance length is thus $l = m + n + 1$. The control scheme is then re-started until the next episode is declared or the end of the data stream is reached. Some additional tweaks of the algorithm are needed at the right endpoint of the data stream; we omit the details.

Figure 1 illustrates the desirability of using a headstart $s_0 > 0$. With a non-zero headstart, we are running a lower risk of declaring the end of the episode prematurely in the last cycle.

The basic *Procedure 0* can be extended to provide additional possibilities for declaring the end of episode. In some applications, one will want to control more tightly the degree of conservatism in declaring the end of an episode. We thus introduce:

Procedure A. In this procedure, the last cycle uses the "twin" process, S_i^*, which is computed in the same way as S_i – however, it is not reflected at 0 and thus can become negative. A special lower threshold -ζ is used to declare the end of the episode as soon as S_i^* falls *strictly* below it (see Figure 2). At this time, the end of disturbance is set to the index of the point at which S_i^* reaches its maximum; the corresponding value of the scheme is denoted by S_{max}^*. The starting point of the disturbance is determined as the last point *i* where $S_i = 0$ prior to the point of disturbance declaration.

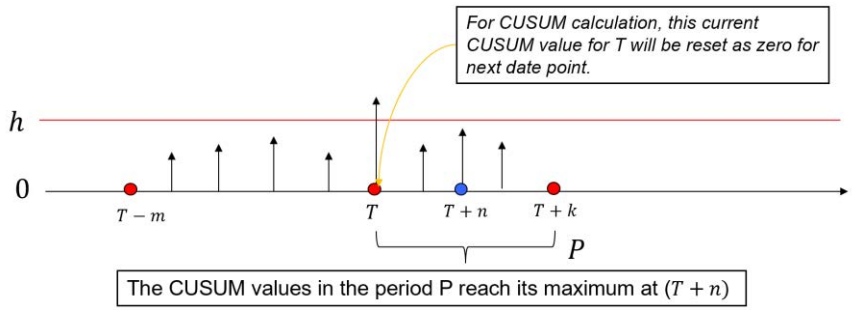


Figure 1. The lifetime of an Episode and the related Disturbance (Procedure 0 with $s_0 = 0$).

The *Procedure A* with $\zeta = 0$ is similar to the Procedure 0; the subtle difference is that in the *Procedure 0*, the end of episode is declared when the evidence reaches the horizontal axis (which also serves as the reflective barrier), while in the *Procedure B* the twin process needs to become *negative* to trigger the end-of-episode declaration. This is in line with the common practice involving control schemes: actions are typically triggered by *strict* exceeding of the thresholds.

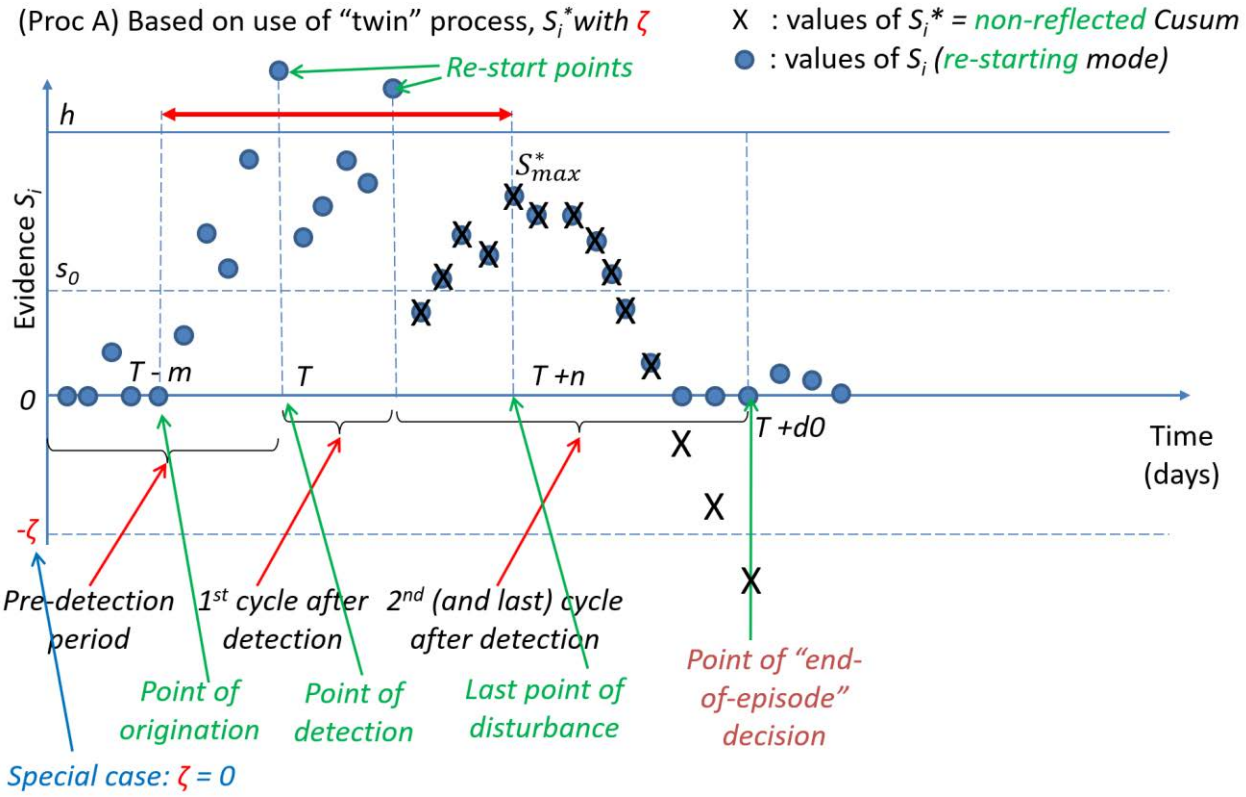


Figure 2. A lifetime of an Episode and the related Disturbance for the *Procedure A*. The end of episode is declared at point ($T+d_0$).

Procedure B. In this procedure, we again use the "twin" process, S_i^* defined in *Procedure A* – however, the criterion for declaring the end of episode depends on the maximal value S^*_{max} of $\{S_i\}$ within the current cycle. If S^*_{max} - $S_i > u$, where $u > 0$ is a suitably chosen parameter, then the end of episode is declared, and the index at

which S^*_{max} is attained is declared as the right endpoint of the disturbance period. The starting point of the disturbance is determined as in *Procedure A*. This procedure is illustrated in Figure 3.
Procedure B offers certain advantages, as it keeps a tighter control on the amount of information needed to declare the end of an episode, independently of the phase the disturbance was in when the last cycle started. However, at the time the end of episode is declared, we still have no definitive statistical "proof" that the *count* process has indeed returned to the baseline conditions. The *Procedure C* described below enables one to impose even a tighter control of the risk that the end of episode will be declared prematurely.

Procedure C. This procedure also uses the "twin" process, S_i^* defined in *Procedure A*. At the decision point ($T+d_2$, see Figure 4), we compute S^*_{max} corresponding to the current cycle and establish SEG_d, the last data segment since the time point (index) corresponding S^*_{max}. For a pre-set level $\varepsilon \geq 0$, we will require evidence that the mean of the scores μ_Y remained below ε for every sub-segment in SEG_d. Formally, we set up a hypothesis testing problem:

H_0: $\mu_Y > \varepsilon \geq 0$ for some sub-segments in the last data segment SEG_d *vs* (3.5)
H_1: $\mu_Y \leq \varepsilon$ for all sub-segments of SEG_d

at a pre-specified level of significance, α. Rejection of H_0 in favor of H_1 leads to the end-of-episode declaration. At that point, the index at which the current value of S^*_{max} is attained is declared the right endpoint of the disturbance period. The starting point of the disturbance is determined as in *Procedure A*.

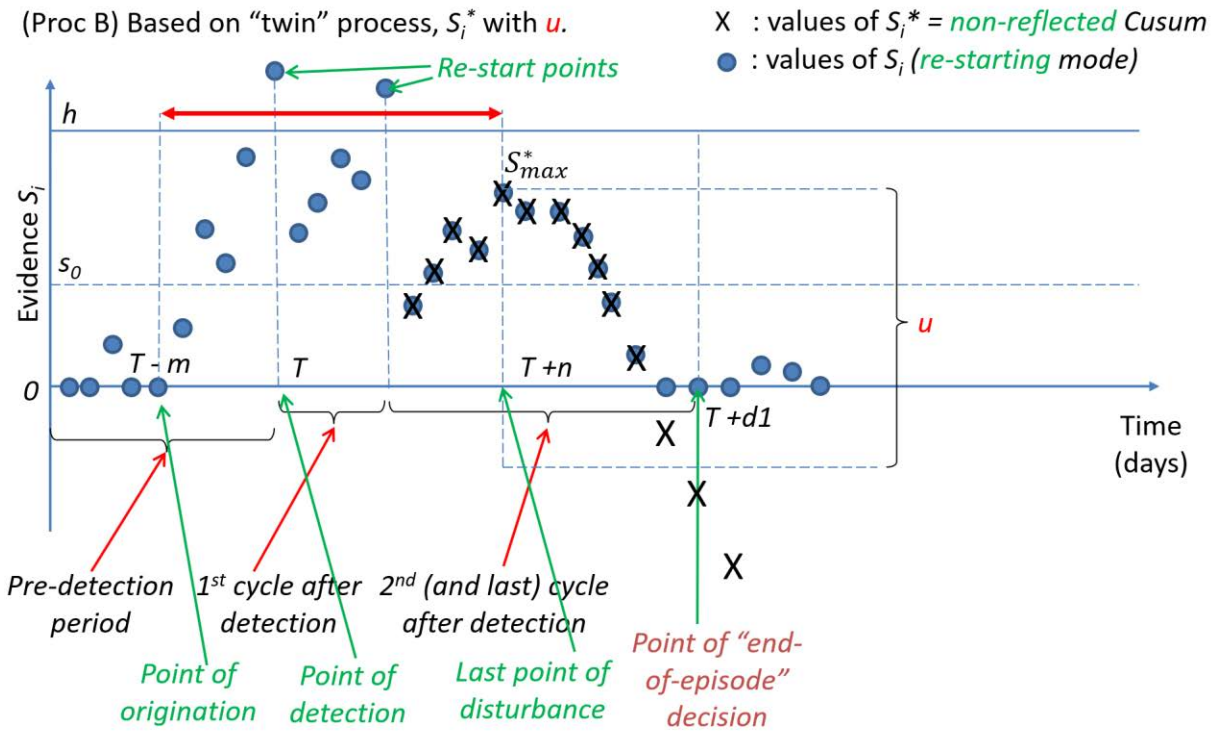


Figure 3. A lifetime of an Episode and the related Disturbance for *Procedure B*. The end of episode is declared at point ($T+d_1$).

One common element of the procedures discussed above is that the point of the disturbance origination ($T - m$) is set to the last point i where $S_i = 0$ prior to the point of disturbance declaration. The *positive bias* introduced by this approach can be addressed in several ways. One possibility is to expand the disturbance starting point

leftward by including additional points (sequentially), as long as the values Y_i support the hypothesis of the elevated rate, e.g.,

(i) as long as $Y_i > \mu_a$, or
(ii) as long as the hypothesis of disturbance vs baseline is supported by the data.

In (ii), we can use a process like that of establishing (T+d), S^*_{max} and (T+n) but scanning the data in the *leftward* direction.

As an alternative approach, we can implement the *dynamic boundary* adjustment policy. In some applications, there is no need to establish the disturbance starting point at the episode detection time T; the operational procedure can also permit adjusting the disturbance boundaries as the new information comes in. In procedures *A-C,* we kept the process of establishing the left disturbance boundary simple. In many practical situations, there is an asymmetry between the process behavior near the left and right endpoints: for example, when disturbances correspond to weather-related storms, the onset (*left* boundary) is often relatively easy to establish – however, determining the *right* endpoint of the disturbance is a more complex task. The main reason is that the elevated counts of events (outage tickets) tend to persist for some time after the physical reasons for the disturbance are no longer in effect.

Note that the procedures of type *A-C* can also be used in conjunction with change detection methods based on the *non-restarting Cusum* technique, see Yashchin (2018).

4. Application: Outage Ticket Labeling

In this section, we illustrate the application of the described methodology in the problem of probabilistic labeling of the electric utility service records (tickets). The records are kept in the Ticket Database (TDB), and the fields of every ticket include the Incident ID, Outage start/end times, Substation, Storm ID, Cause description, and Number of customers affected.

In this application, disturbances correspond to storms. The key question of interest is whether a given service ticket is related to a storm. As noted above, the field Storm ID is present in the TDB – however, this field is often missing or unreliable. Given this data quality issue, our main objective is to bring the TDB to the state where all storms are identified, and tickets labeled as storm-related or not. Since the deterministic labeling is not practically feasible, we wish to assign the probability that it is storm-related to every ticket.

With labeled tickets, we will be able to model the stream of tickets using conventional statistical and machine learning tools. We could also answer the questions of the type:

a. How many storm-related tickets are expected in each period, by substation?
b. What are the contributions of infrastructure factors (number of poles, transformers, miles of lines) to the cost of outages?
c. What are the contributions of Geographic features?
d. What is the effect of weather-related variables (precipitation, wind speeds, wind gusts)?

The application is described in detail in Zhou et al. (2021); in this paper, we will only briefly summarize the results related to the change-point methodology. Our focus is on a special sub-category of the tickets (termed storm-revealing tickets), defined for the purposes of storm identification and probabilistic labeling. The TDB contains more than 140,000 tickets of this type, covering 55 substations over seven years. With the default processing setup parameters, we illustrate the use of Procedure 0 in conjunction with the labeling algorithm.

In line with the strategy outlined at the end of Section 2, our starting point is the estimation of the baseline $\{\hat{\lambda}_{1,i}\}$ for every day *i*, by substation. To detect the storms and establish their boundaries, we use the scores $\{Y_i\}$, with the acceptable/unacceptable means $\mu_a = 0$ and $\mu_u = 2$, respectively. Therefore, the reference value of the *Cusum* detection procedure is $k = (0 + 2)/2 = 1$. Under the assumption that the scores are normally distributed with variance 1, the signal level $h = 3$ should give $ARL_0 = 1962$, and a change in μ_Y from 0 to 2 is detected, on the average, in 3.7 days. However, the actual distribution of the scores is *not* Gaussian because of the strong positive skew, auto-correlation, and the mere fact that the counts are discrete random variables. Our way to address this issue was to use a much higher signal level, $h = 6$. For the relevant values of λ, one could then expect to see about one false alarm in 100 days. The detection capability, of course, suffers too – the Brownian Approximation formula (see Bagshaw and Johnson (1975)) suggests that the change in μ_Y from 0 to 2 would be detectable, on average, in 6.5 days. In practice, however, the selection ($h = 6$, $k = 1$) works quite well – primarily because changes in the scores associated with typical storms tend to exceed $\mu_u = 2$ for low values of the baseline. The elevated false alarm rate can also be remedied, to some extent, by *post-processing*: since we work in an *offline* mode, it is possible to set additional criteria that will eliminate from further consideration storms that appear to be a result of a spurious detection.

The output of the procedure included the modified file, TDBM contains, in addition to the original fields, information about the detected storms, their boundaries, relationship to the *known storms*, and the *probabilistic labels.* The latter we computed using the formula (2.1): for example, for a given day with the number of observed tickets X_i and the estimated baseline $\hat{\lambda}_{1,i}$, the probability that the ticket is storm-related is estimated as

$$\hat{p}_i = \max[0, \frac{X_i - \hat{\lambda}_i}{X_i}], \tag{4.1}$$

for every ticket of the day *i* belonging to the mentioned ticket sub-category of interest.

A sample of periodic counts and the related baseline estimate for one of the substations is shown in Figure 5. In Figure 6, we give the excerpt from TDB, showing the subset of the original fields and the fields added as the result of our analysis. The contents of the fields are modified to preserve customer privacy, so the names of the substations in the field "Substation" are presented as codes. The names of the fields are generally self-explanatory, but we will give explanations related to a few of them.

The three rows in yellow (incidents 3943, 4363) correspond to the newly discovered storm, affecting the substation "fx1". Two of the tickets involved a tree fallen on a line, and one ticket was related to equipment failure (all three happened on January 9, 2013 and affected 65, 270, and 1 customers, respectively). The name given to the newly discovered storm was "fx1_2013-01-08_2013_01_09", indicating that the storm started on January 8, 2013 (i.e., not all the tickets corresponding to this storm are shown in the window of Figure 6). The "Status" field gives the phase of the storm associated with the ticket, as determined by the disturbance detection procedure in Section 3: the letter "E" means that the end of the storm was registered on January 9, 2013. The letter "N" indicates that the ticket was not associated with a storm. The probabilistic label is shown in the field "*P_label*": in accordance with (4.1), all the three tickets were assigned the probability 0.95 of being storm-related. The "Storm_id" field for these records is empty, indicating that the ticket handling system did not associate them with any *known storms*. Furthermore, the last field in Figure 6 indicates that there were no known storms in this time-space neighborhood (as defined in Zhou et al. 2021) that could be plausibly associated with the newly discovered storm.

Another newly discovered storm is shown in orange color. Its leading two tickets 5383, 5523 originated on January 10, 2013, affecting the substation "mx1". The probabilistic labels associated with these tickets are 0.9 and the field "Storm_id" is also empty for them – however, the time-space neighborhood analysis indicates that they are likely associated with the known storm labeled [127000].

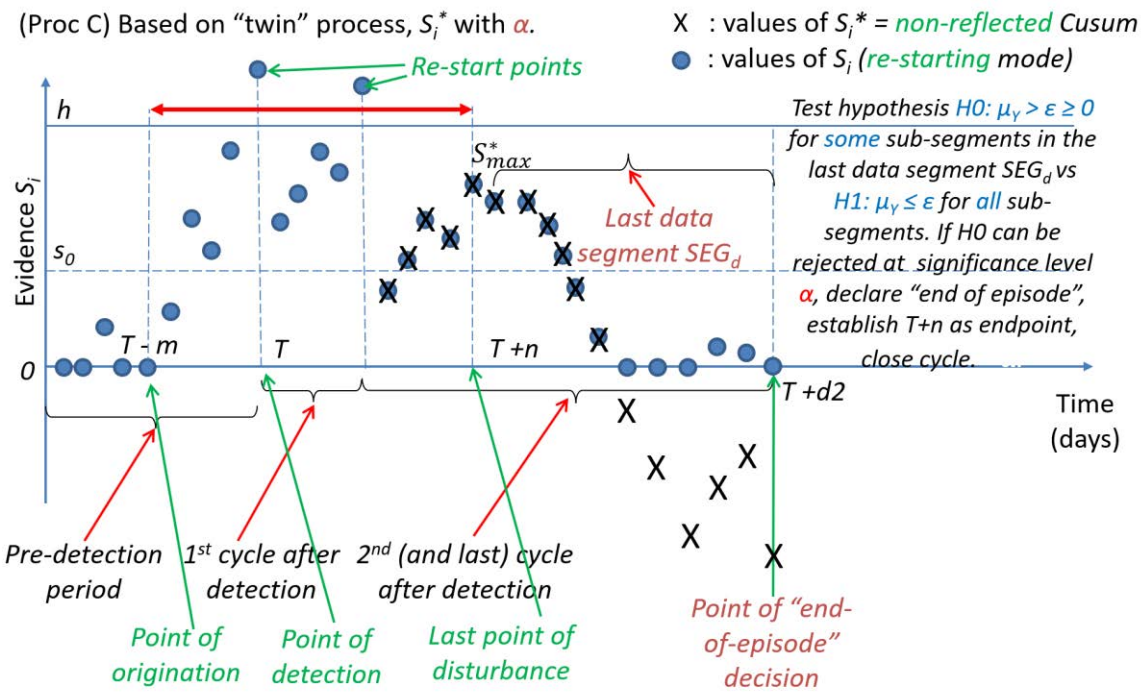


Figure 4. A lifetime of an Episode and the related Disturbance for the *Procedure C*. The end of episode is declared at point ($T+d_2$).

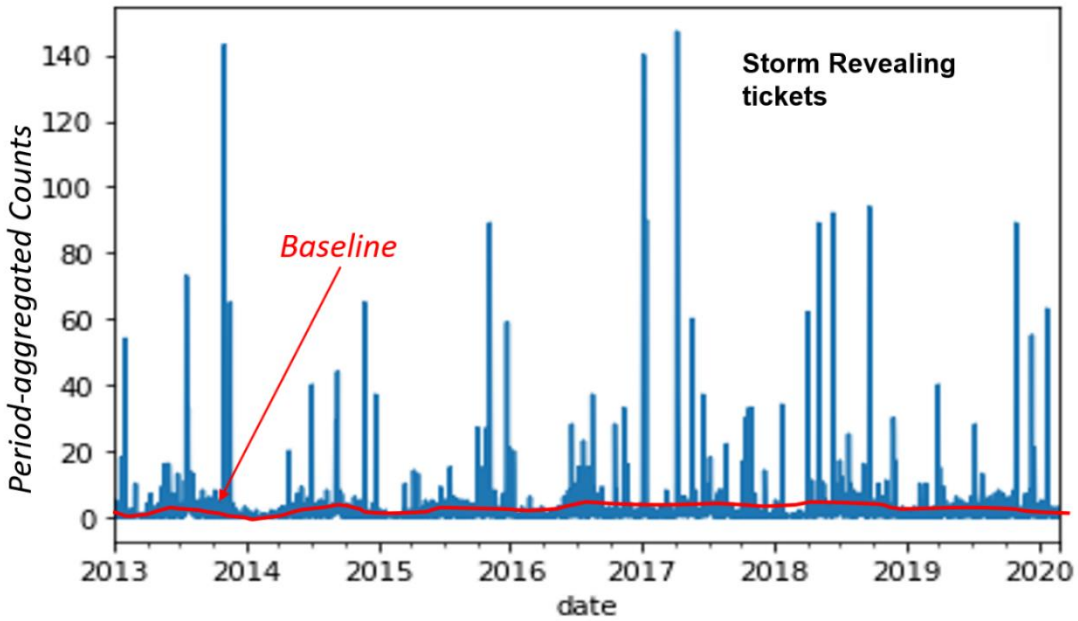
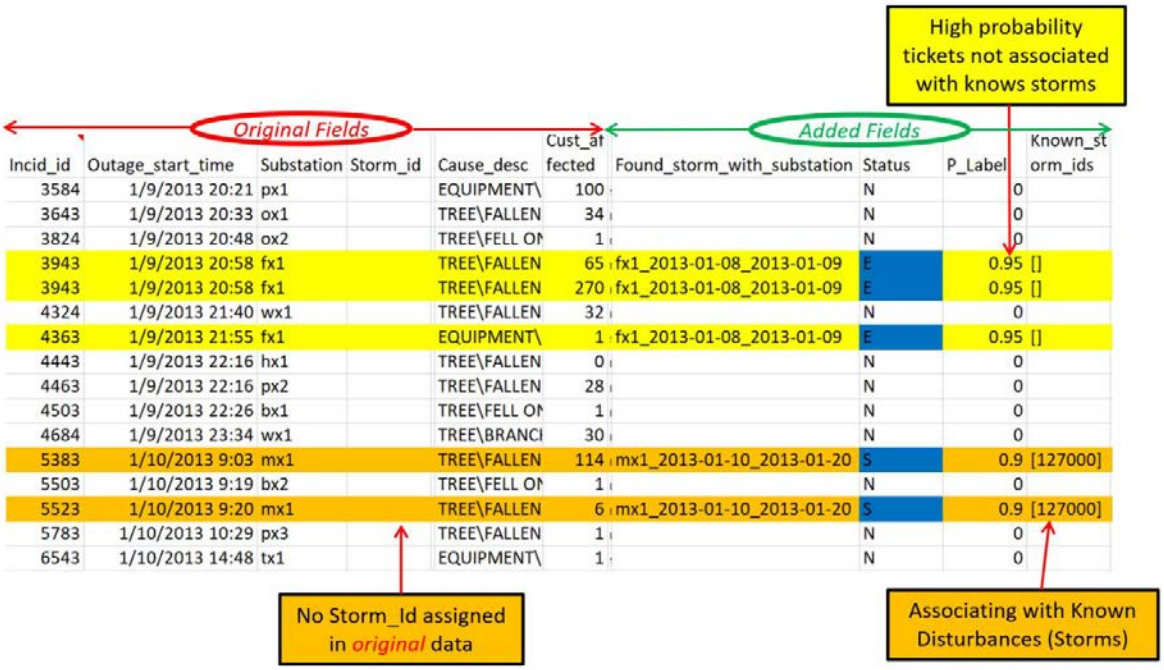


Figure 5. Period-aggregated (weekly) counts for the application in Section 4 and the corresponding estimated baseline, for one substation. Note that seasonality is present in the baseline.



Incid_id	Outage_start_time	Substation	Storm_id	Cause_desc	Cust_affected	Found_storm_with_substation	Status	P_Label	Known_storm_ids
3584	1/9/2013 20:21	px1		EQUIPMENT\	100		N	0	
3643	1/9/2013 20:33	ox1		TREE\FALLEN	34		N	0	
3824	1/9/2013 20:48	ox2		TREE\FELL ON	1		N	0	
3943	1/9/2013 20:58	fx1		TREE\FALLEN	65	fx1_2013-01-08_2013-01-09	E	0.95	[]
3943	1/9/2013 20:58	fx1		TREE\FALLEN	270	fx1_2013-01-08_2013-01-09	E	0.95	[]
4324	1/9/2013 21:40	wx1		TREE\FALLEN	32		N	0	
4363	1/9/2013 21:55	fx1		EQUIPMENT\	1	fx1_2013-01-08_2013-01-09	E	0.95	[]
4443	1/9/2013 22:16	hx1		TREE\FALLEN	0		N	0	
4463	1/9/2013 22:16	px2		TREE\FALLEN	28		N	0	
4503	1/9/2013 22:26	bx1		TREE\FELL ON	1		N	0	
4684	1/9/2013 23:34	wx1		TREE\BRANCI	30		N	0	
5383	1/10/2013 9:03	mx1		TREE\FALLEN	114	mx1_2013-01-10_2013-01-20	S	0.9	[127000]
5503	1/10/2013 9:19	bx2		TREE\FELL ON	1		N	0	
5523	1/10/2013 9:20	mx1		TREE\FALLEN	6	mx1_2013-01-10_2013-01-20	S	0.9	[127000]
5783	1/10/2013 10:29	px3		TREE\FALLEN	1		N	0	
6543	1/10/2013 14:48	tx1		EQUIPMENT\	1		N	0	

Figure 6. Input (Original Fields) and output (Added Fields) of the ticket file view. The probabilistic labels are shown in the field *P_label*. The leading field (*incident id*) indicates that the first two yellow tickets were related to the same incident #3943. The *cause description* field shows limited diversity: it illustrates the use of the sub-family of storm-revealing tickets in the process of storm detection and labeling. One can also see the presence of tickets belonging to this sub-family that affect many customers, but are not associated with a storm, (e.g., incidents #3584, 3643).

Several measures pertaining to the validation of the described methodology for probabilistic labeling are described in Zhou et al. (2021).

5. Concluding Remarks

This paper illustrates the important role of the change-point methods in statistical modeling and machine learning by providing a foundation for addressing the *data quality* issues. For example, in the application described in Section 4, we addressed the problem with the existing labeling approach and achieved a disturbance identification scheme that showed superior properties relative to the alternative approach used by the customer at the time. The concept of probabilistic labels proved to be very useful in this context, and it provided a solid basis for other analytic activities, e.g., identification of global storms affecting groups of substations.

An immediate question is how to establish that a particular change-point methodology is more suitable than others. In the paper, we introduced several procedures and extensions: one common advantage of these procedures is that they are relatively easy to design and administer. The *Cusum* approach enables the efficient determination of disturbance boundaries and is adaptable to various requirements for decision-making time frames. The fine-tuning of the signal levels requires some work, but few simple rules and some experience should suffice. These procedures have their pros and cons – it could be challenging to decide ahead of time which one is most suitable for the application at hand. Of course, they can be compared using simulated data: for example, we could characterize the ability of a procedure to identify the simulated disturbances reliably, estimate the actual false alarm rate and capture the disturbance endpoints correctly. Measures like the mean square error in establishing the end of the simulated disturbance are valuable and well-worth exploring.

At the same time, we need to appreciate the fact that actual disturbances could show strong variability in their patterns and impact, so achieving objective performance measures can be challenging. For example, for storms, it might be known that the effects appear within a short time but fade out gradually. Furthermore, covariates are often available (e.g., weather-related variables, emergency announcements), and their successful incorporation into the scheme could be of value. It is also worth keeping in mind that achieving a high quality of the probabilistic labeling, as measured in the properties of the statistical models based on the resulting labels, is likely to be one of the most important performance factors. In such applications, the change-point procedures are viewed as a tool for achieving this goal, and so their detailed internal performance measures could be considered as being of secondary importance.

Acknowledgments

We deeply appreciate the effort of Harini Srinivasan and Zhangziman Song (IBM AI Applications), who provided the data and highly valuable feedback, help, and support.

References

Beattie, D. W. (1962). A continuous acceptance sampling procedure based upon a cumulative sum chart for the number of defectives. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *11*(3), 137-147.

Cho, H., and Fryzlewicz, P. (2015). Multiple-change-point detection for high dimensional time series via sparsified binary segmentation. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, 475-507.

Dassanayake, S., and French, J. P. (2016). An improved cumulative sum-based procedure for prospective disease surveillance for count data in multiple regions. *Statistics in medicine*, *35*(15), 2593-2608.

Gandy, A., and Lau, F. D. H. (2013). Non-restarting cumulative sum charts and control of the false discovery rate. *Biometrika*, *100*(1), 261-268.

Glaz, J., and Koutras, M. V. (Eds.). (2019). *Handbook of scan statistics*. Springer New York.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Hall, L. M. (2019). *Prospective Disease Surveillance with the CUSUM and Spatial Scan Methods* (Doctoral dissertation, University of Colorado at Denver).

Hall, L. M., and French, J. P. (2019). A modified CUSUM test to control postoutbreak false alarms. *Statistics in medicine*, *38*(11), 2047-2058.

Hawkins, D. M., and Olwell, D. H. (2012). *Cumulative sum charts and charting for quality improvement*. Springer Science & Business Media.

Kulldorff, M., Mostashari, F., Duczmal, L., Katherine Yih, W., Kleinman, K., and Platt, R. (2007). Multivariate scan statistics for disease surveillance. *Statistics in medicine*, *26*(8), 1824-1833.

Lu, Q., Lund, R., & Lee, T. C. (2010). An MDL approach to the climate segmentation problem. *The Annals of Applied Statistics*, *4*(1), 299-319.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J. (2021). Deep Learning--based Text Classification: A Comprehensive Review. *ACM Computing Surveys (CSUR)*, *54*(3), 1-40.

Niu, Y. S., Hao, N., and Zhang, H. (2016). Multiple change-point detection: a selective overview. *Statistical Science*, 611-623.

Ross, S.M. (2014). *Introduction to probability models*, 11th ed., Academic Press.

Snyder, D. L., and Miller, M. I. (2012). *Random point processes in time and space*. Springer Science & Business Media.

Truong, C., Oudre, L., & Vayatis, N. (2020). Selective review of offline change point detection methods. *Signal Processing*, *167*, 107299.

Wasserman, G. S., and Wadsworth, H. M. (1989). A Modified Beattie Procedure Process Monitoring. *Technometrics*, *31*(4), 415-421.

Yashchin, E. (2007). Modeling of risk losses using size-biased data. *IBM journal of research and development*, *51*(3.4), 309-323.

Yashchin, E. (2018). Statistical monitoring of multi-stage processes. In *Frontiers in statistical quality control 12* (pp. 185-209). Springer, Cham.

Yashchin, E., Civil, A., Komatsu, J., and Zulpa, P. (2021). Statistical aspects of target-setting for attribute data monitoring, Frontiers in Statistical Quality Control 13, pp. 99-119, eds. S. Knoth and W. Schmid (Springer)

Zhou, N., Bhamidipaty, A., and Yashchin, E. (2021) "An Approach to Probabilistic Event Labeling and its Applications", submitted for publication.