Real-Time Anomaly Detection and Popularity Prediction for Emerging Events on Twitter

Florian Steuber, Sinclair Schneider, João A. G. Schneider and Gabi Dreo Rodosek
University of the Bundeswehr Munich, Germany
florian.steuber@unibw.de, sinclair.schneider@unibw.de, joao.schneider@unibw.de, gabi.dreo@unibw.de

***Abstract*—Due to their high volume and data recency, communications from social media platforms have become an excellent source for monitoring information diffusion. The insights leveraged are invaluable for social media analysts in the areas of event analysis and emergency management. Existing work ranges from the initial detection of incidents over information enrichment to determining an incident's relevance and life span. Until now, individual parts of this process have been considered separately, but never in combination.**

In this work, we address this crucial need and present an approach for detecting the onset and context of emerging events and predicting their popularity two weeks after emergence on Twitter in real time. Our contribution is threefold. We first present an online learning anomaly detection method refined with temporal clustering to identify abnormal conversational volumes of keywords. Second, we reconstruct potentially underlying events causing the anomaly through the enrichment of contextual and temporal information. Third, we assess an event's relevance and life span by predicting the resonance corresponding tweets receive shortly after their publication.

***Index Terms*—Anomaly Detection, Event Reconstruction, Cascade Prediction, Social Media**

I. Introduction

Twitter and other social media platforms have revolutionized the way we communicate and share information with one another. The ease of access and use enables a high volume of conversations to take place, allowing word of mouth information to spread quickly and widely. Analysts and decision-makers leverage such data for emergency management and early crisis identification. Often the first people to report an incident are users who are at the scene and write about it on social media. In such situations, the timely and accurate extraction and processing of incidents is crucial to support the decision-making process.

Besides the real-time detection of events, it is essential to obtain comprehensive information about their context and life span to plan accordingly and initiate help. To enhance future strategic considerations, obtaining an assessment of the temporal scope of the events is essential, particularly when differentiating between short-term and long-term events to allocate resources effectively.

For instance, while car accidents may not have lasting consequences, events like hazardous incidents and troop movements can impact humanity for longer periods. It is important to recognize medium-term events as they can evolve into lasting trends and draw significant attention from the community beyond emergency management.

To the best of our knowledge, there has been no previous research endeavor that has successfully integrated all aspects of decision support into a unified framework , including identifying events, enriching context, and forecasting attention. We address this issue and present a novel approach for detecting the onset of emerging events and predicting their popularity two weeks after emergence on Twitter in real time.

Our approach consists of three key contributions. First, we present a real-time anomaly detection method to identify the onset of events by analyzing abnormal conversational volumes of keywords. In contrast to more sophisticated approaches, our method relies on basic statistics, requiring less training and complexity, and is best fitting for simple use cases. Second, we perform event reconstruction by enriching the extracted clusters with further context information using frequent itemset mining methods. This process aids in comprehensively understanding the events and their impact. Third, we predict the cascade size of tweets related to the events for a fortnight after their publication as a means to interpret the event's medium-term relevance.

II. Related Work

Anomaly and Trend Detection. Anomaly detection from a topical viewpoint includes topic modeling and clustering techniques, where the detection of outliers is tied to consecutive document similarity comparisons [1]. [2] analyze URLs and related documents contained in tweets to determine topical discrepancies. [3] apply topics extracted by a Latent Dirichlet Allocation as a query set for anomaly detection using sentence transformers. The significance of extracted anomalies and trends can be verified, for example, by incorporating external news corpora [4].

Information retrieval approaches rely on a mixture of document frequency scores and frequency analyses on n-grams to extract trending topics [5]. More advanced approaches allow

ASONAM '23, November 6-9, 2023, Kusadasi, Turkiye

ACM ISBN 979-8-4007-0409-3/23/11…$15.00
https://doi.org/10.1145/3625007.3627517

for the incorporation of recurrent trends, e.g. by modeling Poisson Point Processes [6]. This added value, however, comes with the cost of requiring more computationally complex methods or a previous offline learning phase [7]. In contrast, our approach is based on simple statistical methods that allow for direct usage while still providing competitive results. For a more comprehensive overview of detection methods see [8].

Popularity Prediction. The prediction of post popularity in terms of retweet counts and tweet cascade sizes has been extensively addressed in previous research. Typical approaches to predict the retweet count of posts rely on GNN models [9] or deep multitask learning [10]. Other methods to predict a cascade's future size include time series predictions [11], cascade graphs [12], and LSTMs [13].

Feature analyses in retweet prediction tasks find that structural features have a higher impact on the forcast than textual content [14]. [15] perform an in-depth correlation analysis and come to the conclusion that temporal features further enhance a model's predictive power. In particular, monitoring posts during their early activity period and incorporating proxy variables for early diffusion dynamics significantly increases the forecast's accuracy [16]. This so called *prediction with peeking* performs better than conventional ex-ante predictions [17]. Albeit temporal features are predictive in short-term predictions, they deteriorate for long-term tasks because of the bursty nature of information diffusion in general [18].

III. Methodology

This section proposes an online learning approach to detect and track emerging events through real-time data streams from social networks like Twitter. First, a summarizing overview is provided, as can also be seen in Figure 1, and each component is explained in detail.[1]

Text preprocessing of tweets is performed by employing SpaCy for Part-of-Speech tagging, Named Entity Recognition, as well as stop word removal and text lemmatization. Within each post, word tokens are categorized as either *strong* or *weak* context information. Tokens considered strong encompass hashtags, named entities, and proper nouns, while all other tokens are categorized as weak. Different parts of the framework rely on either of those categories as input.

We perform real-time anomaly detection by processing a time series of strong tokens to identify statistical outliers that may be caused by an emerging event. These outlier points are further subjected to temporal clustering to improve the statistical significance and reduce false detection rates.

For any identified temporal cluster of anomaly points, we collect further information from related posts using Frequent Itemset Mining (FIM) techniques. This context enrichment helps analysts recognize potentially underlying events causing the abnormal conversational volumes of the given token.

Finally, we predict the future popularity of tweets related to the event to determine its medium-term impact on the network. The resulting statistics yield a reliable estimate of the topic's future volume.

[1]Please note that supporting code snippets for parts of the framework are available at https://github.com/fsteuber/adbc.

A. Time Series Convolutions

The framework's first component identifies large and spontaneous increases in a word token's conversational volume in real time. For processing, we keep for each strong token a time series aggregated into five-minute bins, which contain the token's occurrence numbers. A detector algorithm operates on individual time series, outputting an anomaly score for each time step. Three different detectors are examined, both individually and in combination to obtain meaningful scores.

Each detector has a memory of past time steps, implemented using a sliding window approach. The first detector calculates the z-score of the latest point in the time series given its current window slide. The second detector performs a linear regression on past data points within the sliding window. It then determines the residual between the most recent and expected value given the computed regression line. In the third and final detector, an additional smaller window is used which includes fewer data points of the past. The occurrence frequencies in both sliding windows are first normalized on a per-minute basis and then set in relation to each other. Larger deviations at more recent time points are thus represented by more extreme ratios.

The detector's output is further normalized by a monotonically increasing softmax function for better comparability. We combine partial results when multiple detectors are used together by first computing their individual values and then determining their geometric mean.

The calculated anomaly scores are used to define a decision threshold, which classifies corresponding bins as abnormally frequent. The decision threshold depends on the application and acceptable error rates. To further increase the statistical significance of detected candidate points, additional temporal clustering is performed using a density-based clustering algorithm.

In order to obtain authentic constraints for the clustering algorithm, we examined the time series of 20 manually extracted rapidly emerging events in the past. Each event made it into regional or local media and is also observable on Twitter with a dedicated hashtag (e.g. Stephen Hawking's death or Trump's impeachment procedure). For all events, we examined how long it took from the first increase in the time series above its range of expected values to its maximum peak. The duration of these initiation phases depends on the popularity and controversy of the topic and ranges from 15 minutes to multiple hours.

Naturally, the herein presented anomaly detection should account for time intervals of comparable size. We incorporated these findings into a DBSCAN algorithm using a minimum cluster size of three points and a 15-minute ϵ-neighborhood. For each cluster thus identified, the smallest and largest time points serve as time limits for this anomaly.

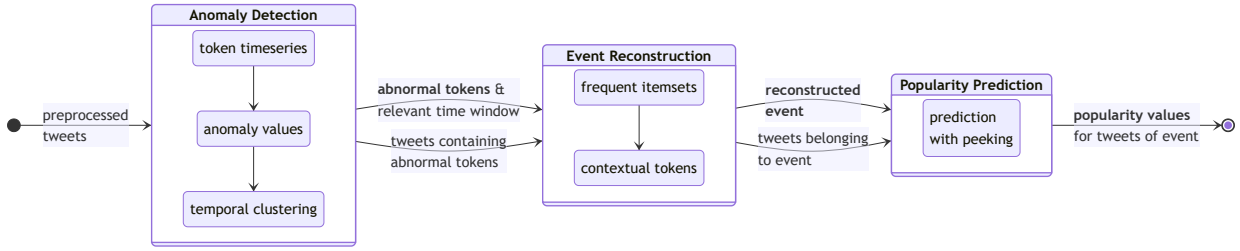

Fig. 1. General workflow of the framework

B. Context Enrichment

The framework's next contribution is to extract additional information about a potentially underlying event that caused the anomaly. This context enrichment procedure involves identifying frequently co-occurring tokens from tweets containing the pivot token which are composed during the anomalies' time period.

A hereby reconstructed event hence can be seen as a set of semantically meaningful and interrelated tokens, which co-occur significantly often in a delimited, but clearly defined time frame. The ensemble of such presented information aids analysts in determining the cause of abnormal conversational volumes and react accordingly.

For the context enrichment procedure we extract frequently co-occurring tokens using Frequent Itemset Mining methods [19]. Tweets are interpreted as transactions, with previously prepared word categories of strong and weak tokens serving as basic elements. As a result, we obtain two sets of transactions, one for each category of words, which are processed separately.

We calculate frequent itemsets for all transactions arising from the strong context group with a minimum support ≥ 0.2 using the efficient FP-Growth implementation to account for the requirement to process data in real time. To incorporate tokens from the weak context category, we further employ a maximum support threshold ≤ 0.6 as an effort to counteract spam and repetitive tweets.

C. Popularity Prediction

In many situations, it can be helpful to get a rough estimate of the relevance an event's topic receives in the near future, for example, to distinguish between short-term incidents with no lasting impact and long-term trends that continue attracting the communities' attention. We map the problem of assessing a topic's mid-term relevance to another related, but easier-to-solve problem. Specifically, we examine how popular tweets associated with the detected event will be two weeks after publication, as measured by the number of retweets and likes (*favourites* on Twitter). A representative number of such impressions provides a reliable indication of the event's future continuance.

To solve this task, we train a machine learning model using *prediction with peeking*. This approach delays the forecast of a tweet published at time t_0 until a later time t_1 with the intention to include additional early diffusion patterns into the feature set. Since the goal is to make a forecast two weeks after the tweet is published at t_0, there arises a time constraint of $t_1 - t_0 < 2$ weeks.

For convenient data collection, we restrict on processing any sort of follow-up tweets, i.e. retweets, quotes, or replies. These tweets have embedded their referenced original tweet and hence contain all relevant early diffusion information, including t_1, i.e. the publication time of the follow-up tweet. For further optimization, we require data points to have a minimum early adopter count (retweets or likes) of $k \geq 5$ at t_1 to be considered as a training sample. This action increases the sensitivity of the model in predicting popular items.

Data collection is performed using Twitter's Sample Stream. Over a period of four weeks, we were able to extract roughly one million data points for training the classifiers. For each data point collected, we queried the original tweet two weeks after its creation to determine the final number of retweets and likes, i.e. favorites, which are used as target variables.

From all training samples, we extracted various features concerning the tweet author's profile, including account age, the number of followers and friends, their total status count, and published tweets per day, as well as whether the profile is verified or not. Furthermore, we extracted features representing the author's activity four weeks prior to t_0. In particular, we obtained their past retweet, favourite and status count as well as the number of unique @-mentions by querying their timeline.

Given the set of training samples, we train different machine learning classifiers to determine the number of retweets and favorites of an original tweet at t_2. Since in most cases, it is not the exact number of retweets that is relevant, but rather the general trend, we simplify the predictive task from a machine-learning perspective by constructing a balanced classification scenario. This can be accomplished by examining the distribution of popularity scores across all collected tweets and calculating the corresponding quantile bounds. These are used to convert numerical values for retweet and favorite counts into equally sized bins, which determine the label of the data point.

Using four bins is usually sufficient to accurately assess the popularity of a tweet and its cascade. By combining the classification results of multiple tweet cascades belonging to the same event, the resulting statistics provide a reliable ranking of the size and importance of the event and its potential to develop into a medium-term trend.

TABLE I
RECONSTRUCTED INFORMATION FOR APPLE TIMESERIES

Attribute	Value
Time Frame	2020-01-07 16.20-16.30
Frequent Itemsets	FBI, iPhones, Apple News
Context Information	have, FBI, ask, unlock, news
Real World Cause	FBI asks Apple to unlock iPhones belonging to a gunman involved in a previous shooting

TABLE II
ACCURACY OF 2- AND 4-CLASS PREDICTIONS

	ACC 2-Class Prediction			ACC 4-Class Prediction		
Scenario	**DT**	**LR**	**SVM**	**DT**	**LR**	**SVM**
RETWEET						
Normal	**.9833**	.7581	.5388	**.9382**	.5000	.2923
Restricted	.9766	.8019	.7292	.9321	.5525	.4827
FAVOURITE						
Normal	.9766	.7871	.5374	**.9358**	.5290	.2820
Restricted	**.9788**	.7965	.7604	.9302	.5636	.5115

IV. EVALUATION

A. Anomaly Detection

We use Numenta Anomaly Benchmark (NAB) to test our anomaly detection method. NAB has 58 datasets of labeled time series from various domains and time resolutions. We focus on the analysis of time series of various companies collected on Twitter. An overview of benchmark scores different algorithms achieve on NAB can be found in [20].

We preprocess the NAB time series data by aggregating bins to five-minute intervals and then apply the presented detector algorithms to compute anomaly scores. The results are evaluated using automated thresholding and comparison with labeled data in the NAB framework.

We analyzed how window sizes affect ratio detector performances (Figure 2A). Results show that memory sizes over 1000 minutes (roughly a day) for the large window provide competitive results. The performance increase diminishes after the two-day mark. Time intervals of a day or more allow for accounting for varying conversational volumes during day and night. Furthermore, higher benchmark scores for the comparison window are achieved using smaller window sizes, because short-term frequency spikes carry more weight. Other detectors exhibit similar performance behaviors regarding window sizes.

Using optimal window sizes, the z-score detector performs best on the benchmark, achieving 51.66% on standard and 66.55% on false negative profiles. These scores are followed by the ratio detector with 49.47% and 63.87% and the residual detector with 40.81% and 54.75%, respectively. Combining multiple detectors for anomaly detection results in benchmark scores of 51.23% for standard and 65.58% for false negative profiles.

B. Event Reconstruction

We conduct a case study on Twitter to evaluate the qualitative output of event recognition, using a methodology similar to that employed by the NAB benchmark. We focused on tweets that mentioned prominent companies like Apple, Facebook, Google, and Amazon. To be considered, the company name had to be mentioned at least once in the tweet. To ensure reproducibility, we utilized a self-collected dataset from various freely available Twitter APIs, consisting of approximately one billion data points from the first two months of 2020.

We detected anomalies in each company's token occurrence timeline by clustering anomaly scores above a threshold set at the 99th percentile to obtain significant time intervals. For each cluster, we identified the corresponding time interval and retrieved all respective tweets containing the company name. Using these tweets, we determined frequent itemsets for the respective cluster.

We listed the frequent 1-itemsets of strong and weak tokens for one exemplary event in Table I. It is extracted from the time series of the company Apple on 2020-01-07. Through the extraction of frequent context words belonging to the strong context group, the main actors of the event could be identified as the FBI and the iPhone. Incorporating additional context tokens yields further insights into the event, revealing that the FBI had requested Apple to unlock certain iPhones in a news report. Subsequent investigation into the real-world events associated with this anomaly point revealed that the FBI's request was in response to a gun shooting that occurred approximately a month earlier.

C. Popularity Prediction

To estimate the popularity of a tweet cascade, we estimate the number of likes and retweets an original tweet will receive two weeks after publication. The exact number of retweets in such scenarios is rarely of interest, but rather the general trend. Hence, we construct a balanced classification scenario, placing exact retweet numbers in different popularity ranges with equal occurrence probability.

Figure 2B shows the density function of collected *retweet counts* and *favorite counts* (i.e. likes) of original tweets two weeks after their publication. Note that the figure only shows values for tweets that already met the early adopter requirement $k \geq 5$ when they were processed at t_1. In practice, a large proportion of tweets do not receive a significant number of replies, so the real distribution curse is much more left-leaning.

We compared different machine learning algorithms, including Decision Trees (DT), Logistic Regression (LR), and Support Vector Machine (SVM) for classification. Two experiments were conducted for each model, forecasting either on *retweet count* (RT) or on *favorite count* (FAV). 1 million samples were used, with 85% for training and 15% for testing. Scoring was done using 5-fold cross-validation.

We report model qualities for the balanced classification task using either two or four classes, each split at their respective quantile boundary in Table II. For the two-class scenario we find that for both target variables, DTs are more accurate

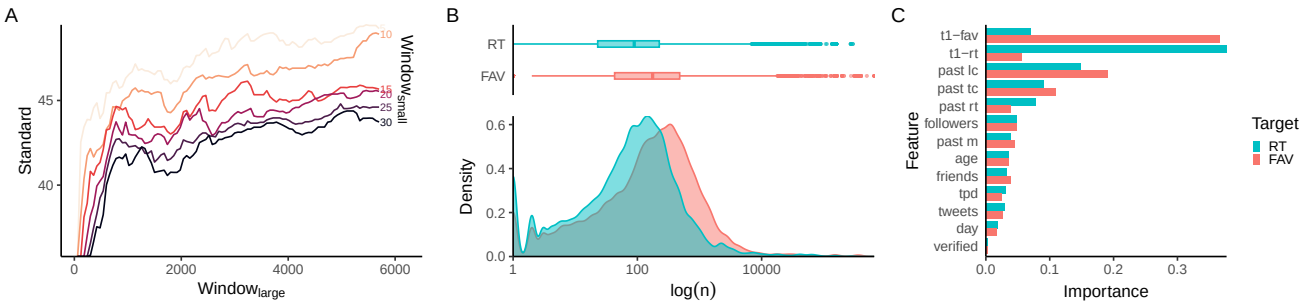

Fig. 2. A. Empirical results for varying window sizes, B. Density of popularity values at t_2, C. Feature importances for scenario 4-C/RT vs 4-C/FAV

than LR or a SVM and reach accuracy values of roughly 98%. In the *restricted* scenario we ignored the largest 10% of cascades in an attempt to reduce their impact in distorting class boundaries.

When extending the classification to four classes, DTs again provide the best results for both variables in relative terms, with a precision of almost 94%. However, the performance of LR and SVM drops sharply compared to their 2-class equivalents. In addition, we observe that the accuracy of DT is slightly reduced by eliminating the largest 10% of tweet cascades.

Finally, we analyze feature importances in Figure 2C. Temporal information gained by peeking has the largest impact, followed by recent activity and the number of followers. Other features, such as account age or the verification of a user profile, are less significant.

V. Conclusion and Future Work

We proposed a novel online learning approach for event detection and trend analysis on social media. Utilizing real-time Twitter data, our method establishes emerging events by identifying abnormal conversational volumes through anomaly detection and temporal clustering on token timelines. Extracted anomalies are enriched with contextual information to provide deeper insights into the underlying event, thus granting a valuable decision aid for social media analysts. For each extracted event, the framework predicts the popularity of related tweet cascades two weeks after their publication, providing further insights into the event's future relevance and lifespan.

Future work could involve testing various detection methods for anomalies that account for seasonal or long-term cyclic patterns. This approach could also be applied to other platforms and domains.

References

[1] S. Kumar, M. B. Khan, M. H. A. Hasanat, A. K. J. Saudagar, A. AlTameem, and M. AlKhathami, "An anomaly detection framework for twitter data,"

[2] P. Anantharam, K. Thirunarayan, and A. Sheth, "Topical anomaly detection from twitter stream," in *Proc. 4th Annu. ACM Web Sci. Conf. (WebSci)*, vol. 4, Jun. 22–24, 2012, pp. 11–14, doi: 10/j8xs.

[3] S. Kumar, M. B. Khan, M. H. A. Hasanat, A. K. J. Saudagar, A. AlTameem, and M. AlKhathami, "An anomaly detection framework for twitter data," *Appl. Sci.*, vol. 12, no. 21, Nov. 2022, Art. no. 11059, doi: 10/j8xx.

[4] K. S. Karimi, A. Shakery, and R. M. Verma, "Enhancement of twitter event detection using news streams," *Natural Lang. Eng.*, vol. 29, no. 2, Mar. 2023, pp. 181–200, doi: 10/ktzm.

[5] J. Benhardus and J. Kalita, "Streaming trend detection in twitter," *Int. J. Web Based Communities*, vol. 9, no. 1, Jan. 2013, pp. 122–139, doi: 10/ggfmm6.

[6] S. Hendrickson, J. Kolb, B. Lehman, and J. Montague, "Trend detection in social data," Twitter Inc. 2015. [Online]. Available: https://developer.twitter.com/content/dam/developer-twitter/pdfs-and-files/Trend-Detection.pdf

[7] Twitter Inc. *Anomaly Detection with R*. GitHub. (2015). Accessed: May 05, 2023. [Online.] Available: https://github.com/twitter/AnomalyDetection

[8] X. Hu *et al.*, "Event detection in online social network: Methodologies, state-of-art, and evolution," *Comput. Sci. Rev.*, vol. 46, Nov. 2022, Art. no. 100500, doi: 10/ktzn.*Appl. Sci.*, vol. 12, no. 21, Nov. 2022, Art. no. 11059, doi: 10/j8xx.

[9] C. T. Lo, Y. H. Lee, and J. H. Peng, "Real-time retweet count prediction using GNN model," in *9th Int. Conf. Appl. Syst. Innov. (ICASI)*, vol. 9, Apr. 21–25, 2023, pp. 181–183, doi: 10/ktzd.

[10] J. Wang and Y. Yang, "Tweet Retweet Prediction Based on Deep Multitask Learning," *Neural Process. Lett.*, vol. 54, no. 1, Feb. 2022, pp. 523–536, doi: 10/ktzh.

[11] I. N. Lymperopoulos, "RC-Tweet: Modeling and predicting the popularity of tweets through the dynamics of a capacitor," *Expert Syst. with Appl.*, vol. 163, Jan. 2021, Art. no. 113785, doi: 10/j8x3.

[12] Y. Shang *et al.*, "Popularity prediction of online contents via cascade graph and temporal information," *Axioms*, vol. 10, no. 3, Jul. 2021, Art. no. 159, doi: 10/gndv36.

[13] J. Wen, Z. Zhang, Z. Yin, L. Sun, S. Su, and S. Y. Philip, "DeepBlue: Bi-layered LSTM for tweet popularity estimation," *IEEE Trans. Knowl. and Data Eng.*, vol. 34, no. 10, Oct. 2022, pp. 4737–4752, doi: 10/j8x9.

[14] R. O'Grady, "Times change and your data should too: The effect of training data recency on twitter classifiers," SANS Institute, White Paper, Jul. 2018. Accessed: Mar. 4, 2023. [Online]. Available: https://sansorg.egnyte.com/dl/mo5V4qRpDF

[15] S. Sharma and V. Gupta, "Retweet prediction for large datasets of random tweets," in *Proc. 2nd Int. Conf. Data Sci., Mach. Learn. and Appl. (ICDSMLA)*, A. Kumar, S. Senatore, and V. K. Gunjan, Eds. Nov. 21–22, 2022, pp. 665–673, doi: 10/ktzf.

[16] B. Shulman, A. Sharma, and D. Cosley, "Predictability of popularity: Gaps between prediction and understanding," *Proc. 10th Int. AAAI Conf. Web and Social Media (ICWSM)*, vol. 10, no. 1, May 17–20, 2016, pp. 348–357, doi: 10/j8w7.

[17] Y. Zhang, M. Feng, K. Shang, Y. Ran, and C. J. Wang, "Peeking strategy for online news diffusion prediction via machine learning," *Physica A: Statist. Mech. Appl.*, vol. 598, Jul. 2022, Art. no. 127357, doi: 10/ktzj.

[18] R. M. Cao, X. F. Liu, and X. K. Xu, "Why cannot long-term cascade be predicted? Exploring temporal dynamics in information diffusion processes," *R. Soc. Open Sci.*, vol. 8, no. 9, Sep. 2021, Art. no. 202245, doi: 10/gnf5gg.

[19] C. H. Chee, J. Jaafar, I.A. Aziz, M.H. Hasan, & W. Yeoh, "Algorithms for frequent itemset mining: a literature review," Artificial Intelligence Review, 52, 2019, 2603-2621.

[20] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms: The numenta anomaly benchmark," in *IEEE 14th Int. Conf. Mach. Learn. and Appl. (ICMLA)*, Dec. 9–11. 2015, pp. 38–44, doi: 10/gf5ctc.