Defining and Validating Similarity Measures for Industrial Alarm Flood Analysis

Marta Fullen*, Peter Schüller‡, Oliver Niggemann*†
*Fraunhofer Application Center Industrial Automation, Lemgo, Germany
‡Marmara University, Istanbul, Turkey
†Institute Industrial IT, Lemgo, Germany

***Abstract*—Industrial plant operators regularly observe a high number of alarms generated in a short period of time, a phenomenon which is referred to as alarm flooding. This causes plant downtime, not only because of the repair time but also by the time needed to identify the root cause of machine failure—which is difficult during an alarm flood. Therefore, diagnosis tools that perform root cause analysis to advise plant operators can help reduce the downtime, which is a crucial issue in industry. We analyse the reproducibility and applicability of an existing approach by Ahmed et al. (2013) which is based on agglomerative hierarchical clustering where raw data in the form of alarm logs is preprocessed, floods are detected, and then clustered. The aim is, that resulting clusters represent floods that originate from the same common root cause. We extend the approach with alternative similarity measures and perform experiments regarding their effectiveness in structuring industrial alarm flood data. In our evaluation we use a real industrial use case which contains more diverse data and a larger amount of data points compared with the original study.**

I. Introduction

The phenomenon of alarm flooding is a recurring problem in industrial plant operation [2]. It occurs when the number of alarms annunciated in succession is so high that it exceeds the operators capability of understanding the situation. This creates a dangerous situation where the operator might overlook critical alarms that could lead to significant downtime, irreversible damage or even loss of life [3].

Diagnosis of a failure of an industrial plant is a non-trivial task that requires extensive knowledge of causalities between symptoms produced by the system [4]. Unfortunately, this information is often not available, and in these cases shallow data-driven approaches are more applicable. Data-driven approaches are based purely on the data obtained from the system, possibly with rudimentary expert knowledge inserted when available to improve the results. Data-driven methods directly analyse, manage and reduce the alarm annunciation and therefore flooding [5], without a semantic representation of the system. Multiple approaches exist to this end, drawing from the data mining fields such as sequence identification and pattern recognition [6], [7], correlation analysis [8] or visualisation [9]. Many of these approaches utilise flood similarity measure of some kind, e.g., [10], [11].

Alarm flood detection and clustering is a data-driven approach to handle alarm flooding. An operator assistance system can detect a newly annunciated alarm flood, compare it to the previously seen floods and identify the most similar cluster. If the history of flood of the plant has annotations, such as a log of repairs done to remedy the reason of an original flood, the system can make a suggestion to the operator regarding the fault diagnosis and repair procedure.

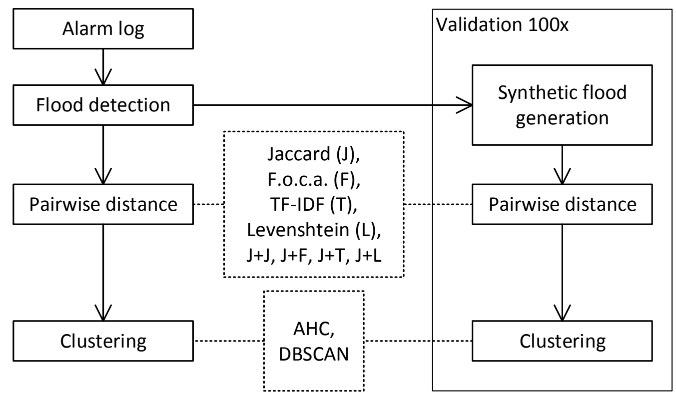


Fig. 1: Flood clustering and validation data flow. Flood distance measures are Jaccard distance (J), Frequency of consecutive alarms (F), TF-IDF (T), and Levenshtein distance (L). Stacked measures J+Ω use J as preprocessing for measure Ω (see II-B).

A major challenge in creating such a system is determining how similarity between alarm floods should be defined. A multitude of similarity measures (and, analogously, distance measures) exists in the field of data mining and clustering [12]. Another challenge is, that real industrial alarm data is difficult to work with, e.g. because of a high volume of alarms, or because of poor alarm system design. While a certain similarity measure might work for a certain application, there is no guarantee it will work in other application scenarios, as there is not systematic method for quantifying the usefulness of a given similarity measure in the industrial setting.

We here propose a semi-formal approach to answering that issue by defining an experimental design to validate the behaviour of a distance measure in regard to alarm flood clustering. Analysis of the behaviour of the distance measure can then help to choose the most suitable distance measure. We also reproduce and extend the alarm flood detection and clustering approach by Ahmed et al. [1] with additional similarity measures based on TF-IDF [13] and Levenshtein distance [14], apply these measures to a large real industrial alarm log, and evaluate them using our validation method.

Our study shows that the flood distance measure in [1] behaves significantly different from our added distance measures, in particular our measures produce a more stable clustering in the presence of noise in the data.

Methodology for detecting floods and computing distance measures is given in Section II. Section III describes our proposed approach for validating the behaviour of clustering results using alarm floods synthetically generated from real alarm flood data. Section IV presents and discusses results of empirical validation on a real industrial dataset. We summarize the results and conclude in Section V.

II. Methodology

Alarm flood clustering consists of three steps: (i) flood detection, (ii) calculation of pairwise distances using a chosen flood representation and distance measure and (iii) clustering. All clusterings obtained in our study are obtained following this methodology, which is shown in Fig. 1 on the left side. We next explain how we realize these steps concretely, which prior work we use and justify our adaptations of prior work for the purpose of flood clustering.

A. Flood detection and preprocessing

Flood detection is based on the alarm flood definition in the industry standard for Management of Alarm Systems for Process Industries: "alarm annunciation rate of more than 10 alarms per 10 minutes exceeds the operator capabilities and is considered an alarm flood" [15]. Floods are detected using a sliding window. We define that a flood begins where the alarm annunciation rate exceeds 10 alarms per 10 minutes, and a flood ends where the alarm annunciation rate drops to under 5 alarms per 10 minutes. Considering that the root cause of an alarm flood is likely to appear before the actual flood begins, the alarms that were annunciated before but are still active during the flood are included in it. Similarly, alarms that were activated during the flood and are active beyond the flood end time-stamp are also included. This approach is slightly different to [1] where the alarm flood is considered to end when the rate of alarms per ten minutes drops to zero. In our dataset, the rate of alarms per ten minutes rarely drops down to zero and therefore the approach of [1] yields just a few floods, each spanning days or weeks, which is not a practically relevant result. Instead, we consider the flood to end when the alarm annunciation rate is again manageable for the operator.

Chattering alarms are a common occurrence in industrial alarm logs [16]. Such alarms are triggered and deactivated in quick succession, for example due to a variable oscillating around an alarm threshold. Ideally, chattering should be foreseen and prevented at the alarm system design stage, but in reality it is not the case, and chatter frequently clutters the alarm log with unnecessary entries. Also for industrial flood analysis, chattering obscures the real behaviour of alarms and disrupts flood detection and flood classification. Therefore, before the floods are detected, chattering alarms are merged: annunciations of the same type of alarm which occur within 1 minute are combined into one entry.

B. Flood representation and distance measure

Choice of an appropriate distance measure is crucial when attempting to cluster alarm floods. Moreover, each distance measure requires a specific flood representation.

1) Jaccard distance (J): Jaccard distance [17] is the ratio between alarm types occurring only in one of the two floods, and the number of alarm types in both floods. Each flood f_i is represented as a binary vector $f_i = (a_1, a_2, \ldots, a_m)$, where m is the number of unique alarm identifiers in the complete alarm log and a_j is a binary value representing whether an alarm appeared in the flood, regardless of its count. The Jaccard distance between floods f_i and f_j is given by

$$J_{ij} = \frac{|f_i \text{ xor } f_j|}{|f_i \text{ or } f_j|}, \tag{1}$$

where $|x|$ returns the number of true values in vector x. Alarm types that are absent from both floods are irrelevant for Jaccard distance. This measure was used as preprocessing in [1].

2) Frequency of consecutive alarms (F): This measure was proposed in [1] based on a simplification of first-order Markov chains. Each flood is represented as a matrix of counts of each pair of alarms appearing consecutively,

$$P = \begin{bmatrix} f_{11} & f_{12} & \dots & f_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{m1} & f_{m2} & \dots & f_{mm} \end{bmatrix}, \tag{2}$$

where f_{ij} is the frequency of alarm a_j being annunciated directly after alarm a_i in a given alarm flood. Then, the distance between two floods can be calculated as a distance between their P matrices, e.g. using Frobenius distance.

3) Term frequency-inverse document frequency (T): TF-IDF is a measure often used in natural language processing to weight terms in a document according to how frequent and discriminative they are with respect to a document collection [13]. We apply TF-IDF to weight alarms in alarm floods with respect to the collection of all floods. TF-IDF is calculated for each alarm a and flood f as

$$\text{tf-idf}(a, f) = \text{tf}(a, f) * \text{idf}(a). \tag{3}$$

Term frequency is calculated as

$$\text{tf}(a, f) = \frac{f_{a,f}}{|f|}, \tag{4}$$

where $f_{a,f}$ is the number of annunciations of alarm a in flood f and $|f|$ is the total number of alarm annunciations in f.

Inverse document frequency is calculated as

$$\text{idf}(a) = \log_e \frac{|F|}{|\{f \in F \mid a \in f\}|}, \tag{5}$$

where $|F|$ is the total number of floods. TF-IDF score is calculated for every alarm type and every flood in the log and yields a flood representation in the form of a vector of length m, the total number of unique alarm signatures.

Two floods f_i and f_j can then be compared using a distance measure such as Euclidean distance between two vectors:

$$d(f_i, f_j) = \sqrt{\sum_{k=1}^{m} (\text{tf-idf}(a_k, f_i) - \text{tf-idf}(a_k, f_j))^2}. \tag{6}$$

4) Levenshtein distance (L): This metric counts the amount of "edits" that are needed to transform one sequence into another one, where an edit is a symbol insertion, symbol deletion, or a symbol substitution [14]. To apply this metric, alarm floods are represented as sequences of symbols, which in turn represent unique alarm types. The Levenshtein distance $d(|f_i|, |f_j|)$ between floods f_i and f_j is calculated recursively, where the distance for the first x and y symbols of f_i and f_j, respectively, is calculated as follows:

$$d(x,y) = \begin{cases} \max(x,y) & \text{if } \min(x,y) = 0 \\ \min \begin{cases} d(x-1, y) + 1 \\ d(x, y-1) + 1 \\ d(x-1, y-1) + \mathbf{1}_{f_i(x) \neq f_j(y)} \end{cases} & \text{otherwise,} \end{cases} \tag{7}$$

where $\mathbf{1}_{condition}$ is the indicator function. The distance score is normalised over the length of floods.

5) Distance matrix filtering: The structure of alarm flood data renders clustering a complex task. Floods are of very different lengths and contain various subsets of unique alarms.

Two floods that do not have many alarms in common should not be assigned to the same cluster, even though they might accidentally have a high similarity, for example due to the fact they both contain annunciations of an alarm that is very common in the whole dataset. As suggested in [1] we preprocess distance matrices such that only pairs of floods that share a majority of alarms can belong to the same cluster. Concretely, we use secondary distance measure $\Omega \in \{J, T, F, L\}$ only if the primary measure J between f_i, f_j is lower than a given threshold (referred to as Jaccard threshold), otherwise we assigning as distance between f_i and f_j the upper bound of Ω between all pairs of floods of the dataset. We denote these stacked measures as JΩ and use Jaccard threshold 0.4 as suggested in [1], see Fig. 1. For example, JJ elevates Jaccard distances above 0.4 to the highest Jaccard distance found in pairwise comparison.

C. Clustering

Two algorithms were used to perform alarm flood clustering: agglomerative hierarchical and DBSCAN. Agglomerative hierarchical clustering gradually clusters observations until only one cluster containing all the points is obtained. At every step, the most similar pair of observations and or clusters are merged, one at a time. Cutting off at a specific number of clusters yields a clustering solution. On the other hand, DBSCAN [18] is a clustering algorithm that intrinsically deduces the most-fitting number of clusters. It is based on the concept of density, where points within a specified distance threshold ϵ to each other are considered to belong to a dense area—a cluster. Points that are distant from dense areas are considered outliers. We group outliers in a separate cluster.

Preliminary experiments showed, that it is a disadvantage to predefine the number of clusters, because the number of natural clusters in our data is expected to change with different distance measures. Therefore, to perform an unbiased comparison of distance measures, we use the DBSCAN algorithm which is not biased to a fixed number of clusters.

III. Validation approach

It is not possible to validate clustering results when full expert annotation is not available, and this is usually the case in an industrial setting. However, it is possible to systematically perform repeatable experiments and observe and evaluate the changes in the results that are induced by controlled modifications of the input.

To that end we propose here an approach for such a systematic evaluation. We generate three new test-sets of observations based on the original set of floods: (i) a set of "very similar" floods, (ii) a set of "somewhat similar" floods and (iii) a set of "very dissimilar" floods. Since the origin of the synthetic floods is known, assumptions can be made on how the synthetic floods should be clustered, depending on their similarity to the origin. This facilitates using measures suitable for supervised learning, e.g. RAND index.

Prior work [1] does not provide any formalised or quantitative way to assess clustering results, instead two-flood-clusters are visually compared using Dynamic Time Warping to align floods on common alarms. Moreover, in [1] it is not specified based on which criteria the number of clusters was chosen.

We here propose an approach for validating the behaviour of similarity measures, by quantifying whether a similarity measures provides the expected clustering result on a dataset that has been modified in a certain controlled way. Note that we do not try to validate the correctness of clustering solutions (for that we would require a gold standard).

A. Synthetic flood generation

Synthetic floods are generated based on an existing dataset. Each original flood is used to create one synthetic flood. We call the original flood "mother" of the synthetic flood and modified the mother flood in three ways to create the synthetic flood: (i) by addition of randomly chosen alarms, (ii) by removal of randomly chosen alarms and (iii) by transposing randomly chosen pairs of alarms. For simplicity, we always apply an equal amount, at least one, of all three modifications to each flood. The number of modifications is varied throughout the experiments to obtain different synthetic flood sets, ranging from very similar to dissimilar to the original floods. We represent the degree of modification as a percentage of the number of alarms in a flood that has been modified.

This way of modification of floods from real datasets is chosen according to our domain knowledge and experience with floods and their variation in the industrial setting.

B. Validation of the results

After adding synthetic floods to the original dataset, we apply the distance computation and clustering (see Fig. 1 and

Sec. II). We then use two measures to obtain a quantitative validation for comparing clustering solutions of the original and the synthetically enriched datasets.

1) Cluster Membership of Synthetic Floods: The first validation approach is the fraction of synthetic floods that is assigned to the same cluster as their mother flood. It is calculated as

$$M_0 = \frac{|\text{synthetic floods in same cluster as mother flood}|}{|\text{synthetic floods}|}. \quad (8)$$

This measure is calculated for each synthetic flood with respect to its mother flood, without considering (potentially random) similarities to other original floods.

2) Cluster Stability: We can consider the original flood clustering results to be the ground truth, or the "target", as in the supervised machine learning validation methods. This assumption is made only for the purpose of validation of the similarity measure behaviour. Each synthetic flood has a known mother flood, and is expected to be treated similarly by the clustering algorithm if it was generated with a low degree of modification; i.e., the synthetic flood is expected to have similar distance scores to other floods as its mother flood, and therefore to be clustered alike. Hence, the synthetic floods are given the same target as their mother floods. On the other hand, if the synthetic flood was generated using a high degree of modification it is expected to be treated differently by the clustering algorithm than its original flood.

Results of clustering original and synthetic floods together can be compared to that target solution. Adjusted RAND index is a well-known measure for quantifying partition agreement between clustering solutions, while disregarding the actual cluster number. Adjusted RAND index for two partitions $C_1 = \{c_0, c_1, ..., c_n\}$ and $C_2 = \{c_0, c_1, ..., c_n\}$ of items is calculated as

$$R = \frac{a+b}{a+b+c+d}, \quad (9)$$

where a is the number of pairs of items that are in the same cluster in C_1 and in C_2, b is the number of pairs of items that are in different clusters in C_1 and in C_2, c is the number of pairs of items that are in the same cluster in C_1 but in different clusters in C_2, and d is the number of pairs of items that are in different clusters in C_1 but in the same cluster in C_2.

IV. Empirical evaluation

A. Experimental design

Figure 1 presents the experimental design. The general procedure for alarm flood clustering consists of flood detection, calculating the pairwise distances and clustering using preferred algorithm. We propose a parallel clustering validation process. The dataset is extended by a set of synthetic alarm floods and the distance calculation and clustering is performed analogously. Addition of randomly generated alarm floods allows systematic validation of the results using measures described in section III-B .

The dataset used in this paper is a 25-day alarm log of a production plant from the manufacturing industry. In that period, there was a total of 15 thousand annunciations of 96 alarm types. Merging the chattering alarms reduced the alarm log to 11.5 thousand entries and within those, a total of 166 floods of various length was detected. We evaluate the results and analyse the behaviour of the distance measure as it changes in the structure of clusters induced by adding synthetic floods to the dataset. Here, addition of synthetic floods enlarges the set to 332 floods.

DBSCAN clustering algorithm is used in the experiments because it does not require specifying the number of clusters. Therefore, we can observe how do the synthetic floods change the structure of the data, for example by creating new inherent clusters.

A single experiment consists of generating three new sets of synthetic floods: a set with 10% of modifications ("very similar" floods), a set with 25% of modifications ("somewhat similar" floods) and finally a set with 50% of modifications ("not very similar" floods). The enlarged dataset is then clustered using all four distance measures and DBSCAN clustering algorithm. The solution is evaluated using adjusted RAND index and percentage of synthetic floods clustered consistently with their original floods—measures described in section III-B—as well as the number of clusters and the maximum cluster size. The results are compared to the original dataset and averaged over 100 experiments.

B. Results

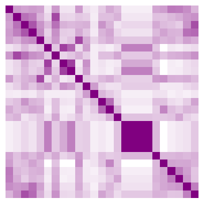
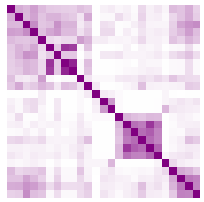
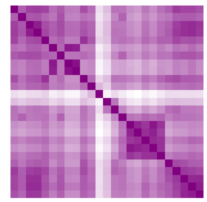
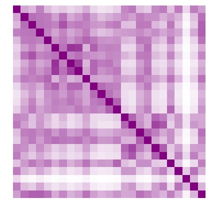
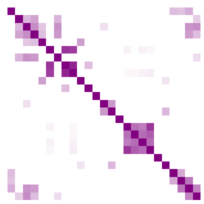
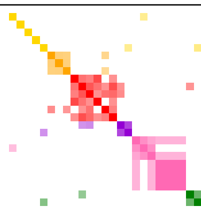
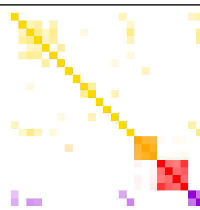
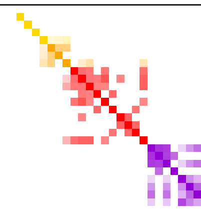
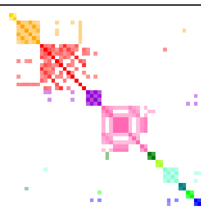
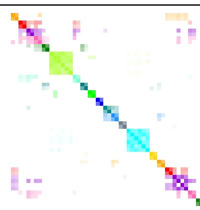
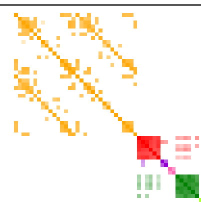
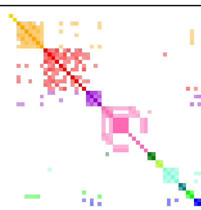
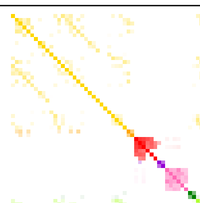
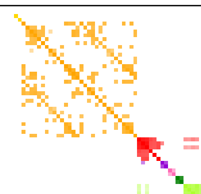
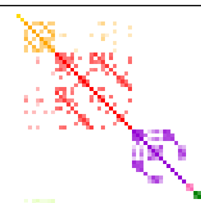
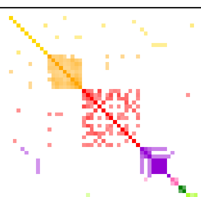
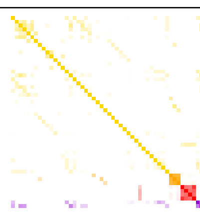
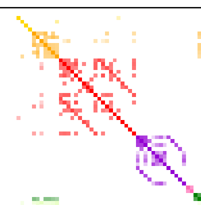
1) Clustering with synthetic floods on a demonstrative set of 25 floods: We visually demonstrate the empirical evaluation methodology on a reduced dataset of 25 floods. Table I presents the distance matrices obtained at every step of the experiment.

The original distance matrices (table Ia.) provides first insights into comparison of distance metrics. Jaccard distance identified four floods as exactly the same, but other distance metrics show that they are in fact not identical; albeit they are composed of the same alarms, they differ in the number and order of their annunciations. Preprocessing with Jaccard threshold (table Ib.) filters out many values in the distance matrices. Especially in the case of TF-IDF, this filters out many low distance values.

TF-IDF provides the weakest signal of all distance measures we compare, in particular with TF-IDF many floods that are distant in Jaccard are quite similar. The principle of TF-IDF is to put more weight on terms that occur more often in one flood and that occur less often in other floods. To obtain high distance, a pair of floods needs to contain a distinct set of alarms that has low frequency in other floods. In our dataset this rarely happens, and if it happens floods are usually short, so that the relative term frequency of rare alarms can contribute a larger value to the distance measure (TF-IDF is normalized over document size, i.e., over flood length).

Table Ic. presents the DBSCAN clustering results. Floods have been rearranged and coloured according to the cluster number. The top cluster represents the outliers whose distance

TABLE I: Distance matrices obtained at every stage of the experiments for measures J, F, T, and L (see II-B) for a reduced dataset of 25 floods. Pixels indicate pairwise similarity between floods which are arranged on X- and Y-axis in the same order: stronger colours show lower distance. Rows show (a) original flood distance matrices; (b) distance matrices of stacked metrics (JΩ); (c) clustering solutions using distances from (b), coloured according to the cluster number, with outliers in the top left cluster highlighted by a black border; (d) clustering solutions after adding a 10% modified synthetic flood for each flood; (e) clustering solutions after adding a 25% modified synthetic flood for each flood and (f) clustering solutions after adding a 50% modified synthetic flood for each flood.

	J	F	T	L
(a)				
(b)				
(c)				
(d)				
(e)				
(f)				

to the other floods was under the ϵ threshold and therefore couldn't be assigned to any of the clusters.

Table Id. presents the clustering results after introducing synthetic floods with 10% degree of modification. Introduction of synthetic floods does causes changes in the cluster structure, although the structure of original clusters is mostly retained. If the synthetic floods are generated as "very similar" to the original floods, many of the outlier floods are clustered together with their synthetic counterpart. Otherwise, they join the clusters of their corresponding original floods.

2) Clustering with synthetic floods on the full dataset: Fig. 2 depicts clustered JJ distance matrices for three degrees of modification. The order of the original floods is maintained, therefore it is possible to visually detect similar structures arising from the original dataset, and it is apparent that the majority of original clusters remains coupled despite the introduction of more and more diverse synthetic floods. An abundance of 2-flood clusters for the lowest degree of modification, similar as in the case of 25 flood example, is visible. Moreover, we observe that the number of outliers (top left cluster) increases with the degree of modification.

Table II presents four validation measures for each degree of modification and each distance measure.

Frequency of consecutive alarms distance measure has a noticeably lower RAND index score ($M0$) than the other distance measures, while having a high percentage of synthetic floods in the same cluster ($M1$). This is due to the fact that clustering with this distance measure puts the outlier floods in small 2-flood clusters together with their synthetic counterparts, yielding significantly more clusters of smaller size. When the degree of modification is high, clustering of frequency of consecutive alarms distance matrix received a high score for synthetic flood membership M_0 because many of the synthetic floods are considered outliers as their original floods and assigned to the outlier cluster. In general, the frequency of consecutive alarms measure causes clustering algorithm to consider many floods to be outliers. Whether or not that is reasonable must be judged by an expert plant operator.

The other distance measures display analogous behaviour according to the validation measures. They yield high scores for synthetic flood cluster membership and cluster stability when the degree of modification is low and the scores decrease as the degree of modification increases. The number of clusters and maximum cluster size are also uniform between the three measures.

In this validation process, synthetic floods of low degree of modification are expected to be clustered together with their "mother" floods, and the solution to yield high RAND and $M-0$ indices. Frequency of consecutive alarms measure performs significantly worse that the other measures in this aspect. When the degree of modification increases, we consider the floods to become rather dissimilar to their original floods, and therefore expect them not to be clustered with the original floods. At a 50% degree of modification level all the measures yield similar validation results, with an exception of synthetic flood cluster membership M_0 for frequency of consecutive alarms measure, caused be the high count of outliers in general.

V. Conclusion

In this paper, we introduce a novel approach for quantitatively and empirically evaluating similarity measures for alarm flood clustering. Our evaluation approach is focused on enriching a real dataset with synthetic floods of varying similarity to original floods. Modifications are based on domain knowledge and experience. This controlled dataset modification allows quantitative assessment of the influence of changes on the results. To the best of our knowledge, no formalizations for variations in alarm floods have been studied in literature, moreover there is no known systematic way for performing evaluation and choice of similarity measures used in alarm flood analysis. Our contributions provide such a methodology and can help researchers, fault detection and isolation systems developers, as well as industrial plant operators to assess the performance of alarm flood clustering and verify its results.

As a second contribution, we compare the clustering method and similarity measure of Ahmed et al. [1] and compare it with novel distance measures that we introduce as adaptations of existing commonly used methods, namely TF-IDF scores and Levenshtein distance. We show that the measure introduced in [1], produces very different results than our newly introduced measures, and results suggest that the measure of [1] is less favourable. Moreover, DBSCAN clustering appears to produce more meaningful results because of its adaptive choice of the number of clusters.

Future research into alarm flood clustering using presented methodology can be particularly useful in modifying and tweaking the distance measures and analysing the resulting changes in clustering behaviour. In particular, the results indicate that TF-IDF-based approach should be investigated more closely. The evaluation approach itself can be refined by adapted and fine-tuned to specific industrial settings.

Acknowledgement

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 678867.

References

[1] K. Ahmed, I. Izadi, T. Chen, D. Joe, and T. Burton, "Similarity analysis of industrial alarm flood data," in *IEEE Transactions on Automation Science and Engineering*, Apr 2013.

[2] J. Wang, F. Yang, T. Chen, and S. L. Shah, "An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems," *IEEE Transactions on Automation Science and Engineering*, vol. 13, pp. 1045–1061, April 2016.

[3] I. Izadi, S. L. Shah, D. S. Shook, and C. T., "An introduction to alarm analysis and design," in *IFAC SAFEPROCESS*, pp. 645–650, 2009.

[4] O. Niggemann and V. Lohweg, "On the diagnosis of cyber-physical production systems: State-of-the-art and research agenda," in *Proc. AAAI*, pp. 4119–4126, AAAI Press, 2015.

[5] T. Bergquist, J. Ahnlund, and J. E. Larsson, "Alarm reduction in industrial process control," in *Proc. IEEE Conference on Emerging Technologies and Factory Automation*, vol. 2, pp. 58–65 vol.2, 2003.

[6] B. Vogel-Heuser, D. Schütz, and J. Folmer, "Criteria-based alarm flood pattern recognition using historical data from automated production systems (aps)," *Mechatronics*, vol. 31, pp. 89 – 100, 2015.

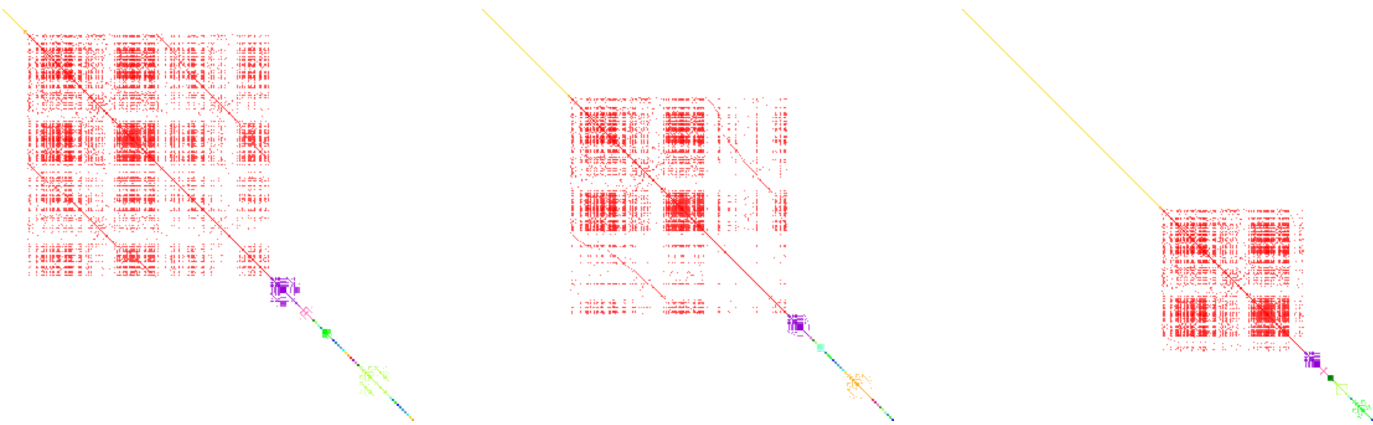
[7] S. Charbonnier, N. Bouchair, and P. Gayet, "Fault template extraction to assist operators during industrial alarm floods," *Engineering Applications of Artificial Intelligence*, vol. 50, pp. 32 – 44, 2016.

[8] F. Yang, S. Shah, D. Xiao, and T. Chen, "Improved correlation analysis and visualization of industrial alarm data," *ISA Transactions*, vol. 51, no. 4, pp. 499–506, 2012.

[9] J. C. Laberge, P. Bullemer, M. Tolsma, and D. V. C. Reising, "Addressing alarm flood situations in the process industries through alarm summary display design and alarm response strategy," *Intl. J. of Industrial Ergonomics*, vol. 44, no. 3, pp. 395–406, 2014.

TABLE II: Averaged values of validation measures from 100 experiments. R denotes cluster stability calculated by RAND index, M_0 denotes cluster membership of synthetic floods measure, n_c denotes the number of clusters in the solution, max denotes maximum cluster size, Frequency denotes distance based on frequency of consecutive alarms, TF-IDF denotes euclidean distance for TF-IDF flood representations.

	10% degree of modification				25% degree of modification				50% degree of modification			
	R	M_0	n_c	max	R	M_0	n_c	max	R	M_0	n_c	max
Jaccard	0.84	0.90	30.89	199.15	0.60	0.61	27.21	165.74	0.26	0.25	13.32	151.99
Frequency	0.47	0.90	73.77	130.58	0.28	0.51	19.15	200.29	0.27	0.43	8.34	235.63
TF-IDF	0.84	0.90	30.90	199.15	0.60	0.61	27.18	165.74	0.25	0.25	13.30	152.31
Levenshtein	0.84	0.90	32.67	197.10	0.59	0.62	27.23	164.12	0.24	0.27	12.58	163.10



(a) 10% degree of modification (b) 25% degree of modification (c) 50% degree of modification

Fig. 2: Clustered Jaccard distance matrices of 166-flood data set with synthetic floods generated with different values of the degree of modification.

[10] J. Wang, H. Li, J. Huang, and C. Su, "A data similarity based analysis to consequential alarms of industrial processes," *Journal of Loss Prevention in the Process Industries*, vol. 35, pp. 29 – 34, 2015.

[11] S. Charbonnier, N. Bouchair, and P. Gayet, "A weighted dissimilarity index to isolate faults during alarm floods," *Control Engineering Practice*, vol. 45, pp. 110 – 122, 2015.

[12] M. M. Deza and E. Deza, *Encyclopedia of Distances*. Springer, 2009.

[13] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, pp. 11–21, 1972.

[14] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals.," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[15] Instrumentation, Systems, and Automation Society, *ANSI/ISA-18.2-2009: Management of Alarm Systems for the Process Industries*, 2009.

[16] E. Naghoosi, I. Izadi, and T. Chen, "Estimation of alarm chattering," *Journal of Process Control*, vol. 21, no. 9, pp. 1243 – 1249, 2011.

[17] P. Jaccard, "Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines," *Bulletin de la Société Vaudoise des Sciences Naturelles*, vol. 37, pp. 241 – 272, 1901.

[18] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," pp. 226–231, AAAI Press, 1996.