ISSN 1870-4069

Revisión del estado del arte en técnicas de minería de flujos de datos

María Yesenia Zavaleta-Sánchez,
Edgard Iván Benítez-Guerrero

Universidad Veracruzana,
Facultad de Estadística e Informática,
México

{yzavaleta, edbenitez}@uv.mx

Resumen. Actualmente, con el avance de la tecnología y el uso de diferentes dispositivos electrónicos, en muchas aplicaciones se generan grandes cantidades de datos de manera continua a altas velocidades llamados flujos de datos, (data streams, en inglés), que requieren ser procesados en cortos lapsos de tiempo. Las técnicas de minería de flujos de datos (Data Stream Mining, DSM por sus siglas en inglés) permiten extraer conocimiento e identificar patrones ocultos en flujos de datos continuos. El objetivo principal de este trabajo es presentar una revisión del estado del arte en el área de minería de flujos de datos para identificar modelos, técnicas y herramientas para el procesamiento de flujos de datos. Se realizó un mapeo sistemático con 69 artículos recuperados de cuatro bases de datos electrónicas. Como resultado se identificó el uso de modelos de series de tiempo y modelo de ventanas deslizantes para representación y procesamiento, respectivamente. Clasificación, conteo de frecuencias y agrupación como las técnicas de DSM más utilizadas. Herramientas para su manejo como Apache Hadoop, Apache Spark y Esper. Finalmente, para trabajos futuros la detección de eventos usando DSM.

Palabras clave: Flujo de datos, minería de flujos de datos, mapeo sistemático de la literatura.

State of the Art Review of Data Stream Mining Techniques

Abstract. Nowadays, with the advancement of technology and the use of different electronic devices, in many applications large amounts of data are generated continuously at high speeds called data streams, which need to be processed in short time lapses. Data Stream Mining techniques (DSM) allow you to extract knowledge and identify hidden patterns in continuous data streams. The main objective of this work is to present a review of the state of the art in the DSM area to identify models, techniques and tools for data stream processing. A systematic literature review was carried out with 69 articles retrieved from four electronic databases. As a result, the use of time

series models and sliding window models for representation and processing, respectively, was identified. Classification, frequency counting and clustering as the most used DSM techniques. Tools for its management such as Apache Hadoop, Apache Spark and Esper. Finally, for future work the event detection using DSM techniques.

Keywords: Data stream, data stream mining, a systematic literature review.

1. Introducción

Actualmente, con el avance de la tecnología y el uso de diferentes dispositivos electrónicos, en muchas aplicaciones se generan grandes cantidades de datos de manera continua a altas velocidades llamados flujos de datos, (data streams, en inglés), que requieren ser procesados en cortos lapsos de tiempo. Los flujos de datos son secuencias infitas de datos que se generan continuamente en el tiempo. Ejemplos de estos son los que se generan en transacciones bancarias, redes sociales, sensores, correo electrónico, entre otros [2, 36].

Asimismo, [2, 9] definen un flujo como una secuencia ilimitada de tuplas de la forma (s, t) ordenada por t, donde s, es una tupla relacional y t es la marca de tiempo de la tupla. Más formalmente, un flujo de datos S es una secuencia masiva de datos $x^1, x^2, ..., x^N$, es decir, $S = \{x^i\}_{i=1}^{N}$, potencialmente ilimitada $(N \rightarrow \infty)$. Cada flujo de datos se describe mediante un vector de atributo n-dimensional $x^i = [x_j^i]_{j=1}^{n}$ que pertenece a un espacio Ω de atributos que puede ser continuo, categórico o mixto [81].

En los últimos años, se ha desarrollado un creciente interés en el estudio de los flujos. Algunos casos de aplicación incluyen datos generados por redes de sensores, datos meteorológicos, análisis del mercado de valores y monitoreo del tráfico de la red, entre otros. Estas aplicaciones involucran conjuntos de datos que son demasiado grandes por lo que generalmente se almacenan en dispositivos secundarios.

Extraer conocimiento útil de los flujos de datos es un desafío, DSM debe considerar las siguientes restricciones: Los flujos llegan contínuamente, es decir, no hay control sobre el orden en que se deben procesar, el tamaño de una secuencia puede ser ilimitado, después de haber sido procesados los flujos son descartados y posiblemente el proceso de generación de flujos no es estacionario, es decir, su distribución de probabilidad puede cambiar con el tiempo [35].

En general, cuando se pretende extraer información o reconocer patrones de comportamiento en grandes cantidades de datos se recurre al uso de técnicas de minería de datos. Por las características de los flujos de datos, las técnicas de minería de datos tradicionales no pueden aplicarse directamente a las secuencias de datos. Esto se debe a que la mayoría requiere múltiples escaneos de datos para extraer información, lo que no es realista para los flujos de datos que pueden cambiar con el tiempo.

Por lo que surge la necesidad de adaptar dichas técnicas para dar un mejor tratamiento a estas estructuras de datos. De ahí se deriva la motivación de esta investigación y la importancia del uso de técnicas de DSM cuyo objetivo es extraer conocimiento y patrones ocultos de flujos de datos continuos [74]. En la literatura, existen otros trabajos no recientes donde se ha abordado el tema de DSM.

Tabla 1. Resultados de búsqueda avanzada.

	ACM	**IEEE**	**SD**	**SL**	**Total**
Búsqueda inicial	111	252	453	244	1066
Primera selección	6	59	78	42	185
Segunda selección	5	22	55	6	88
Selección final	5	16	43	5	69

Por ejemplo, en el 2003 en el trabajo de [41] se hace énfasis en los sistemas para la gestión de flujos, modelos de datos y lenguajes para las consultas continuas.

En 2005, [34] presentaron los fundamentos teóricos para el análisis de flujos de datos. Por otra parte, en trabajos más recientes [68] abordan los métodos para identificar patrones frecuentes en flujos de datos, [74] presentan un survey sobre agrupación y clasificación, [38] hacen una revisión del estado del arte para agrupación y [59] elaboran una revisión sistemática de la literatura para comparar herramientas y tecnologías para el análisis de flujos de datos masivos.

Debido a lo anterior, en este trabajo se presentan los resultados de un mapeo sistemático de la literatura basado en el análisis de 69 artículos publicados en bases de datos electrónicas en el periodo del 2010 al 2019, con la finalidad de conocer aspectos relevantes del área como modelos de datos y de procesamiento de flujos de datos, técnicas de preprocesamiento de flujos de datos, técnicas de DSM, herramientas para su gestión y los temas de investigación pendientes por abordar.

El resto del artículo está organizado de la siguiente forma, en la sección 2 se describe el método de investigación usado para la costrucción del mapeo sistemático, en la sección 3 se presentan los resultados obtenidos para cada pregunta de investigación planteada; y finalmente, en la sección 4 se resumen las conclusiones del estudio.

2. Método de investigación

Para la construcción del mapeo sistemático se tomaron en consideración los lineamientos propuestos por [56]. Las categorías utilizadas en un estudio de mapeo se basan en información de las publicaciones (nombres de autores, afiliaciones de autores, fuente de publicación, tipo de publicación, fecha de publicación, etc.) y/o información sobre los métodos de investigación utilizados [57]. Los pasos esenciales del proceso de nuestro estudio fueron la definición de preguntas de investigación, la búsqueda sistematizada de trabajos relevantes en bases de datos electrónicas, la selección de trabajos, el análisis del contenido de los artículos y los resultados del mapeo.

2.1. Preguntas de investigación

- P1. ¿Cuáles son los modelos que comúnmente se utilizan con flujos de datos?
- P2. ¿Cuáles son las técnicas de preprocesamiento de flujos de datos?
- P3. ¿Cuáles son las técnicas de la minería de flujos de datos más utilizadas?
- P4. ¿Cuáles son las herramientas que se emplean con mayor frecuencia?
- P5. ¿Cuáles son las principales temáticas pendientes por abordar en el área?

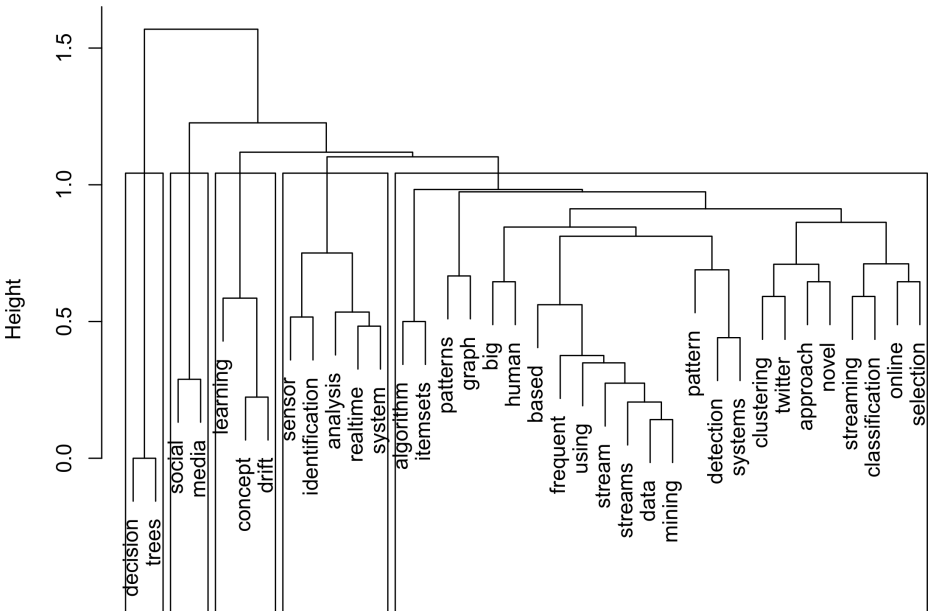

Fig. 1. Dendograma de temáticas abordadas en títulos de artículos.

2.2. Proceso de búsqueda

La estrategia de búsqueda incluyó el uso de cuatro bases de datos científicas de acceso Universitario: ACM Digital Library[1](ACM), IEEE Xplore[2](IEEE), ScienceDirect[3](SD) y SpringerLink[4](SL). En cada base se utilizó la cadena de búsqueda "DATA STREAM MINING". **Criterios de inclusión**: Journals en inglés publicados del 2010 al 2019. Para las bases de datos que lo permitieron, se realizó un refinamiento adicional limitando la búsqueda (ACM-PDF en el área de Computing Sciences in Colleges de enero del 2010 a agosto del 2019; SD–búsqueda en título, resumen y palabras clave).

Como resultado de la búsqueda inicial se encontraron 1066 artículos (111 ACM, 252 IEEE, 453 SD y 244 en SL). En la primera selección se revisó el título, resumen y palabras clave de los documentos, obteniendo como resultado 185 artículos y excluyendo un total de 881 artículos. En la segunda selección, se revisó la introducción y conclusiones de los 185 trabajos, eligiendo 88 artículos.

Finalmente, se realizó una revisión completa de los artículos para identificar si respondian al menos una de las preguntas de investigación con lo que se descartaron 19 artículos y se consideraron para el estudio 69 artículos (5 ACM, 16 IEEE, 43 SD y 5 en SL), ver Tabla 1.

[1] https://dl.acm.org/

[2] https://ieeexplore.ieee.org/Xplore/home.jsp

[3] https://www.sciencedirect.com/

[4] https://link.springer.com/

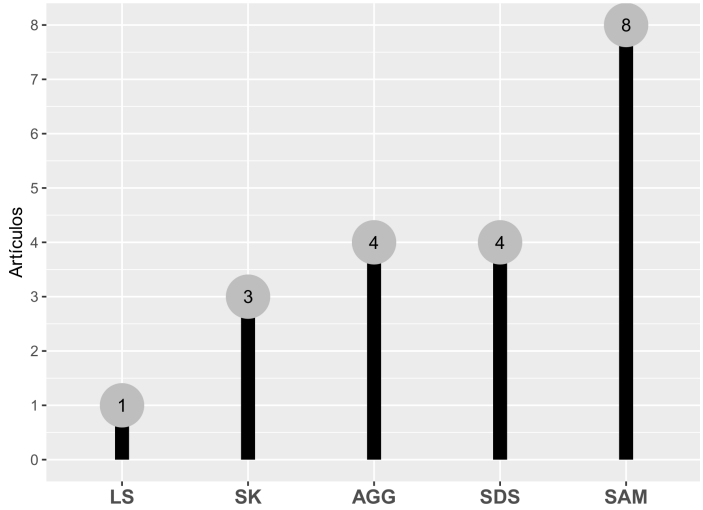

Fig. 2. Lollipop sobre técnicas de preprocesamiento de flujos.

Criterios de exclusión: Se excluyeron 997 artículos que, después de una revisión detallada, no respondían al menos una de las preguntas de investigación planteadas. De cada artículo se extrajo: Título, autores, año de publicación, palabras clave, modelos de datos y de procesamiento, técnicas de preprocesamiento, técnicas DSM, herramientas utilizadas para la minería de flujos y trabajos futuros.

Se construyó una base de datos con las variables de los 69 artículos. Para las variables cualitativas (modelos, técnicas de preprocesamiento y técnicas DSM) se realizaron tablas de frecuencias o gráficos de barras (lollipop) para representar la distribución de las categorías y clasificar los trabajos revisados.

Finalmente, para las variables textuales (título, herramientas y trabajos futuros) se aplicaron técnicas de minería de textos como dendogramas para agrupación jerárquica de palabras y un Trigrama para crear una red semántica y buscar asociaciones entre grupos de 3 palabras sobre trabajos futuros. Cabe resaltar que todos los análisis se realizaron con el Software R Project Versión 4.0.3[5]. En la Figura 1 se muestra un Dendograma donde se agrupan las temáticas principales que se abordan en los títulos de los artículos.

3. Resultados

P1. ¿Cuáles son los modelos que comúnmente se utilizan con flujos de datos? De acuerdo con los trabajos de [39, 72, 42], existen al menos cuatro formas en que un flujo de datos puede representar una señal subyacente $A : [1...N] \rightarrow R$: Modelo de serie temporal, modelo de caja registradora, modelo de torniquete y modelo agregado. En el modelo de serie temporal (time series model, en inglés) cada observación proporciona directamente el valor de la señal subyacente y aparece en orden creciente.

[5] https://www.r-project.org/

Tabla 2. Resumen de técnicas de minería de flujos de datos.

Técnica	Artículos	Autores
Classification	34	[64], [91], [18], [12], [6], [8], [30], [73], [71], [93], [70], [62], [17], [23], [52], [85], [32], [14], [31], [49], [5], [1], [60], [92], [65], [7], [19], [78], [21],[79], [90], [46], [63], [76]
Frequency counting	15	[94], [75], [50], [61], [84], [28], [20], [24], [83], [16], [77], [29], [44], [27], [22]
Clustering	13	[86], [95], [43], [88], [87], [37], [26], [13], [15], [69], [82], [55], [33]
Time series analysis	6	[89], [86], [66], [45], [47], [54]

En el modelo de caja registradora (cash register model, en inglés) cada flujo representa un incremento positivo al valor anterior de la señal para obtener el nuevo valor. El modelo de torniquete (turnstile model, en inglés) es una generalización del modelo de caja registradora al permitir que los valores que se incrementan del rango parcial sean negativos.

Finalmente, en el modelo agregado (aggregate model, en inglés) cada elemento del flujo contiene un rago de valores para un valor particular en el dominio de A.

Según la investigación de [96, 51], hay tres modelos de procesamiento de flujos de datos: Landmark, Damped y Sliding Windows. El modelo landmark extrae todos los conjuntos de elementos frecuentes en todo el historial de flujos de datos desde un punto de tiempo específico llamado landmark hasta el presente.

El modelo damped o time-fading model, extrae conjuntos de elementos frecuentes en los flujos de datos en los que cada transacción tiene un peso y este peso disminuye con la edad. El modelo de ventanas deslizantes (sliding windows, en inglés) encuentra y mantiene conjuntos de elementos frecuentes en ventanas deslizantes cuyo tamaño está predefinido y se puede decidir de acuerdo con las aplicaciones y los recursos del sistema.

Elegir qué tipo de modelo se debe usar depende en gran medida de las necesidades de la aplicación. En los trabajos revisados, el modelo de flujo de datos más utilizado fue el modelo de series de tiempo [89, 86, 66, 45, 47, 54]. Por su parte, los modelos de procesamiento de datos identificados fueron dos: el modelo de ventana deslizante [65, 29, 22]; y el modelo de ventana landmark [28].

P2. ¿Cuáles son las técnicas de preprocesamiento de flujos de datos? En el trabajo de [34], se describe a las técnicas de preprocesamiento como enfoques generales para procesar flujos de datos antes de aplicar DSM. El muestreo (del inglés, sampling, SAM) es el proceso de elección probabilística de un elemento o dato para ser procesado o no [25]. Load shedding (LS), se refiere al proceso de descartar una secuencia de flujos de datos en consultas contínuas.

Sin embargo, es difícil de usar con algoritmos de minería de flujos porque elimina fragmentos de flujos de datos que podrían usarse en la generación de modelos [11]. Sketching (SK) permite proyectar aleatoriamente un subconjunto de las características [10]. Synopsis data structures (SDS), consiste en aplicar técnicas de resumen a la secuencia entrante para su posterior análisis. El análisis de wavelet, histogramas, cuantiles y momentos de frecuencia son algunos ejemplos [10, 40].

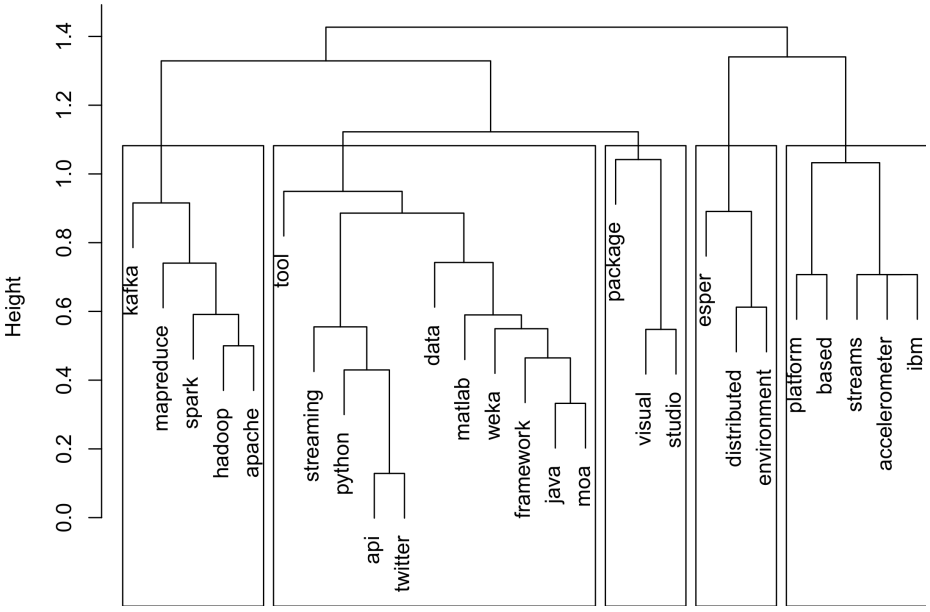

Fig. 3. Dendograma sobre herramientas para minería de flujos de datos.

Por último, la agregación (en inglés, aggregation, AGG), se obtiene al calcular medidas estadísticas como las media y la varianza que resumen el flujo entrante [4, 3]. En nuestro estudio se identificó SAM en 8 artículos: [91, 95, 80, 23, 85, 32, 83, 63] seguida de SDS [64, 94, 88, 78] y AGG [87, 92, 19, 46] en 4 artículos cada una; SK [58, 73, 93] en 3 artículos y LS en 1 artículo [24], ver Figura 2.

P3. ¿Cuáles son las técnicas de la minería de flujos de datos más utilizadas? La agrupación (clustering, en inglés), permite agrupar datos con un comportamiento similar. Se puede considerar como una partición o segmentación de elementos en grupos que pueden o no ser disjuntos.

En muchos casos, la respuesta a un problema de agrupación no es única, es decir, se pueden encontrar muchas respuestas, e interpretar el significado práctico de cada agrupación puede ser difícil [4]. La clasificación, (classification, en inglés), asigna datos en grupos predefinidos (clases). Su diferencia con clustering es que, en esta, el número de grupos está predeterminado y fijo [53].

Por otra parte, la técnica de conteo de frecuencias, (frequency counting, en inglés), permite el conteo de frecuencias para identificar conjuntos de elementos frecuentes. Sin embargo, aunque la minería de conjuntos de elementos frecuentes se ha estudiado ampliamente en la minería de datos, extenderla a los flujos de datos es un desafío, especialmente para los flujos con distribuciones no estáticas [51].

Por último, la técnica de análisis de series de tiempo, (time series analysis, en inglés), modela flujos de datos que contienen solo valores numéricos como series de tiempo. Las tareas de minería de flujos en series temporales se pueden clasificar brevemente en dos tipos: detección de patrones y análisis de tendencias [48, 67].

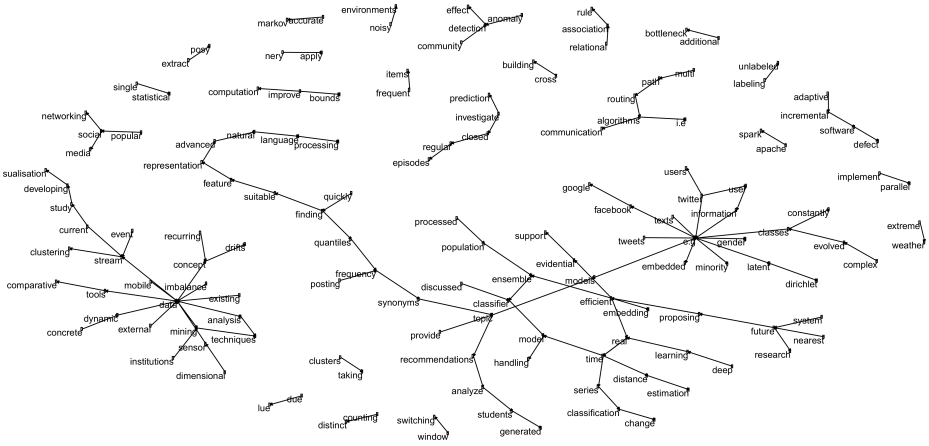

Fig. 4. Trigrama de trabajos futuros en DSM.

Derivado de nuestro análisis se encontró que la clasificación es la técnica de DSM que se emplea con mayor frecuencia. Como se observa en la Tabla 2, la distribución del uso de las técnicas de DSM fue: 34 artículos de clasificación, 15 de conteo de frecuencias, 13 de agrupación y 6 artículos sobre análisis de series de tiempo.

P4. ¿Cuáles son las herramientas que se emplean con mayor frecuencia? Los resultados nos muestran el uso de Apache Hadoop, Apache Spark, Mapreduce y Kafka. Así como tambien las Api´s de Twitter y Python para la generación de flujos. Otras herramientas relevantes que figuran en los trabajos son los frameworks de MOA y Java, Weka, Esper y MATLAB. Por otra parte, tambien se resalta el uso de una plataforma de IBM que no es opensource pero es una de las más completas para el procesamiento de flujos (ver Figura 3).

P5. ¿Cuáles son las principales temáticas pendientes por abordar en el área? En la Figura 4 se muestra un Tri-grama para identificar agrupaciones de tres términos más asociados a Data Stream Mining.

En el lado izquierdo se muestra una de las ramificaciones principales donde se observa la agrupación de tres términos Clustering-Stream-Event, es decir la agrupación de flujos de datos para la detección de eventos es un área de la minería de flujos que actualmente esta tomando ventaja en la investigación pues se requiere realizar análisis considerando la variación en la distribución de los flujos (Concept drift) y su efecto en la conformación de los clusters que también varían con la llegada de nuevos flujos.

También se requiere realizar análisis comparando diferentes herramientas que permitan el desarollo de la visualización de las agrupaciones.

Por otra parte, en el lado derecho de la red se observa una ramificación más grande que conecta varios temas de interés con el uso de técnicas de series de tiempo y clasificación en la minería de flujos: Técnicas de procesamiento de lenguaje natural (PLN), aprendizaje profundo (deep learning) y métodos de ensamble para la clasificación de flujos, evolución de clases complejas, el uso de series de tiempo para clasificación y su aplicación en flujos de textos que se generan en redes sociales como Facebook, Twitter o consultas en la Web como Google.

4. Conclusiones

Recientemente, se cuenta con aplicaciones en las que se generan flujos de datos a tasas muy altas con variaciones en el tiempo y posiblemente impredecibles.

La señal emitida por la entrada de los flujos de datos se puede modelar de diferentes formas siendo mas representativo el modelo de ventana deslizante.

Asímismo, aunque existen diferentes enfoques para realizar un preprocesamiento de los flujos, las técnicas mas utilizadas son el muestreo, sinopsis y agregación. Por otra parte, para realizar la extracción de conocimiento e información ocultos en flujos, se necesita el empleo de técnicas de DSM que permiten realizar tareas específicas como el agrupamiento, clasificación, conteo de frecuencias y análisis de series de tiempo, siendo de mayor uso la técnica de clasificación. Cabe resaltar que existen diferentes herramientas que permiten las generación de flujos y su procesamiento como son; Apache Hadoop, Apache Spark y MOA.

Finalmente, el análisis de flujos de datos masivos, aplicando técnicas de DSM aun tiene varios desafíos que resolver como es el caso de la agrupación de flujos para la detección de eventos, aplicaciones relacionadas con PLN, aprendizaje profundo, el uso de series de tiempo para clasificación y su aplicación en flujos de textos que se generan en redes sociales, entre otros.

Agradecimientos. Al Consejo Nacional de Ciencia y Tecnología (CONACYT) de México en el marco del proyecto de Cátedras "Infraestructura para Agilizar el Desarrollo de Sistemas Centrados en el Usuario" (Ref. 3053). También por la beca de Doctorado número 743385 del primer autor, así como a la Universidad Veracruzana.

Referencias

1. Adeniyi, D. A., Wei, Z., Yongquan, Y.: Automated web usage data mining and recommendation system using k-nearest neighbor (knn) classification method. Applied Computing and Informatics, vol. 12, no. 1, pp. 90–108 (2016) doi: 10.1016/j.aci.2014.10.001
2. Aggarwal, C. C.: Data streams: models and algorithms. vol. 31. Springer Science & Business Media (2007) doi: 10.1007/978-0-387-47534-9
3. Aggarwal, C. C., Han, J., Wang, J., Yu, P. S.: A framework for projected clustering of high dimensional data streams. In: Proceedings of the Thirtieth international conference on Very large data bases, vol. 30, pp. 852–863 (2004)
4. Aggarwal, C. C., Philip, S. Y., Han, J., Wang, J.: A framework for clustering evolving data streams. In: Proceedings of the 2003 Very Large Data Base Endowment Inc. Conference, pp. 81–92 (2003) doi: 10.1016/b978-012722442-8/50016-1
5. Ait-Alla, A., Lütjen, M., Lewandowski, M., Freitag, M., Thoben, K. D.: Real-time fault detection for advanced maintenance of sustainable technical systems. Procedia CIRP, vol. 41, pp. 295–300 (2016) doi: 10.1016/j.procir.2016.01.015
6. Akbar, A., Kousiouris, G., Pervaiz, H., Sancho, J., Ta-Shma, P., Carrez, F., Moessner, K.: Real-time probabilistic data fusion for large-scale iot applications. IEEE Access, vol. 6, pp. 10015–10027 (2018) doi: 10.1109/access.2018.2804623
7. Amphawan, K., Soulas, J., Lenca, P.: Mining top-k regular episodes from sensor streams. Procedia Computer Science, vol. 69, pp. 76–85 (2015) doi: 10.1016/j.procs.2015.10.008

8. Anupama, N., Jena, S.: A novel approach using incremental oversampling for data stream mining. Evolving Systems, vol. 10, no. 3, pp. 351–362 (2019) doi: 10.1007/s12530-018-9249-5
9. Arasu, A., Babu, S., Widom, J.: The CQL continuous query language: Semantic foundations and query execution. The VLDB Journal, vol. 15, no. 2, pp. 121–142 (2006) doi: 10.1007/s00778-004-0147-z
10. Babcock, B., Babu, S., Datar, M., Motwani, R., Widom, J.: Models and issues in data stream systems. In: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, pp. 1–16 (2002) doi: 10.1145/543613.543615
11. Babcock, B., Datar, M., Motwani, R.: Load shedding techniques for data stream systems. In: Proceedings of the 2003 Workshop on Management and Processing of Data Streams, vol. 577 (2003)
12. Benjelloun, F. Z., Oussous, A., Bennani, A., Belfkih, S., Lahcen, A. A.: Improving outliers detection in data streams using LiCS and voting. Journal of King Saud University-Computer and Information Sciences, vol. 33, no. 10 (2021) doi: 10.1016/j.jksuci.2019.08.003
13. Bhagat, A., Kshirsagar, N., Khodke, P., Dongre, K., Ali, S.: Penalty parameter selection for hierarchical data stream clustering. Procedia Computer Science, vol. 79, pp. 24–31 (2016) doi: 10.1016/j.procs.2016.03.005
14. Bodyanskiy, Y. V., Tyshchenko, O. K., Kopaliani, D. S.: Adaptive learning of an evolving cascade neo-fuzzy system in data stream mining tasks. Evolving Systems, vol. 7, no. 2, pp. 107–116 (2016) doi: 10.1007/s12530-016-9149-5
15. Bones, C. C., Romani, L. A., de Sousa, E. P.: Improving multivariate data streams clustering. Procedia Computer Science, vol. 80, pp. 461–471 (2016), doi: 10.1016/j.procs.2016.05.325
16. Braun, P., Cameron, J. J., Cuzzocrea, A., Jiang, F., Leung, C. K.: Effectively and efficiently mining frequent patterns from dense graph streams on disk. Procedia Computer Science, vol. 35, pp. 338–347 (2014) doi: 10.1016/j.procs.2014.08.114
17. Cerquitelli, T.: Predicting large scale fine grain energy consumption. Energy Procedia, vol. 111, pp. 1079–1088 (2017) doi: 10.1016/j.egypro.2017.03.271
18. Chao, S. C., Lin, K. C. J., Chen, M. S.: Flow classification for software-defined data centers using stream mining. IEEE Transactions on Services Computing, vol. 12, no. 1, pp. 105–116 (2016) doi: 10.1109/tsc.2016.2597846
19. Chen, X., Vorvoreanu, M., Madhavan, K.: Mining social media data for understanding students' learning experiences. IEEE Transactions on learning technologies, vol. 7, no. 3, pp. 246–259 (2014) doi: 10.1109/TLT.2013.2296520
20. Cuzzocrea, A., Han, Z., Jiang, F., Leung, C. K., Zhang, H.: Edge-based mining of frequent subgraphs from graph streams. Procedia Computer Science, vol. 60, pp. 573–582 (2015) doi: 10.1016/j.procs.2015.08.184
21. Czarnowski, I., Jedrzejowicz, P.: Ensemble classifier for mining data streams. Procedia Computer Science, vol. 35, pp. 397–406 (2014) doi: 10.1016/j.procs.2014.08.120
22. Dai, C., Chen, L.: An algorithm for mining frequent closed itemsets in data stream. Physics Procedia, vol. 24, pp. 1722–1728 (2012) doi: 10.1016/j.phpro.2012.02.254
23. Dao, M. S., Nguyen Gia, T. A., Mai, V. C.: Daily human activities recognition using heterogeneous sensors from smartphones. Procedia computer science, vol. 111, pp. 323–328 (2017) doi: 10.1016/j.procs.2017.06.030
24. Desai, D., Joshi, A.: A deviant load shedding system for data stream mining. Procedia Computer Science, vol. 45, pp. 118–126 (2015) doi: 10.1016/j.procs.2015.03.103
25. Domingos, P., Hulten, G.: A general method for scaling up machine learning algorithms and its application to clustering. In: Proceedings of the Eighteenth International Conference on Machine Learning, vol. 1, pp. 106–113 (2001)

26. Drosou, A., Kalamaras, I., Papadopoulos, S., Tzovaras, D.: An enhanced graph analytics platform (gap) providing insight in big network data. Journal of Innovation in Digital Ecosystems, vol. 3, no. 2, pp. 83–97 (2016) doi: 10.1016/j.jides.2016.10.005
27. Erra, U., Frola, B.: Frequent items mining acceleration exploiting fast parallel sorting on the GPU. Procedia Computer Science, vol. 9, pp. 86–95 (2012) doi: 10.1016/j.procs.2012.04.010
28. Farhat, A., Gouider, M. S., Said, L. B.: New algorithm for frequent itemsets mining from evidential data streams. In: Proceedings of the 20th International Conference KES, pp. 645–653 (2016) doi: 10.1016/j.procs.2016.08.246
29. Farzanyar, Z., Kangavari, M., Cercone, N.: Max-FISM: Mining (recently) maximal frequent itemsets over data streams using the sliding window model. Computers & Mathematics with Applications, vol. 64, no. 6, pp. 1706–1718 (2012)
30. Fong, S., Li, J., Song, W., Tian, Y., Wong, R. K., Dey, N.: Predicting unusual energy consumption events from smart home sensor network by data stream mining with misclassified recall. Journal of Ambient Intelligence and Humanized Computing, vol. 9, no. 4, pp. 1197–1221 (2018) doi: 10.1007/s12652-018-0685-7
31. Fong, S., Liu, K., Cho, K., Wong, R., Mohammed, S., Fiaidhi, J.: Improvised methods for tackling big data stream mining challenges: Case study of human activity recognition. The Journal of Supercomputing, vol. 72, no. 10, pp. 3927–3959 (2016) doi: 10.1007/s11227-016-1639-5
32. Fong, S., Wong, R., Vasilakos, A. V.: Accelerated PSO swarm search feature selection for data stream mining big data. IEEE Transactions on Services Computing, vol. 9, no. 1, pp. 33–45 (2015) doi: 10.1109/tsc.2015.2439695
33. Gaber, M. M., Krishnaswamy, S., Gillick, B., AlTaiar, H., Nicoloudis, N., Liono, J., Zaslavsky, A.: Interactive self-adaptive clutter-aware visualisation for mobile data mining. Journal of Computer and System Sciences, vol. 79, no. 3, pp. 369–382 (2013) doi: 10.1016/j.jcss.2012.09.009
34. Gaber, M. M., Zaslavsky, A., Krishnaswamy, S.: Mining data streams: A review. ACM Sigmod Record, vol. 34, no. 2, pp. 18–26 (2005) doi: 10.1145/1083784.1083789
35. Gama, J.: Knowledge discovery from data streams, vol. 12. IOS Press (2010), doi: 10.3233/ida-2008-123
36. Gama, J., Gaber, M. M.: Learning from data streams. Processing techniques in sensor networks. Springer (2007) doi: 10.1007/3-540-73679-4
37. Gao, T., Li, A., Meng, F.: Research on data stream clustering based on FCM algorithm 1. Procedia Computer Science, vol. 122, pp. 595–602 (2017) doi: 10.1016/j.procs.2017.11.411
38. Ghesmoune, M., Lebbah, M., Azzag, H.: State-of-the-art on clustering data streams. Big Data Analytics, vol. 1, no. 1, pp. 13 (2016) doi: 10.1186/s41044-016-0011-3
39. Gilbert, A. C., Kotidis, Y., Muthukrishnan, S., Strauss, M.: Surfing wavelets on streams: One-pass summaries for approximate aggregate queries. In: Proceedings of the 27th International Conference on Very Large Data Bases, vol. 1, pp. 79–88 (2001)
40. Gilbert, A. C., Kotidis, Y., Muthukrishnan, S., Strauss, M. J.: One-pass wavelet decompositions of data streams. IEEE Transactions on knowledge and data engineering, vol. 15, no. 3, pp. 541–554 (2003) doi: 10.1109/tkde.2003.1198389
41. Golab, L., Özsu, M. T.: Issues in data stream management. ACM Sigmod Record, vol. 32, no. 2, pp. 5–14 (2003) doi: 10.1145/776985.776986
42. Golab, L., Ozsu, M. T.: Data stream management. Morgan & Claypool Publishers (2010)
43. Groß-Klußmann, A., König, S., Ebner, M.: Buzzwords build momentum: Global financial twitter sentiment and the aggregate stock market. Expert Systems with Applications, vol. 136, pp. 171–186 (2019) doi: 10.1016/j.eswa.2019.06.027
44. Guo, J., Zhang, P., JianlongTan, Guo, L.: Mining hot topics from twitter streams. Procedia Computer Science, vol. 9, pp. 2008–2011 (2012) doi: 10.1016/j.procs.2012.04.224

45. Hassan, M. H., Tizghadam, A., Leon-Garcia, A.: Spatio-temporal anomaly detection in intelligent transportation systems. Procedia Computer Science, vol. 151, pp. 852–857 (2019) doi: 10.1016/j.procs.2019.04.117
46. Hemalatha, C. S., Vaidehi, V.: Frequent bit pattern mining over tri-axial accelerometer data streams for recognizing human activities and detecting fall. Procedia Computer Science, vol. 19, pp. 56–63 (2013) doi: 10.1016/j.procs.2013.06.013
47. Hu, Y., Jiang, Z., Zhan, P., Zhang, Q., Ding, Y., Li, X.: A novel multi-resolution representation for streaming time series. Procedia Computer Science, vol. 129, pp. 178–184 (2018) doi: 10.1016/j.procs.2018.03.069
48. Indyk, P., Koudas, N., Muthukrishnan, S.: Identifying representative trends in massive time series data sets using sketches. In: Proceedings of the 26th International Conference on Very Large Data Bases, pp. 363–372 (2000)
49. Jankowski, D., Jackowski, K., Cyganek, B.: Learning decision trees from data streams with concept drift. Procedia Computer Science, vol. 80, pp. 1682–1691 (2016) doi: 10.1016/j.procs.2016.05.508
50. Jaysawal, B. P., Huang, J.-W.: PSP-AMS: Progressive mining of sequential patterns across multiple streams. ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 13, no. 1, pp. 1–23 (2018) doi: 10.1145/3281632
51. Jiang, N., Gruenwald, L.: Research issues in data stream association rule mining. ACM SIGMOD Record, vol. 35, no. 1, pp. 14–19 (2006) doi: 10.1145/1121995.1121998
52. Kazanskiy, N., Protsenko, V., Serafimovich, P.: Performance analysis of real-time face detection system based on stream data mining frameworks. Procedia Engineering, vol. 201, pp. 806–816 (2017) doi: 10.1016/j.proeng.2017.09.602
53. Khan, M., Ding, Q., Perrizo, W.: k-nearest neighbor classification on spatial data streams using p-trees. In: Advances in Knowledge Discovery and Data Mining, pp. 517–528 (2002) doi: 10.1007/3-540-47887-6_51
54. Khodabakhsh, A., Arí, I., Bakír, M., Ercan, A. O.: Multivariate sensor data analysis for oil refineries and multi-mode identification of system behavior in real-time. IEEE Access, vol. 6, pp. 64389–64405 (2018), doi: 10.1109/access.2018.2877097
55. Kianfar, J., Edara, P.: A data mining approach to creating fundamental traffic flow diagram. Procedia-Social and Behavioral Sciences, vol. 104, no. 1, pp. 430–439 (2013) doi: 10.1016/j.sbspro.2013.11.136
56. Kitchenham, B.: Procedures for performing systematic reviews. Keele, UK, Keele University, vol. 33, no. 2004, pp. 1–26 (2004)
57. Kitchenham, B. A., Budgen, D., Brereton, O. P.: Using mapping studies as the basis for further research–a participant-observer case study. Information and Software Technology, vol. 53, no. 6, pp. 638–651 (2011) doi: 10.1016/j.infsof.2010.12.011
58. Ko, J. M., Hong, S. R., Choi, J. Y., Kim, C. O.: Wafer-to-wafer process fault detection using data stream mining techniques. International Journal of Precision Engineering and Manufacturing, vol. 14, no. 1, pp. 103–113 (2013), doi: 10.1007/s12541-013-0015-0
59. Kolajo, T., Daramola, O., Adebiyi, A.: Big data stream analysis: A systematic literature review. Journal of Big Data, vol. 6, no. 1, pp. 47 (2019) doi: 10.1186/s40537-019-0210-7
60. Krawczyk, B.: GPU-accelerated extreme learning machines for imbalanced data streams with concept drift. Procedia Computer Science, vol. 80, pp. 1692–1701 (2016) doi: 10.1016/j.procs.2016.05.509
61. Kusumakumari, V., Sherigar, D., Chandran, R., Patil, N.: Frequent pattern mining on stream data using hadoop cantree-gtree. Procedia Computer Science, vol. 115, pp. 266–273 (2017) doi: 10.1016/j.procs.2017.09.134
62. Le, T., Stahl, F., Gaber, M. M., Gomes, J. B., Di Fatta, G.: On expressiveness and uncertainty awareness in rule-based classification for data streams. Neurocomputing, vol. 265, pp. 127–141 (2017) doi: 10.1016/j.neucom.2017.05.081

63. Li, P., Wu, X., Hu, X.: Mining recurring concept drifts with limited labeled streaming data. In: Proceedings of 2nd Asian conference on machine learning, vol. 3, pp. 1–32 (2012) doi: 10.1145/2089094.2089105
64. Liang, C., Li, M., Liu, B.: Online computing quantile summaries over uncertain data streams. IEEE Access, vol. 7, pp. 10916–10926 (2019) doi: 10.1109/access.2019.2891550
65. Lifna, C., Vijayalakshmi, M.: Identifying concept-drift in twitter streams. Procedia Computer Science, vol. 45, pp. 86–94 (2015)
66. Lim, S., Tucker, C. S.: Mining twitter data for causal links between tweets and real-world outcomes. Expert Systems with Applications: X, vol. 3 (2019) doi: 10.1016/j.eswax.2019.100007
67. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery, pp. 2–11 (2003) doi: 10.1145/882082.882086
68. Liu, H., Lin, Y., Han, J.: Methods for mining frequent items in data streams: An overview. Knowledge and information systems, vol. 26, no. 1, pp. 1–30 (2009) doi: 10.1007/s10115-009-0267-2
69. Liu, Y., Ma, P., Cui, H.: Design and development of fpga-based high-performance radar data stream mining system. Procedia Computer Science, vol. 55, pp. 876–885 (2015) doi: 10.1016/j.procs.2015.07.147
70. Miholca, D. L., Czibula, G., Crivei, L. M.: A new incremental relational association rules mining approach. Procedia Computer Science, vol. 126, pp. 126–135 (2018) doi: 10.1016/j.procs.2018.07.216
71. Molano-Pulido, J., Jiménez-Guarín, C.: SEAbIRD: Adaptable daily living activity identification from sensor data streams. Procedia Computer Science, vol. 130, pp. 939–946 (2018) doi: 10.1016/j.procs.2018.04.093
72. Muthukrishnan, S.: Data streams: Algorithms and applications. Foundations and Trends® in Theoretical Computer Science, vol. 1, no. 2, pp. 117–236 (2005) doi: 10.1561/0400000002
73. Nalavade, J. E., Murugan, T. S.: HRNeuro-fuzzy: Adapting neuro-fuzzy classifier for recurring concept drift of evolving data streams using rough set theory and holoentropy. Journal of King Saud University-Computer and Information Sciences, vol. 30, no. 4, pp. 498–509 (2018) doi: 10.1016/j.jksuci.2016.11.005
74. Nguyen, H. L., Woon, Y. K., Ng, W. K.: A survey on data stream clustering and classification. Knowledge and Information Systems, vol. 45, no. 3, pp. 535–569 (2015) doi: 10.1007/s10115-014-0808-1
75. Peng, M., Zhu, J., Wang, H., Li, X., Zhang, Y., Zhang, X., Tian, G.: Mining event-oriented topics in microblog stream with unsupervised multi-view hierarchical embedding. ACM Transactions on Knowledge Discovery from Data, vol. 12, no. 3, pp. 1–26 (2018)
76. Qu, Z. Y., Zhang, L., Ding, G. L.: A data stream mining algorithm used in power grid disturbance identification. Physics Procedia, vol. 24, pp. 966–970 (2012) doi: 10.1016/j.phpro.2012.02.145
77. Rashid, M. M., Gondal, I., Kamruzzaman, J.: ACSP-tree: A tree structure for mining behavioral patterns from wireless sensor networks. In: Proceedings of the 38th Annual IEEE Conference on Local Computer Networks, pp. 691–694 (2013) doi: 10.1109/lcn.2013.6761312
78. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: Decision trees for mining data streams based on the gaussian approximation. IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 1, pp. 108–119 (2013) doi: 10.1109/tkde.2013.34
79. Rutkowski, L., Pietruczuk, L., Duda, P., Jaworski, M.: Decision trees for mining data streams based on the McDiarmid's bound. IEEE Transactions on Knowledge and Data Engineering, vol. 25, no. 6, pp. 1272–1279 (2012) doi: 10.1109/tkde.2012.66

80. Shi, L. L., Liu, L., Wu, Y., Jiang, L., Hardy, J.: Event detection and user interest discovering in social media data streams. IEEE Access, vol. 5, pp. 20953–20964 (2017) doi: 10.1109/access.2017.2675839
81. Silva, J. A., Faria, E. R., Barros, R. C., Hruschka, E. R., De Carvalho, A. C., Gama, J.: Data stream clustering: A survey. ACM Computing Surveys, vol. 46, no. 1, pp. 1–31 (2013)
82. Spezzano, G., Vinci, A.: Pattern detection in cyber-physical systems. Procedia Computer Science, vol. 52, pp. 1016–1021 (2015) doi: 10.1016/j.procs.2015.05.096
83. Srimani, P., Malini, M. P.: Frequent item set mining using INC_MINE in massive online analysis frame work. Procedia Computer Science, vol. 45, pp. 133–142 (2015) doi: 10.1016/j.procs.2015.03.105
84. Subbian, K., Aggarwal, C., Srivastava, J.: Mining influencers using information flows in social streams. ACM Transactions on Knowledge Discovery from Data, vol. 10, no. 3, pp. 1–28 (2016) doi: 10.1145/2815625
85. Sun, Y., Tang, K., Minku, L. L., Wang, S., Yao, X.: Online ensemble learning of data streams with gradually evolved classes. IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 6, pp. 1532–1545 (2016) doi: 10.1109/tkde.2016.2526675
86. Tajalizadeh, H., Boostani, R.: A novel stream clustering framework for spam detection in twitter. IEEE Transactions on Computational Social Systems, vol. 6, no. 3, pp. 525–534 (2019) doi: 10.1109/TCSS.2019.2910818
87. Tennant, M., Stahl, F., Rana, O., Gomes, J. B.: Scalable real-time classification of data streams with concept drift. Future Generation Computer Systems, vol. 75, pp. 187–199 (2017) doi: 10.1016/j.future.2017.03.026
88. Vadivukarassi, M., Puviarasan, N., Aruna, P.: A framework of keyword based image retrieval using proposed hog_sift feature extraction method from twitter dataset. Procedia Computer Science, vol. 132, pp. 1422–1431 (2018), doi: 10.1016/j.procs.2018.05.073
89. Weng, Y., Liu, L.: A collective anomaly detection approach for multidimensional streams in mobile service security. IEEE Access, vol. 7, pp. 49157–49168 (2019) doi: 10.1109/access.2019.2909750
90. Won, S., Cho, I., Sudusinghe, K., Xu, J., Zhang, Y., Van Der Schaar, M., Bhattacharyya, S. S.: A design methodology for distributed adaptive stream mining systems. Procedia Computer Science, vol. 18, pp. 2482–2491 (2013), doi: 10.1016/j.procs.2013.05.425
91. You, D., Wu, X., Shen, L., Deng, S., Chen, Z., Ma, C., Lian, Q.: Online feature selection for streaming features using self-adaption sliding-window sampling. IEEE Access, vol. 7, pp. 16088–16100 (2019) doi: 10.1109/access.2019.2894121
92. Yu, K., Ding, W., Simovici, D. A., Wang, H., Pei, J., Wu, X.: Classification with streaming features: An emerging-pattern mining approach. ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 9, no. 4, pp. 1–31 (2015) doi: 10.1145/2700409
93. Za'in, C., Pratama, M., Lughofer, E., Ferdaus, M., Cai, Q., Prasad, M.: Big data analytics based on panfis mapreduce. Procedia Computer Science, vol. 144, pp. 140–152 (2018) doi: 10.1016/j.procs.2018.10.514
94. Zhang, S., Zhang, Y., Yin, L., Yuan, T., Wu, Z., Luo, H.: Mining frequent items over the distributed hierarchical continuous weighted data streams in internet of things. IEEE Access, vol. 7, pp. 74890–74898 (2019) doi: 10.1109/access.2019.2911573
95. Zhou, Z., Zhang, X., Zhou, X., Liu, Y.: Semantic-aware visual abstraction of large-scale social media data with geo-tags. IEEE Access, vol. 7, pp. 114851–114861 (2019) doi: 10.1109/access.2019.2935471
96. Zhu, Y., Shasha, D.: Efficient elastic burst detection in data streams. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 336–345 (2003) doi: 10.1145/956750.956789