**INSTITUTO TECNOLÓGICO DE BUENOS AIRES – ITBA**

**ESCUELA DE INGENIERÍA Y GESTIÓN**

# DETECCIÓN AUTOMÁTICA DE ANOMALÍAS EN LOGS: UNA REVISIÓN VISUAL DEL ESTADO DEL ARTE

**AUTOR: Lic. ABOITIZ, TXOMIN MARTIN (Leg. Nº 105085)**

**TUTOR: Lic. AIZEMBERG, DIEGO ARIEL**

**TRABAJO FINAL INTEGRADOR**

**ESPECIALIZACIÓN EN CIENCIA DE DATOS**

**BUENOS AIRES**

**5 de Octubre de 2022**

# Tabla de contenidos

# Resumen

Con el objetivo de elaborar una revisión visual del estado del arte de la detección automática de anomalías en logs, se recopilaron 20 trabajos (publicados entre 2009 y 2021) centrados en dicha línea de investigación. 17 de ellos son trabajos experimentales y 3 son revisiones. Los trabajos fueron procesados para extraer información descriptiva asociada a la publicación y el contenido. Se prestó particular atención a las categorías de aprendizaje y los modelos de aprendizaje automático entrenados en los trabajos para detectar anomalías. También se prestó atención a la interacción entre los trabajos, a través de las citas. Con la información extraída, se construyó una base de datos de 3 tablas describiendo los autores, los trabajos y sus interacciones.

Con los datos, se construyó un notebook de visualizaciones en *ObservableHQ*. Estas son útiles para obtener una idea inicial de los países del mundo más involucrados en esta investigación, los trabajos más influyentes y los modelos de aprendizaje más utilizados. Dado que el tamaño muestral de trabajos utilizados aquí es pequeño, no se plantea que las tendencias observadas por las visualizaciones sean representativas de las tendencias reales. Sin embargo, con un mayor tamaño muestral, esta revisión visual podría ser útil para resumir información importante sobre el estado del arte de este tema, de manera que un lector no especializado pueda identificar rápidamente trabajos de interés para sus requerimientos específicos.



# *Abstract*

*With the objective of developing a visual survey of the state of the art of automatic anomaly detection in logs, 20 papers (published between 2009 and 2021) centered on this topic were gathered. 17 of them are experimental projects, and 3 are surveys. The papers were processed in order to extract descriptive information associated with the publication and the content. Particular attention was paid to the learning categories and the machine learning models trained to detect anomalies. The interactions between the papers was also noted, by way of their citations. With the information gathered, a database with 3 tables was constructed, describing the authors, the papers and their interactions.*

*With this data, a notebook of visualizations was built in ObservableHQ. These visualizations are useful to obtain an initial understanding of the countries most involved in this research, the most influential papers, and the most commonly used learning models. Given that the sample size of papers taken into account here is small, the tendencies observed by these visualizations are not considered to be representative of reality. However, with a larger sample size, this visual survey could be useful to*

*summarize important information about the state of the art of this topic, so that a non-specialized reader can quickly identify papers of interest for their specific requirements.*

## Palabras clave / *keywords*

Logs, Detección de anomalías, Aprendizaje automático, Revisión, Visualización /

*Logs, Anomaly detection, Machine learning, Survey, Visualization*

# Introducción

La automatización del análisis de logs en sistemas informáticos es un problema cuya relevancia e investigación ha aumentado significativamente, observándose una tendencia creciente del número de trabajos publicados en los últimos 20 años [Korzeniowski & Goczyła, 2022].

Debido al crecimiento del uso y de la complejidad de sistemas online, los volúmenes de logs han alcanzado niveles muy altos, llegando al orden de TBs y hasta PB de logs generados a diario [Lin et al., 2016]. Debido al volumen de estos logs, su complejidad y falta de estandarización, y la necesidad de responder en tiempo real a los errores, los métodos tradicionales de análisis y detección de anomalías, como la detección manual y el uso de palabras clave y expresiones regulares se están tornando cada vez más inviables [Li et al., 2019].

A pesar de la creciente investigación sobre este tema, existe una brecha importante entre el conocimiento generado por el sector académico y la implementación de soluciones dentro de las organizaciones privadas [Chen et al., 2021].

Este trabajo propone achicar esa brecha y facilitar la transmisión de conocimiento sobre las soluciones existentes en cuanto a la detección automática de anomalías en los archivos de logs. Bajo la hipótesis de que esta transmisión de conocimiento podría ser facilitada por medio de una revisión actualizada, interactiva y accesible, se buscará desarrollar una revisión visual para describir y visualizar el estado del arte de la detección automática de anomalías en logs.

Se implementarán diversas visualizaciones para 1) demostrar la evolución de la investigación en el tema, 2) comparar los tipos de modelos utilizados y 3) comparar el impacto de dichos trabajos. Esta revisión buscará ser interactiva, para permitir que el interesado pueda explorar en forma guiada dicho estado del arte, y así poder identificar con mayor rapidez las soluciones existentes y su relevancia con respecto a los objetivos específicos del usuario.

El informe se divide en tres partes. En la Parte I, se describen primeramente los antecedentes y el estado del arte de la investigación en el tema. Luego se define con mayor detalle el problema en cuestión. Seguido, se explaya sobre la justificación del estudio, y finalmente se discuten sus alcances y limitaciones. En la Parte II se especifica la hipótesis, los objetivos y la metodología propuesta. Finalmente, en la Parte III se presentan los resultados y su discusión subsiguiente. Por último, se plantea la conclusión, en función de los resultados obtenidos y los objetivos propuestos inicialmente.

# PARTE I

## Antecedentes y estado del arte

La automatización del análisis de logs es un problema que se viene discutiendo hace por lo menos dos décadas [Korzeniowski & Goczyła, 2022]. Uno de los primeros trabajos en proponer una solución es el de [Dickinson et al., 2001]. Los autores desarrollaron un método de detección de errores por medio de agrupamiento (*cluster filtering)*. Este trabajo sirvió como una aproximación inicial al problema, dado que en ese momento la investigación sobre el tema era más escasa. Uno de los problemas que destacan los autores en su discusión es que contaron con pocos datos para entrenar su modelo, y el error a detectar fue introducido explícitamente. Resaltan que se requiere más evidencia empírica y datos reales para validar el funcionamiento de la técnica.

Sin embargo, el análisis del problema tomó otra dimensión en los últimos diez años. Esto tiene que ver no sólo con los avances en las técnicas, sino con el aumento de la demanda de estas soluciones [Aué, 2016]. Además, hoy en día hay una gran disponibilidad de logs que se han hecho públicos para estas investigaciones. Un repositorio muy utilizado es Loghub [He et al., 2020], el cual contiene archivos de logs de muchos sistemas muy utilizados en la comunidad, como Hadoop, Apache, Mac y Windows. La utilidad de estos repositorios no es sólo la disponibilidad de datos, sino la posibilidad de estandarizar las condiciones experimentales entre distintos proyectos y así poder realizar comparaciones controladas.

Al día de hoy hay muchos trabajos que proponen distintos tipos de soluciones basadas en modelos de aprendizaje automático para detectar anomalías. Algunos trabajos proponen metodologías de aprendizaje supervisado. En [Lu et al., 2018], por ejemplo, se presenta un modelo utilizando redes neuronales convolucionales (CNN). [Du et al., 2017] utiliza un modelo de *Deep Learning* utilizando *Long Short Term Memory* (LSTM). Otros investigadores desarrollaron modelos de aprendizaje no supervisado, utilizando distintas técnicas. [Farzad & Gulliver, 2020] plantea un uso combinado de dos redes de *Autoencoders* y un *Isolation Forest* para extraer *features*, detectar anomalías y luego predecir valores positivos (anómalos) en datos nuevos. Por otro lado, [Brown et al., 2018] plantea un modelo de redes neuronales recurrentes (RNN). Además, en este trabajo se discute que la falta de interpretabilidad de estos modelos de caja negra es un problema a la hora de analizar los logs, dado que es fundamental un entendimiento analítico de los errores a la hora de diagnosticar y resolver. Ellos intentan aumentar esta interpretabilidad utilizando modelos de lenguaje con mecanismos de atención.

En general, la dicotomía entre modelos supervisados y no supervisados es que los modelos

supervisados tienen niveles de eficacia más altos, pero son menos robustos frente a los problemas que se enfrenta con el análisis de logs, siendo estos: la imposibilidad de etiquetar todos los mensajes, los volúmenes abrumadores de logs generados a diario y las diferencias *entre* y *dentro* de sistemas. Debido a estas características, entrenar un modelo supervisado es muy costoso y poco eficiente, especialmente si se tiene en cuenta que las anomalías deben ser detectadas en tiempo real.

Sin embargo, también se han desarrollado modelos de aprendizaje semi-supervisado, con el objetivo de amalgamar lo mejor de ambas metodologías y reducir el *trade-off*. Un ejemplo de este enfoque es el trabajo de [Yang et al., 2021], donde en una primera etapa se agrupan los mensajes para generar etiquetas por medio de estimación probabilística (*Probabilistic Label Estimation)*. Una vez generadas las etiquetas, se procede a entrenar un modelo supervisado para detectar anomalías. Por otro lado, [Duan et al., 2021] implementa otro modelo semi-supervisado utilizando *Q-Learning*. Además de detectar anomalías, su modelo permite ordenar las mismas en función de su severidad, lo cual es una capacidad que no está muy desarrollada en otros estudios.

En cuanto a las revisiones de literatura, estas se actualizan periódicamente. Uno de los más recientes es el trabajo de [Korzeniowski & Goczyła, 2022]. Este trabajo es muy abarcativo y toma en cuenta muchos factores importantes, tales como los objetivos, los dominios de investigación y el tipo de logs utilizados. Sin embargo, dado que su objetivo principal es proveer una revisión de alto nivel para estadíos iniciales de investigación, no llega a proveer explicaciones conceptuales de los tipos de modelos utilizados.

## Definición del problema

Hoy en día, el uso de software de diversos tipos para ejecutar tareas dentro de una empresa es muy común y hasta imprescindible en la mayoría de los casos. En particular, las empresas tecnológicas utilizan software externo (ya sea de servicios contratados o de código abierto) pero también software propio, tanto para proveer sus servicios a sus clientes como para ejecutar tareas internas que garanticen el funcionamiento del sistema. Para que todos estos procesos funcionen adecuadamente, se requiere un monitoreo constante, lo cual se hace cada vez más difícil a medida que crece la empresa, debido al aumento de la cantidad de procesos en ejecución.

Para monitorear dichos procesos, una de las herramientas fundamentales son los archivos de logs generados por los mismos, en los cuales se registra información semi estructurada, reportando los detalles no sólo del correcto funcionamiento del sistema, sino también de alertas y errores. Son estos los

que permiten identificar y diagnosticar problemas y posibles optimizaciones asociados a la seguridad, la *performance*, el aprovisionamiento de recursos y análisis de perfiles [Oliner et al., 2012].

Apareado al aumento en tamaño y complejidad de sistemas online con gran comunidad de usuarios (como Amazon, Google y Microsoft) y el aumento de dependencia de software y sistemas informáticos, el volumen de logs generados a diario ha alcanzado proporciones abrumadoras. Sistemas de este tipo pueden generar decenas de terabytes de logs por día, y hasta petabytes en algunos casos [Lin et al., 2016].

Debido al inmenso volumen de logs generados a diario (que además tiende a aumentar a medida que crece una organización), su naturaleza semi estructurada e inestable [Zhang et al., 2019] , el monitoreo de estos logs se convierte en un problema que cada vez se hace más difícil resolver por medio de detección humana.

Sumado a esto, a pesar de haber mucha investigación sobre el problema y varias soluciones propuestas, existe una brecha entre el conocimiento generado por el sector académico y el conocimiento adquirido por el sector privado. A pesar de estar al tanto del problema, en muchas organizaciones se tarda en implementar soluciones de este tipo. Esto se puede deber a diversos motivos, pero [Chen et al., 2021] destaca la falta de:

1) conocimiento especializado para abordar una solución propia,

2) soluciones de código abierto, y

3) comprensión sobre las existentes soluciones desarrolladas en el sector académico.

Estas limitaciones están asociadas a la poca cantidad de revisiones comparativas de literatura [He et al., 2016]. En este trabajo se postula que además se requiere más literatura con un lenguaje y formato accesible para lectores no especializados.

## Justificación del estudio

Si bien en los últimos años se ha realizado mucha investigación sobre este tema, y se han propuesto diversas soluciones, el problema sigue estando sin resolver por completo. Esto se podrá deber a diversos motivos, ya sea que las soluciones existentes no interpreten adecuadamente el contenido de los logs, que las soluciones sean difíciles y/o costosas de implementar o porque más allá del conocimiento académico, que las empresas no tengan suficiente conocimiento sobre el tema, o que simplemente no estén en un estadío de desarrollo que imposibilite el análisis manual de los logs (lo cual no significa que ese método sea eficiente en estos casos).

Este es un problema vasto con muchas dimensiones de análisis, y por lo tanto requiere continua investigación y desarrollo. Estas dimensiones son parte de dos niveles principales. Por un lado el nivel asociado al *desarrollo* de soluciones individuales. Algunas dimensiones asociadas a este nivel son: el desarrollo del pipeline para procesar los logs y convertirlos en información estructurada que sea analizable, el desarrollo de modelos de aprendizaje (ya sea supervisado, semi-supervisado o no supervisado) para catalogar los mensajes, la optimización de dichos modelos para separar adecuadamente el ruido de la información y para interpretar adecuadamente los mensajes, y el diagnóstico de los errores y sus motivos. El segundo nivel corresponde a la *integración* de estas soluciones en un sistema productivo. Aquí entran en juego otras dimensiones de análisis, que involucran la revisión del estado del arte y la transmisión de conocimiento. Estas dimensiones de análisis son fundamentales porque son el puente entre la investigación y la implementación. Las revisiones no sólo actualizan y resumen la información disponible, sino que también retroalimentan la comunicación entre los distintos actores, permitiendo el avance del desarrollo y el desarrollo de la implementación.

Este trabajo se enfocará sobre el segundo nivel de análisis. Dada la continua investigación sobre el tema, es esencial generar revisiones actualizadas, ya que el estado del arte evoluciona constantemente. Además, para poder implementar soluciones automatizadas de análisis de logs dentro de una empresa, puede ser muy costoso realizar una investigación detallada de todas las tecnologías desarrolladas. En este sentido, una revisión que sea fácil de entender e interpretar sería muy valiosa para la exploración inicial del problema, porque permitiría la rápida identificación de las tecnologías desarrolladas y las diferencias de enfoques. Como cada empresa tiene sus propias condiciones y necesidades, encontrar el enfoque más compatible es fundamental para que puedan desarrollar una solución pertinente.

## Alcances del trabajo y limitaciones

Se mencionó anteriormente que el problema en cuestión tiene muchas dimensiones de análisis. Este trabajo propone enfocarse en revisar el estado del arte de la investigación asociada a la detección automática de anomalías en logs. A su vez, se restringirá a estudiar y comparar los trabajos a nivel cualitativo. Es decir, se comparará metodologías y modelos utilizados, pero no los resultados y las evaluaciones de los modelos. Sin embargo, se realizaron algunas comparaciones cuantitativas asociadas a la cantidad de trabajos en función de sus respectivos enfoques.

Dado el contexto de este trabajo, cabe destacar algunas limitaciones importantes. Al ser el trabajo final de una especialización, no se cuenta con los recursos fundamentales de un proyecto de investigación: fondos, tiempo, y un equipo con conocimiento especializado del dominio.

Por un lado, proyectos de esta dimensión pueden demorar años en obtener resultados, pero para este trabajo se cuenta con un margen temporal de meses. Además, debido a las responsabilidades laborales del alumno, no será posible dedicarle el tiempo completo de la semana al desarrollo del trabajo.

Por el otro, los recursos humanos disponibles para este proyecto están limitados a un alumno y su director. No se contará con un equipo de investigadores para distribuir las tareas, y tampoco se contará con investigadores con conocimiento específico sobre el tema. Al ser un problema con muchos aspectos técnicos y conceptuales, esta limitación también será una de gran importancia.

Estas limitaciones determinan en gran medida los alcances restringidos del estudio, ya que idealmente un estudio de este tipo incluiría comparaciones cuantitativas entre modelos. Además, al acotar el rango temporal, se excluirán del estudio muchos trabajos potencialmente valiosos.

# PARTE II

## Hipótesis

Una revisión sobre la detección automática de anomalías en logs actualizada, interactiva y accesible para lectores no especializados facilitaría la transmisión de conocimiento y la identificación de soluciones en función de los requerimientos específicos de una organización.

## Objetivos

### A) Objetivo general

Desarrollar una revisión visual para describir y explorar el estado del arte de la detección automática de anomalías en logs.

### B) Objetivos específicos

1. Recopilar trabajos de investigación recientes sobre detección automática de anomalías en logs.
2. Construir una base de datos para identificar y describir:
    a. Los trabajos
    b. Los autores y sus instituciones
    c. La interacción entre los trabajos
3. A partir de la información recopilada, generar visualizaciones que describan las tendencias y los contenidos de las investigaciones.
4. Publicar los datos y las visualizaciones en un notebook de *ObservableHQ* de manera que un usuario pueda interactuar con los resultados.

## Metodología

### A) Recopilación de bibliografía

Utilizando el motor de búsqueda de trabajos académicos de Google (*Google Scholar)*, se seleccionaron 20 trabajos asociados a la detección automática de anomalías en logs (ya sean trabajos experimentales o revisiones).

Para acotar el rango de estudio y evitar sesgos en la selección de trabajos, se implementó un criterio de búsqueda y selección. Para la búsqueda, se utilizó dos combinaciones de palabras claves: *automated log analysis* y *log analysis machine learning*. Sólo se seleccionaron trabajos asociados a la detección de *anomalías* en logs utilizando modelos de aprendizaje automático. No hubo discriminación en cuanto a al tipo de logs analizados o el tipo de modelos implementados.

**B) Base de datos**

A partir de los trabajos seleccionados, se construyeron tres tablas principales[1]:

*Papers:* Consiste en una descripción resumida de los trabajos (categorizados en función del tipo de estudio: trabajo experimental o revisión), su contenido y su impacto. En esta tabla se incluye información sobre las categorías de modelos de aprendizaje automático implementados (supervisado, no supervisado, semi-supervisado, refuerzo), los modelos específicos implementados, la cantidad de veces que el trabajo fue citado de acuerdo a *Google Scholar* (hasta el 7 de julio 2022), las palabras clave definidas en el trabajo, y otros aspectos descriptivos de la publicación, como el año y la revista de publicación. Además, se incluyó una métrica de impacto del trabajo, definida como la razón entre el número de citas y el número de años transcurridos desde la publicación hasta el año vigente (2022).

*Authors:* Una lista de todos los autores referenciados en las publicaciones y las instituciones a las cuales están afiliados.

*Cited:* Identifica, para cada publicación seleccionada, los trabajos citados (dentro de los seleccionados), en formato *source-targets*.

Las tablas fueron guardadas en formato *.tsv*, y son accesibles públicamente en un repositorio de *GitHub* (ver apéndice).

**C) Notebook de visualizaciones**

Se desarrolló un notebook en *ObservableHQ*[2] (ver apéndice), donde se muestran las tablas desarrolladas, y una serie de visualizaciones descriptivas de los trabajos. El procesamiento de los datos fue programado en JavaScript, y las visualizaciones se implementaron con la librería Plot (también de JavaScript). Las visualizaciones generadas son las siguientes:

[1] Las tablas serán identificadas por los nombres (en inglés) otorgados en el notebook de *ObservableHQ*. Para una descripción detallada de los campos incluidos en las tablas, referirse a la tabla *Table Details*, disponible en el notebook y como archivo *.tsv* en el repositorio de *GitHub* (ver apéndice).
[2] https://observablehq.com/

- Número de afiliaciones de autores por país (asociados a las publicaciones seleccionadas).
- Número de trabajos publicados por año, agrupados por categoría de estudio (experimental/revisión).
- Nubes de palabras:
    - Palabras clave de los trabajos,
    - Modelos de aprendizaje automático implementados,
- Gráfico de nodos y enlaces describiendo las interacciones (por citas) entre los trabajos.
- Número de trabajos agrupados por categoría de aprendizaje.

# PARTE III

## Resultados

A continuación se muestran un conjunto de gráficos de las visualizaciones generadas y publicadas en el notebook de *ObservableHQ* (identificados en este informe con los mismos números que en el notebook):

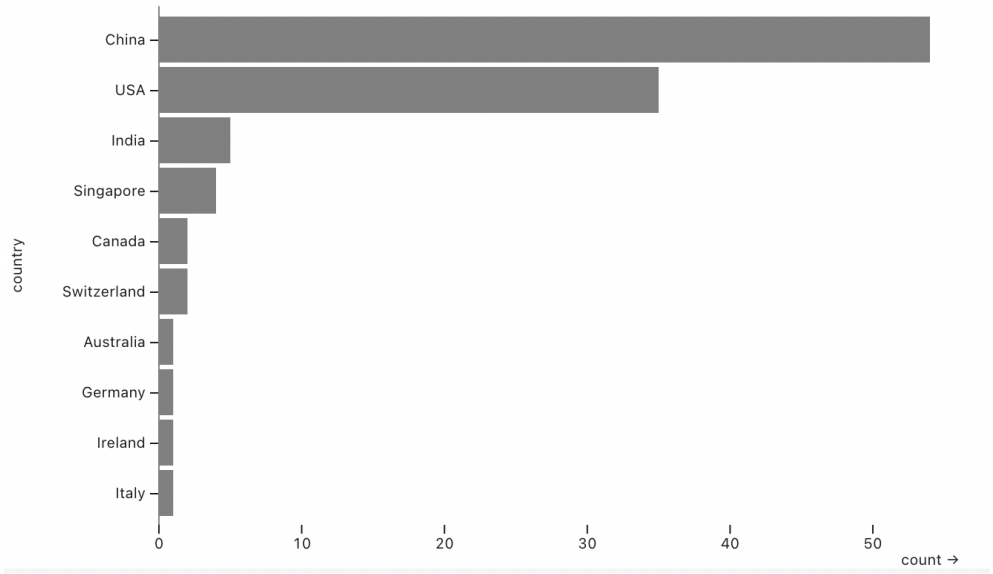


***Figura 2.1:*** *Afiliaciones de autores agrupadas por país*

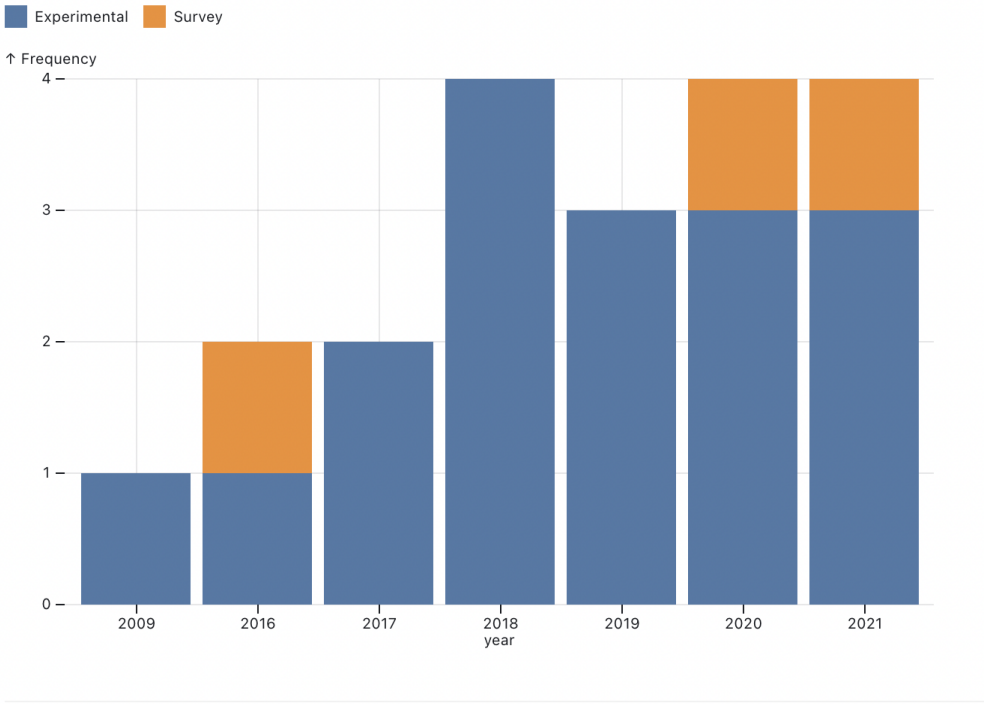


***Figura 2.2:*** *Número de trabajos publicados por año, agrupados por tipo de estudio*

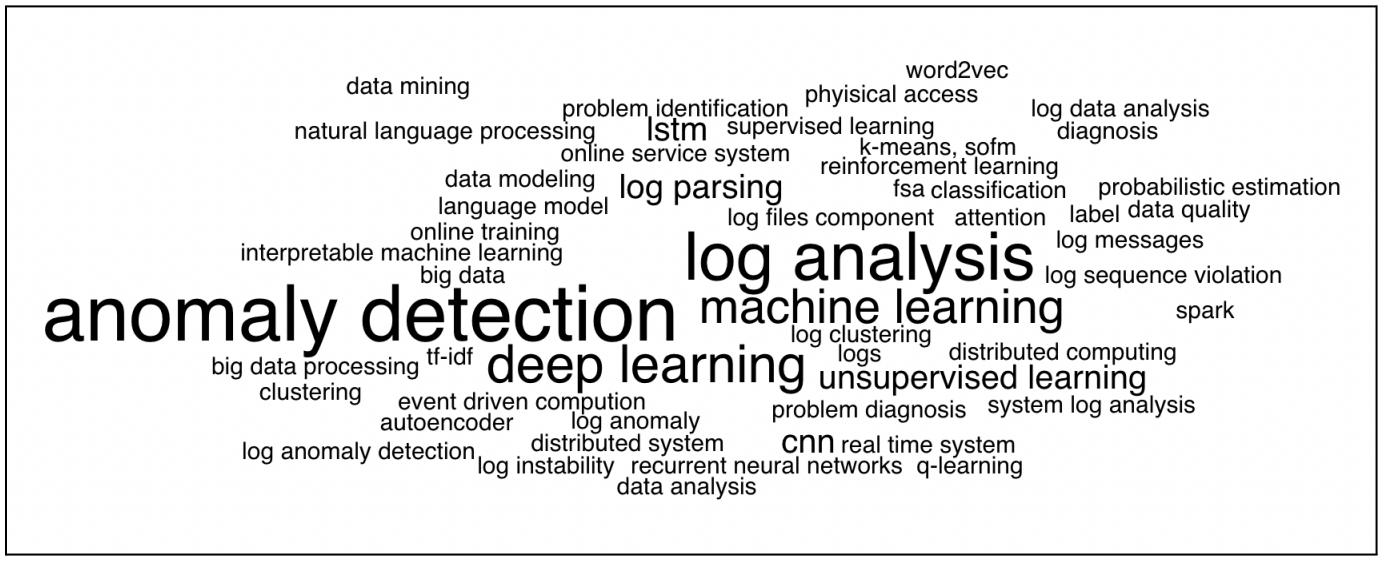


***Figura 2.3:*** *Nube de palabras - palabras clave de los trabajos*

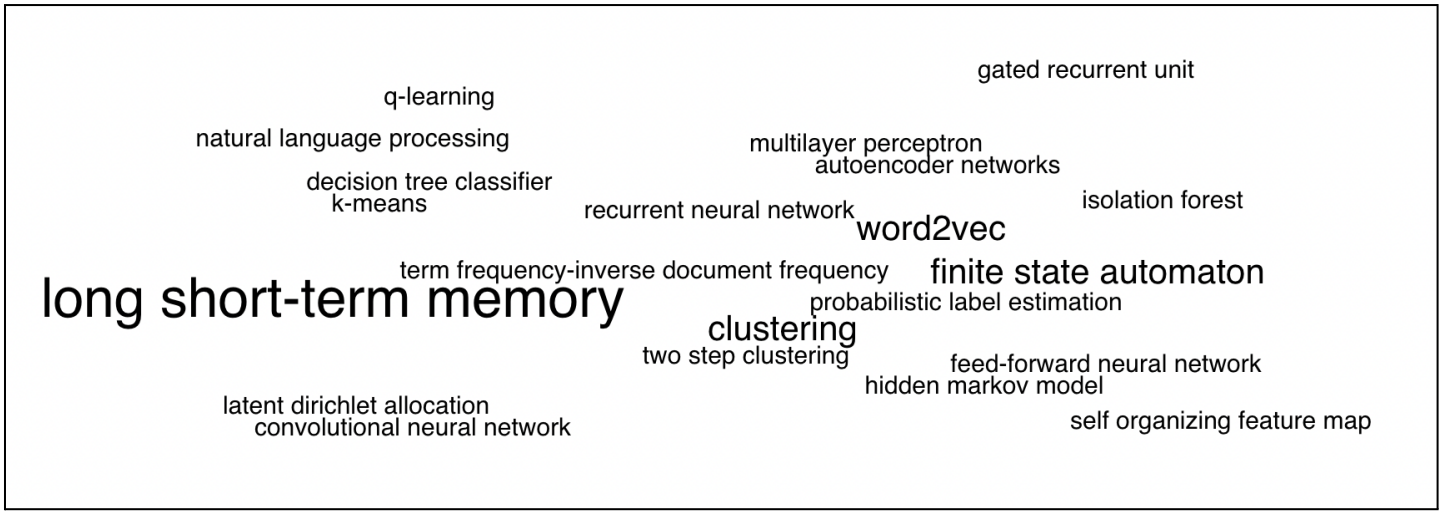


***Figura 2.4:*** *Nube de palabras - modelos implementados en los trabajos*

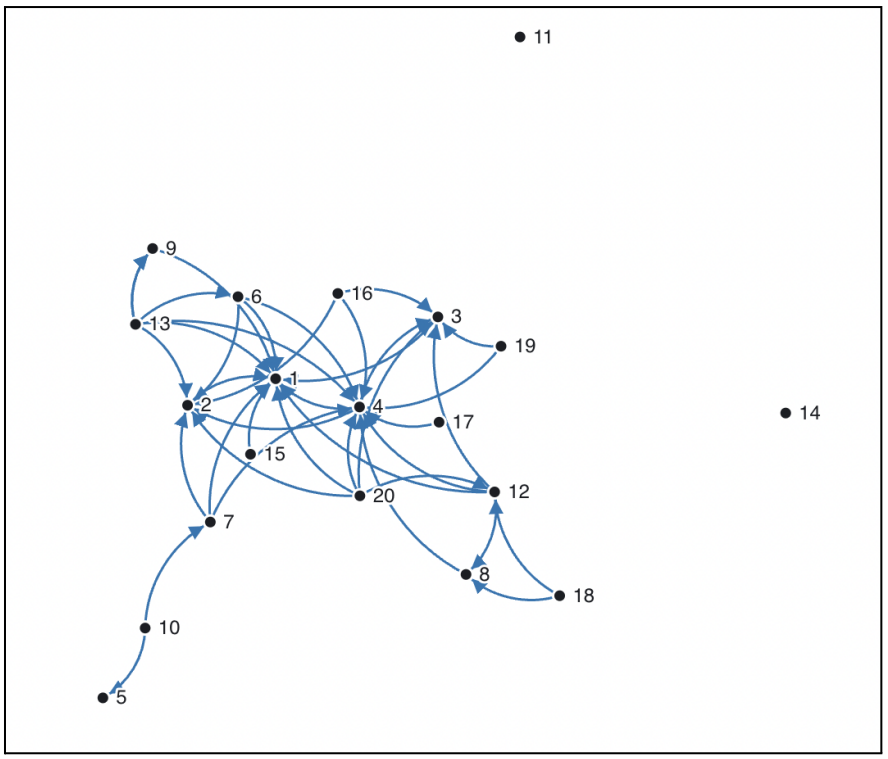


***Figura 2.5:*** *Gráfico de nodos - red de citas entre los trabajos seleccionados*

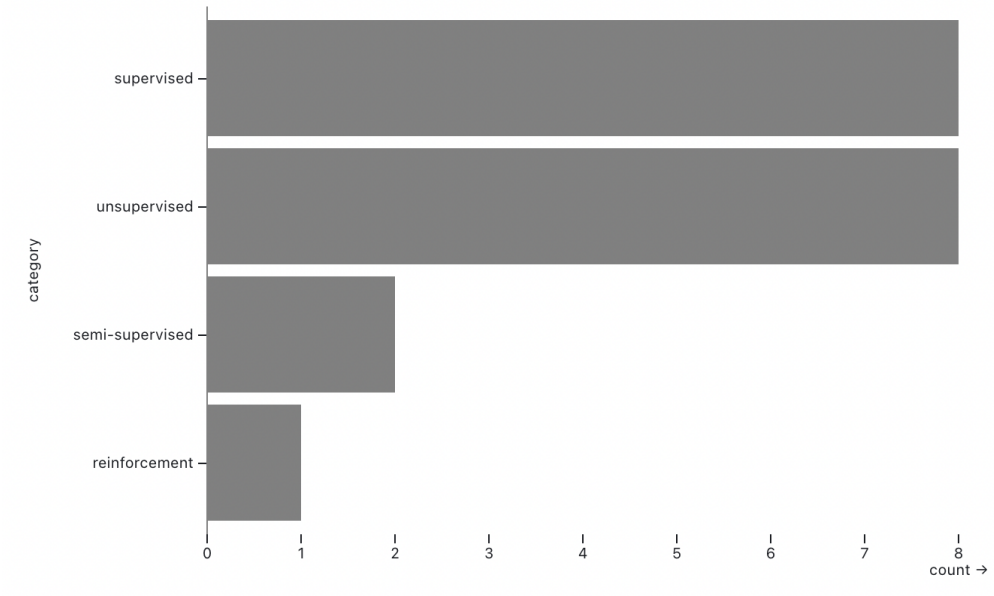


***Figura 2.6:*** *Número de trabajos agrupados por categoría de aprendizaje*

## Discusión

Las visualizaciones generadas fueron útiles para discernir ciertas tendencias (cuya validez será discutida a continuación).

Se observa que la mayoría de los trabajos fueron publicados por instituciones chinas, seguidas por instituciones estadounidenses (Figura 2.1). El resto de los países representados tienen un número de publicaciones mucho menor que los dos primeros.

Por otro lado, a partir de las nubes de palabras (Figuras 2.2 y 2.3), se observa que *deep learning* es una técnica muy común. En particular, *long short-term memory* parece ser un tipo de modelo muy utilizado en los trabajos.

Cuantificando las categorías de aprendizaje, vemos que en los trabajos seleccionados no hay una preferencia marcada entre modelos de aprendizaje supervisado y no supervisado, dado que están igualmente representados (Figura 2.6).

Se pueden detectar dos trabajos como particularmente influyentes, debido a su presencia en las citas de otros trabajos (Figura 2.5). Referenciados por su ID (ver tabla *Papers*), vemos que los trabajos con ID 1 y 4 fueron los más citados. Estos corresponden a las publicaciones de [Qiang Fu et al., 2009] y [Min Du et al., 2017], respectivamente.

Es importante mencionar que las tendencias destacadas en esta discusión son *tentativas*. El alcance de este análisis está limitado por el número de trabajos incluídos. Con más recursos (principalmente de tiempo), se podrían incluir más publicaciones, para aumentar la robustez de los resultados.

También se podrían haber generado otras visualizaciones interesantes, que tomen en consideración, por ejemplo, el impacto del trabajo asociado a otras métricas, como las categorías de aprendizaje o los modelos implementados.

Sin embargo, más allá de los resultados obtenidos, esta “revisión visual” tiene el potencial de ser una herramienta muy útil para llevar a cabo exploraciones iniciales sobre el tema. Hay dos motivos principales a destacar. Por un lado, permite estudiar el tema sin tener mucho conocimiento inicial, y luego identificar trabajos de interés en base a características que resaltan en las visualizaciones (por ejemplo buscando trabajos que tengan las palabras clave o los modelos de aprendizaje prevalentes, o identificando trabajos en función de su influencia sobre otros). Por otro lado, es una herramienta que, por su naturaleza de código abierto y de programación dinámica, es muy simple de desarrollar, ya sea

agregando nuevos trabajos (sólo requiere actualizar los datos, y las visualizaciones se actualizan automáticamente) o trabajando de manera colaborativa entre desarrolladores para generar más visualizaciones. El notebook también tiene la posibilidad de ser bifurcado y utilizado para visualizar otros temas de investigación (haciendo las modificaciones pertinentes al código).

# Conclusión

Se pudo llevar a cabo los objetivos planteados en este trabajo. Habiendo recopilado 20 trabajos de investigación sobre la detección automática de anomalías en logs, se generó una base de datos a partir de variables medibles (principalmente cualitativas), lo cual a su vez permitió desarrollar un notebook en *ObservableHQ* con visualizaciones sobre los países representados en esta línea de investigación, los enfoques de los trabajos, y las interacciones entre ellos.

Si bien los resultados obtenidos en las visualizaciones resultan útiles sólo a modo tentativo para esclarecer el estado del arte de este tema, el objetivo de este trabajo no se trata de los resultados en sí, sino de proveer una *herramienta* de exploración. La utilidad de esta herramienta depende en gran medida de la validez de los resultados. Dado que el tamaño muestral de los trabajos seleccionados es pequeño, se considera que la validez de los resultados en este caso es de bajo alcance. Sin embargo, los cimientos sobre los cuales está construida esta herramienta de exploración permiten aumentar su alcance con relativamente poco esfuerzo, lo cual es potencialmente muy valioso.

# BIBLIOGRAFÍA


Allagi, S., & Rachh, R. (2019, March). Analysis of Network log data using Machine Learning. In *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)* (pp. 1-3). IEEE.

Aué, J. (2016). Log analysis from A to Z: a literature survey.

Brown, A., Tuor, A., Hutchinson, B., & Nichols, N. (2018, June). Recurrent neural network attention mechanisms for interpretable system log anomaly detection. In *Proceedings of the First Workshop on Machine Learning for Computing Systems* (pp. 1-8).

Cao, Q., Qiao, Y., & Lyu, Z. (2017, December). Machine learning to detect anomalies in web log analysis. In *2017 3rd IEEE International Conference on Computer and Communications (ICCC)* (pp. 519-523). IEEE.

Chen, Z., Liu, J., Gu, W., Su, Y., & Lyu, M. R. (2021). Experience Report: Deep Learning-based System Log Analysis for Anomaly Detection. *arXiv preprint arXiv:2107.05908*.

Debnath, B., Solaimani, M., Gulzar, M. A. G., Arora, N., Lumezanu, C., Xu, J., ... & Khan, L. (2018, July). LogLens: A real-time log analysis system. In *2018 IEEE 38th international conference on distributed computing systems (ICDCS)* (pp. 1052-1062). IEEE.

Dickinson, W., Leon, D., & Fodgurski, A. (2001, May). Finding failures by cluster analysis of execution profiles. In *Proceedings of the 23rd International Conference on Software Engineering. ICSE 2001* (pp. 339-348). IEEE.

Du, M., Li, F., Zheng, G., & Srikumar, V. (2017, October). Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In Proceedings of the 2017 ACM SIGSAC conference on computer and communications security (pp. 1285-1298).

Duan, X., Ying, S., Yuan, W., Cheng, H., & Yin, X. (2021). QLLog: A log anomaly detection method based on Q-learning algorithm. *Information Processing & Management*, *58*(3), 102540.

Farzad, A., & Gulliver, T. A. (2020). Unsupervised log message anomaly detection. *ICT Express*, *6*(3), 229-237.

Fu, Q., Lou, J. G., Wang, Y., & Li, J. (2009, December). Execution anomaly detection in distributed systems through unstructured log analysis. In *2009 ninth IEEE international conference on data mining* (pp. 149-158). IEEE.

He, S., He, P., Chen, Z., Yang, T., Su, Y., & Lyu, M. R. (2021). A survey on automated log analysis for reliability engineering. *ACM Computing Surveys (CSUR)*, *54*(6), 1-37.

He, S., Zhu, J., He, P., & Lyu, M. R. (2016, October). Experience report: System log analysis for anomaly detection. In *2016 IEEE 27th international symposium on software reliability engineering (ISSRE)* (pp. 207-218). IEEE.

He, S., Zhu, J., He, P., & Lyu, M. R. (2020). Loghub: a large collection of system log datasets towards automated log analytics. *arXiv preprint arXiv:2008.06448*.

Korzeniowski, Ł., & Goczyła, K. (2022). Landscape of Automated Log Analysis: a Systematic Literature Review and Mapping Study. *IEEE Access*.

Layer, L., Abercrombie, D. R., Bakhshiansohi, H., Adelman-McCarthy, J., Agarwal, S., Hernandez, A. V., ... & Vlimant, J. R. (2020). Automatic log analysis with NLP for the CMS workflow handling. In *EPJ Web of Conferences* (Vol. 245, p. 03006). EDP Sciences.

Li, G., Zhu, P., Cao, N., Wu, M., Chen, Z., Cao, G., ... & Gong, C. (2019). Improving the system log analysis with language model and semi-supervised classifier. *Multimedia Tools and Applications*, *78*(15), 21521-21535.

Lin, Q., Zhang, H., Lou, J. G., Zhang, Y., & Chen, X. (2016, May). Log clustering based problem identification for online service systems. In *Proceedings of the 38th International Conference on Software Engineering Companion* (pp. 102-111).

Lu, S., Wei, X., Li, Y., & Wang, L. (2018, August). Detecting anomaly in big data system logs using convolutional neural network. In 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech) (pp. 151-158). IEEE.

Oliner, A., Ganapathi, A., & Xu, W. (2012). Advances and challenges in log analysis. *Communications of the ACM*, *55*(2), 55-61.

Poh, J. P., Lee, J. Y. C., Tan, K. X., & Tan, E. (2020, July). Physical Access Log Analysis: An Unsupervised Clustering Approach for Anomaly Detection. In *Proceedings of the 3rd International Conference on Data Science and Information Technology* (pp. 12-18).

Wang, M., Xu, L., & Guo, L. (2018, September). Anomaly detection of system logs based on natural language processing and deep learning. In *2018 4th International Conference on Frontiers of Signal Processing (ICFSP)* (pp. 140-144). IEEE.

Yadav, R. B., Kumar, P. S., & Dhavale, S. V. (2020, June). A survey on log anomaly detection using deep learning. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 1215-1220). IEEE.

Yang, L., Chen, J., Wang, Z., Wang, W., Jiang, J., Dong, X., & Zhang, W. (2021, May). Semi-supervised log-based anomaly detection via probabilistic label estimation. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)* (pp. 1448-1460). IEEE.

Zhang, X., Xu, Y., Lin, Q., Qiao, B., Zhang, H., Dang, Y., ... & Zhang, D. (2019, August). Robust log-based anomaly detection on unstable log data. In *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (pp. 807-817).

Zhao, Z., Xu, C., & Li, B. (2021). A LSTM-Based Anomaly Detection Model for Log Analysis. *Journal of Signal Processing Systems*, *93*(7), 745-751.

# APÉNDICE

## Trabajos seleccionados

| ID | Año | Autores | Título |
|---|---|---|---|
| 1 | 2009 | Qiang Fu; Jian-Guang Lou; Yi Wang; Jiang Li | Execution Anomaly Detection in Distributed Systems through Unstructured Log Analysis |
| 2 | 2016 | Shilin He; Jieming Zhu; Pinjia He; Michael R. Lyu | Experience Report: System Log Analysis for Anomaly Detection |
| 3 | 2016 | Qingwei Lin; Hongyu Zhang; Jian-Guang Lou; Yu Zhang; Xuewei Chen | Log clustering based problem identification for online service systems |
| 4 | 2017 | MIn Du; Feifei Li; Guineng Zheng; Vivek Srikumar | DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning |
| 5 | 2017 | Qiming Cao; Yinrong Qiao; Zhong Lyu | Machine learning to detect anomalies in web log analysis |
| 6 | 2018 | Siyang Lu; Xiang Wei; Yandong Li; Liqiang Wang | Detecting Anomaly in Big Data System Logs Using Convolutional Neural Network |
| 7 | 2018 | Biplog Bebnath; Mohiuddin Solaimani; Muhammad Ali Gulzar; Nipun Arora; Cristian Lumezanu; Jianwu Xu; Bo Zong; Hui Zhang; Guofei Jiang; Latifur Khan | LogLens: A Real-time Log Analysis System |
| 8 | 2018 | Andy Brown; Aaron Tuor; Brian Hutchinson; Nicole Nichols | Recurrent Neural Network Attention Mechanisms for Interpretable System Log Anomaly Detection |
| 9 | 2018 | Mengying Wang; Lele Xu; Lili Guo | Anomaly Detection of System Logs Based on Natural Language Processing and Deep Learning |
| 10 | 2019 | Shridhar Allagi; Rashmi Rachh | Analysis of Network log data using Machine Learning |
| 11 | 2019 | Guofu Li; Pengjia Zhu; Ning Cao; Mei Wu; Zhiyi Chen; Guangsheng Cao; Hongjun Li; Chenjing Gong | Improving the system log analysis with language model and semi-supervised classifier |
| 12 | 2019 | Xu Zhang; Yong Xu; Hongyu Zhang; Yingnong Dang; Chunyu Xie; Xinsheng Yang; Junjie Chen; Xiaoting He; Randolph Yao; Jiang-Guang Lou; Murali Chintalapati; Furao Shen; Dongmei Zhang | Robust log-based anomaly detection on unstable log data |
| 13 | 2020 | Rakesh Bahadur Yadav; P Santosh Kumar; Sunita Vikrant Dhavale | A Survey on Log Anomaly Detection using Deep Learning |
| 14 | 2020 | Lukas Layer; Daniel Robert Abercrombie; Hamed Bakhshiansohi; Jennifer Adelman-McCarthy; Sharad Agarwal; Andres Vargas Hernandez; Weinan Si; Jean-Roch Vlimant | Automatic log analysis with NLP for the CMS workflow handling |
| 15 | 2020 | Ju Peng Poh; Jun Yu Charles Lee; Kah Xuan Tan; Eric Tan | Physical Access Log Analysis: An Unsupervised Clustering Approach for Anomaly Detection |
| 16 | 2020 | Amir Farzad; T. Aaron Gulliver | Unsupervised log message anomaly detection |
| 17 | 2021 | Zhijun Zhao; Chen Xu; Bo Li | A LSTM-Based Anomaly Detection Model for Log Analysis |
| 18 | 2021 | Shilin He; Pinjia He; Zhuangbin Chen; Tianyi Yang; Yuxing Su; Michael R. Lyu | A Survey on Automated Log Analysis for Reliability engineering |
| 19 | 2021 | Xiaoyu Duan; Shi Ying; Wanli Yuan; Hailong Cheng; Xiang Yin | QLLog: A log anomaly detection method based on Q-learning algorithm |
| 20 | 2021 | Lin Yang; Junjie Chen; Zan Wang; Weijing Wang; Jiajun Jiang; Xuayuan Dong; Wenbing Zhang | Semi-Supervised Log-Based Anomaly Detection via Probabilistic Label Estimation |

## Enlaces

URL del notebook de *ObservableHQ* (código abierto y accesible públicamente):

https://observablehq.com/@txaboitiz/automated_log_analysis

URL del repositorio con los datos recopilados (accesible públicamente):

https://github.com/txaboitiz/automated_log_analysis_review