Evaluación y comparación de algoritmos para la detección automática de eventos sísmicos

Evaluation and comparison of algorithms for the automatic detection of seismic events

Álvaro Eduardo Armijos Sarango
Departamento de Eléctrica, Electrónica y Telecomunicaciones, Universidad de Cuenca, Ecuador
alvaro.armijos@ucuenca.edu.ec

Iván Santiago Palacios Serrano
Departamento de Eléctrica, Electrónica y Telecomunicaciones, Universidad de Cuenca, Ecuador
ivan.palacios@ucuenca.edu.ec

Santiago Renán González Martínez
Departamento de Eléctrica, Electrónica y Telecomunicaciones, Universidad de Cuenca, Ecuador
santiago.gonzalezm@ucuenca.edu.ec

Revista Tecnológica ESPOL - RTE
vol. 33, núm. 2, Esp. p. 58 - 74, 2021
Escuela Superior Politécnica del Litoral, Ecuador
ISSN: 0257-1749
ISSN-E: 1390-3659
rte@espol.edu.ec

Recepción: 25 junio 2021
Aprobación: 28 septiembre 2021

DOI: https://doi.org/10.37815/rte.v33n2.830

Resumen: La detección temprana de eventos sísmicos permite reducir daños materiales, el número de personas afectadas e incluso salvar vidas. En particular, la actividad sísmica en Ecuador es alta, dado que se encuentra en el denominado Cinturón de Fuego del Pacífico. En tal contexto, el presente artículo tiene como objetivo comparar algoritmos para la detección automática de eventos sísmicos. Dicha comparación se realiza con respecto a la funcionalidad y configuración de los parámetros requeridos para cada algoritmo. Además, la implementación se lleva a cabo sobre una plataforma computacional tipo SBC (Single Board Computer) con la finalidad de obtener una herramienta portable, escalable, económica y de bajo costo computacional. Los métodos comparados son: Classic STA/LTA, Recursive STA/LTA, Delayed STA/LTA, Z-detector, Baer and Kradolfer picker y AR-AIC (Autoregressive-Akaike-Information-Criterion-picker). Para la evaluación y comparación se desarrollan múltiples experimentos empleando registros sísmicos reales proporcionados por la Red Sísmica del Austro (RSA), disponibles como fuente de entrada a los algoritmos. Como resultado se obtiene que el algoritmo Classic STA/LTA presenta el mejor rendimiento, ya que del total de eventos reales (58), solo un evento no fue detectado. Además, se consiguen 6 falsos negativos, logrando un 98,2% de precisión. Cabe recalcar que el software utilizado para la comparación de algoritmos de detección de eventos sísmicos está disponible de forma libre.

Palabras clave: SBC, Classic STA/LTA, Recursive STA/LTA, Delayed STA/LTA, Z Detector, Baer and Kradolfer picker, AR-AIC, eventos sísmicos.

Abstract: Early seismic events detection might reduce the number of people affected and save lives. In particular, the seismic activity in Ecuador is high, given its location along the zone named the Pacific Belt of Fire. In this context, this paper presents a solution to compare algorithms for detecting seismic events. This comparison was performed both concerning the functionality and the configuration of the parameters required

for each algorithm. This solution was implemented on an SBC platform (Single Board Computer) to obtain a portable, scalable, economical, and low-cost computational tool. The methods compared were: Classic STA/LTA, Recursive STA/LTA, Delayed STA/LTA, Z Detector, Baer and Kradolfer picker, and AR-AIC (Autoregressive-Akaike-Information-Criterion-picker). For the evaluation and comparison, multiple experiments were carried out using real data provided by the Regional Seismological Network (RSA). In particular, such registers were used as input data to the seismic algorithms. Results revealed that the algorithm with the best performance was the Classic STA/LTA, since from the total number of real events (58), only one was not detected. In addition, 6 false negatives were obtained, achieving 98,2% of precision. Finally, the software used for the comparison of the algorithms has been released for free usage, which represents another contribution of this work in the context of seismic analysis.

Keywords: SBC, Classic STA/LTA, Recursive STA/LTA, Delayed STA/LTA, Z Detector, Baer and Kradolfer picker, AR-AIC, seismic events.

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

Introducción

Los movimientos sísmicos son una causante crucial de grandes pérdidas de vidas y destrucción de propiedades. Acorde a los registros y reportes disponibles, en Ecuador el historial sísmico da cuenta de eventos a lo largo de cinco siglos en la región Sierra y al menos un siglo para la zona del Litoral, ver Beauval et al. (2013). Si bien eventos de magnitudes considerables han tenido lugar en la región del Litoral debido a la zona de subducción de las placas continentales, es en la región de la Sierra donde se han producido las mayores devastaciones, como es el caso de los terremotos acontecidos en las ciudades de Riobamba (1797) y Ambato (1949), ver Beauval et al. (2010).

En la Región Andina, las principales fuentes sísmicas son producto de la actividad volcánica, ver Lara Cueva et al. (2016), así como de las numerosas fallas geológicas existentes, ver Eguez et al. (2003). En particular, en la provincia del Azuay existen cuatro fallas geológicas (Paute, Girón, Gualaceo y Tarqui) y un historial de al menos cuatro sismos importantes. El más significativo ocurrió en 1887 y afectó considerablemente ciertas edificaciones de la ciudad de Cuenca. Dicho sismo fue de origen superficial y en una de las fallas geológicas cercanas, ver Jiménez et al. (2018).

Se sabe que los terremotos irradian energía en forma de ondas sísmicas. Las mismas se clasifican en ondas de cuerpo o internas y ondas superficiales. Las internas son aquellas que se propagan desde su origen hasta la superficie de la Tierra y se subdividen en ondas P (ondas primarias) y ondas S (ondas secundarias), como describe Otero (2018) y Sánchez et al. (2016). En cuanto a la captura de información, las estaciones sísmicas incorporan sensores especializados (v.g. acelerómetros, geófonos) para elaborar un registro continuo de datos. A su vez, la detección de las ondas P y S dentro de este registro permite determinar los eventos sísmicos, como se discute en Narverkar (2018).

En particular, el daño causado por un sismo depende del grado de preparación y de la capacidad de respuesta de la sociedad. Por lo tanto, la detección de eventos es fundamental para mitigar los detrimentos ocasionados por los sismos. Sin embargo, la detección automática de eventos sísmicos, en un registro continuo de datos de una red de estaciones sísmicas, es un desafío por el ruido sísmico que existe junto a las señales, como se describe en Sevilla (2019).

En este contexto, se revisan numerosos trabajos relacionados donde se emplean herramientas de *software.hardware* para la detección automática de eventos sísmicos, así como diversos estudios donde se comparan algoritmos y también se describen diferentes técnicas utilizadas en la detección de eventos en tiempo real.

En cuanto a la comparación de algoritmos, Sharma et al. (2010) discuten varios enfoques para detectar eventos sísmicos que presentan ruido de fondo. De igual forma, Vaezi y Van der Baan (2015) realizan la comparación entre el algoritmo STA/LTA (*Short-Time-Average Through Long-Time-Average Trigger*) y un método que emplea la Densidad Espectral de Potencia (PSD, *Power Spectral Density*) para la detección de microsismos. Los resultados del estudio indican que el mecanismo basado en la PSD permite detectar un mayor número de eventos que se pierden con el ruido de fondo. En cambio, Liu (2014), desarrolla un estudio donde además de comparar los métodos, se detallan un conjunto de criterios para el ajuste de parámetros característicos empleados en los algoritmos de detección.

Por otra parte, múltiples estudios proponen aplicar técnicas de Inteligencia Artificial para la detección de sismos. Por ejemplo, Mosher y Audet (2020) combinan un algoritmo de detección con una etapa de clasificación de eventos mediante redes neuronales. De forma similiar, Zhu et al. (2019), proponen un sistema clasificador basado en Redes Neuronales Convolucionales (CNN, *Convolutional Neural Networks)* diseñado para la detección e identificación de fase de una onda sísmica. En cuanto al análisis de datos, Rojas et al. (2019), proponen la aplicación de Redes Neuronales Artificiales (ANN, *Artificial Neural Network*) para la interpretación automática de datos sísmicos, con un enfoque especial en la detección de terremotos. Además, Ghorbani et al. (2018) describen un estudio que tiene como objetivo mejorar la autenticidad de los datos de eventos sísmicos. Esto se lleva a cabo mediante el uso de un método difuso y un algoritmo de red neuronal integrado, que involucra al perceptrón Multicapa *(MLP, Multilayer Perceptron*) y la red RBF (*Base Radial Function)* en forma de un sistema de aprendizaje colectivo con el fin de identificar eventos sísmicos a pequeña escala.

Adicionalmente, Reynen y Audet (2019) detallan un caso de estudio realizado en la región Sur de California, donde se aplican técnicas de aprendizaje automático para discriminar señales sísmicas. Los resultados indican que el sistema es capaz de distinguir entre explosiones y terremotos con una precisión del 99%. Por otro lado, Chamberlain y Townend (2018) exploran una extensión del método *matched-filter* para detectar terremotos de baja frecuencia en escalas locales a regionales. También existen nuevos algoritmos que utilizan como base el método STA/LTA. Uno de ellos es el STAFD/LTAFD (*Short-Time-Average Fractal Dimension Through Long-Time-Average Fractal Dimension*) desarrollado por Xhang et al. (2018). De forma similar, Jones y van der Baan (2015) presentan un algoritmo adaptativo que utiliza STA/LTA y los modelos ocultos de Markov (HMM). Otra implementación realizada por Choubik et al. (2020), utiliza el algoritmo STA/LTA y el modelo *Hadoop MapReduce*

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

(modelo de programación para dar soporte a la computación paralela sobre grandes colecciones de datos) para acelerar el proceso de detección al reducir el tiempo de procesamiento.

En este contexto, en el presente artículo se evalúan y comparan algoritmos para la detección automática de ondas sísmicas tipo P y S. En particular, se emplea un conjunto de algoritmos destacados en la literatura para el procesamiento de eventos sísmicos, tales como: Classic STA/LTA, Recursive STA/LTA, Delayed STA/LTA, Z-detector, ver Toledo (2014), Baer and Kradolfer picker, ver Baer y Kradolfer (1987) y AR-AIC (Autoregressive-Akaike-Information-Criterion-picker), ver Sleeman y Van Eck (1999). Esta comparación se realiza con respecto a la funcionalidad y configuración de los parámetros requeridos para cada algoritmo. Para llevar a cabo dicho análisis se diseña e implementa una herramienta de *software* denominada RSADE, la cual emplea la librería ObsPy. La herramienta implementada permite realizar varios experimentos por cada método utilizando registros sísmicos reales. Como resultado se obtiene que el algoritmo Classic STA/LTA presenta el mejor rendimiento, logrando un 98,2% de precisión. Cabe destacar que el *software* RSADE se encuentra disponible para uso de forma libre, ver Armijos (2021).

En cuanto a las herramientas de *software* disponibles para análisis sísmicos, existen diferentes propuestas. Por ejemplo, PICOSS es una plataforma gráfica en Python para la detección, segmentación y clasificación de datos sísmicos-volcánicos. El usuario puede seleccionar flujos de trabajo automáticos o manuales, incluidas las redes neuronales profundas como se explica en Bueno et al. (2020). También hay métodos analíticos en la ingeniería sísmica basados en el rendimiento (PBEE, Performance-based earthquake engineering) generalmente emplean conjuntos de registros de movimiento del suelo para obtener métricas de rendimiento estructural. En este contexto, EaRL, ver Elkady y Lignos (2020), es un *software* basado en Matlab que tiene como objetivo facilitar los cálculos PBEE proporcionando una interfaz gráfica, varias opciones de visualización de datos y además es de código abierto. De igual forma, R2R-EU es una herramienta PBEE que implementa numéricamente varios métodos para estimar la fragilidad sísmica específica de una estructura. El usuario puede elegir el método de análisis entre *Bootstrap*no paramétrico y el método delta, como se detalla en Baraschino et al. (2020). Finalmente, APASVO, ver Romero et al. (2020), es una herramienta grafica en Python de código abierto que permite la detección automática de la onda P y de eventos sísmicos. La aplicación utiliza los algoritmos STA/LTA, AMPA y AIC.

Al comparar las herramientas descritas anteriormente con el *software* implementado RSADE, cabe destacar que es de código

abierto en Python que tiene las funcionalidades de análisis en intervalos de tiempo, generación de gráficas de los resultados, exportación de los resultados de detección de eventos, creación de nuevos archivos miniSeed con los datos de los intervalos de interés y además permite elegir entre varios algoritmos de detección de eventos. Todas estas funcionalidades se desarrollaron en una plataforma SBC (Raspberry Pi 3), la cual es ampliamente utilizada en múltiples aplicaciones por sus capacidades de versatilidad, robustez, con prestaciones actuales y disponibilidad de diversos periféricos. En consecuencia, se implementa una herramienta portátil, escalable, de bajo costo y uso mínimo de CPU para optimizar el rendimiento del dispositivo SBC.

El artículo está organizado de la siguiente manera: en la Sección 2 se presenta la Metodología y Componentes, en la Sección 3 se presenta la Herramienta implementada y los Resultados obtenidos y finalmente en la Sección 4 las Conclusiones.

Metodología y componentes

En esta sección se detalla la metodología que se utilizó para la evaluación de los diferentes algoritmos de detección automática de eventos sísmicos. En la Figura 1 se presenta un diagrama de la metodología que se empleó para la comparación de cada algoritmo. Como se aprecia, en primera instancia, el sistema se alimenta con información de eventos sísmicos reales previamente obtenidos con estaciones acelerográficas de la Red Sísmica del Austro (RSA) adscrita a la Facultad de Ingeniería de la Universidad de Cuenca, Cuenca-Ecuador, en formato miniSeed. Se utilizaron datos en formato miniSeed para la comparación de los métodos (Armijos, 2021). Luego se realizó un preprocesamiento de estos datos, que consistió en una etapa de normalización mediante un factor de conversión y en una etapa de filtrado (filtro pasa banda) para eliminar el *offset* generado por los sensores y las frecuencias altas.

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

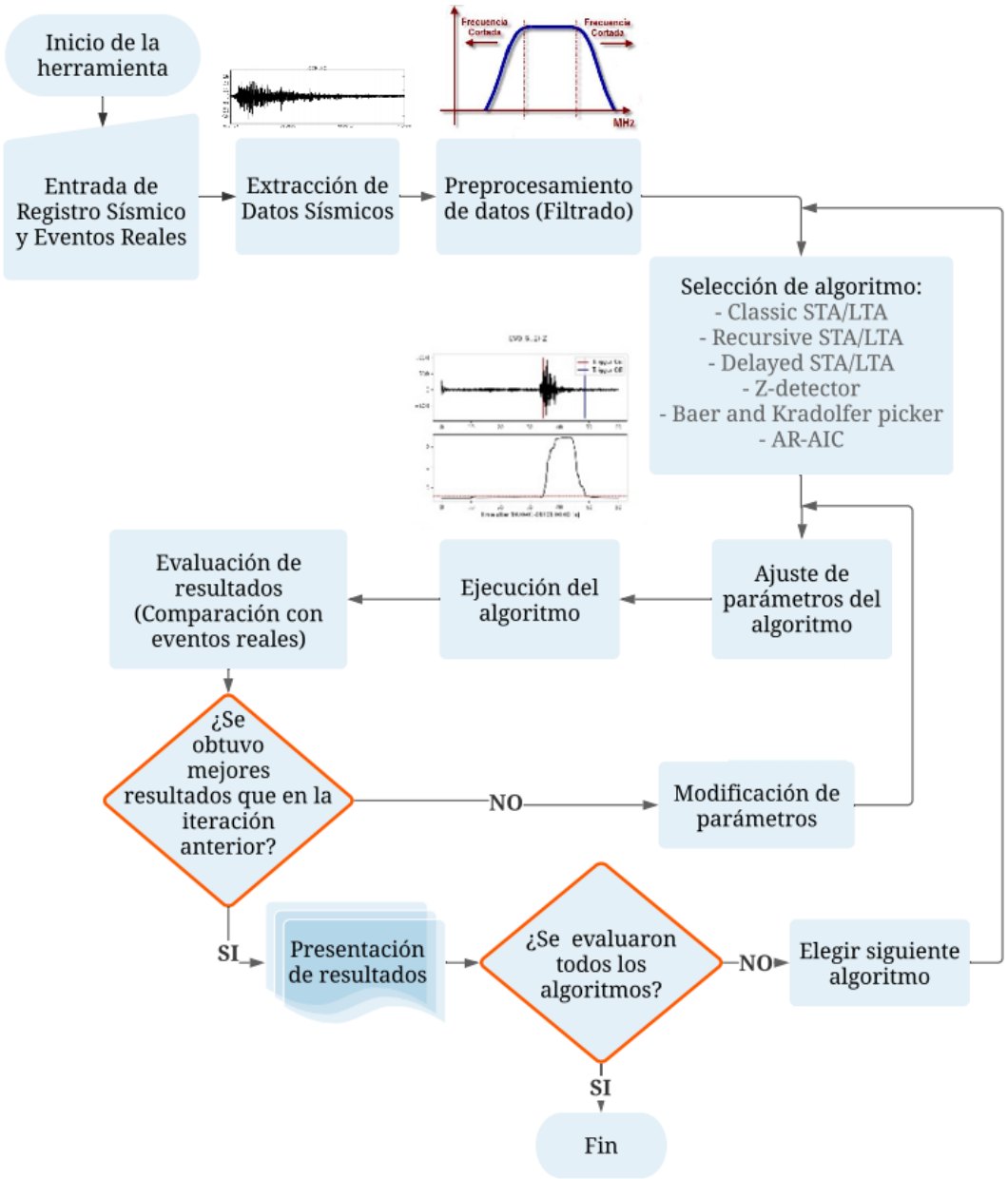


Figura 1

Metodología para el análisis y comparación de algoritmos de detección automática de eventos sísmicos

A continuación, en el procesamiento se utilizaron los diferentes métodos de detección disponibles en la librería ObsPy. Para cada método se configuraron distintos parámetros con valores iniciales y se obtuvieron los primeros resultados con registros de eventos sísmicos reales. Posteriormente, se ajustaron los parámetros, se logró nuevamente los resultados y se verificó si se consiguió mayor éxito en comparación a la iteración anterior. De esta manera, se repitió el

procedimiento de ajuste de parámetros hasta alcanzar los mejores valores para cada algoritmo. Además, para cada algoritmo se realizaron 10 pruebas, solo para el método AR-AIC se ejecutaron 8 experimentos. Para comparar los resultados obtenidos se empleó un conjunto de 58 registros con eventos sísmicos y 58 sin eventos, para evaluar el mejor resultado de cada algoritmo de detección y obtener la matriz de confusión de cada uno. En la matriz de confusión se tuvo los parámetros: Verdaderos Positivos (VP), Falsos Positivos (FP), Falsos Negativos (FN), que vendría a ser el ruido y Verdaderos Negativos (VN). Finalmente, como resultados se presentaron graficas con la información de los VP, FP y FN, además de la matriz de confusión para cada algoritmo.

Características de los métodos de detección

Los parámetros adecuados para cada algoritmo obedecen a diferentes factores, como por ejemplo el nivel de ruido promedio, la sensibilidad de los sensores sísmicos y la distancia a la que se encuentran las estaciones sísmicas. Por consiguiente, los valores adecuados de los parámetros difieren para cada método. No obstante, existen recomendaciones que se pueden tomar como referencia para el desarrollo de experimentos. En particular, los valores iniciales de los parámetros empleados en el presente trabajo se basan en las recomendaciones descritas por Jones y van der Bann (2015), así como en los ejemplos iniciales de *Trigger/Picker Tutorial.ObsPy Documentation* (2021).

En este sentido, según Trnkoczy (2009), el parámetro STA sirve como un filtro de señal, y cuanto menor sea la duración seleccionada mayor será la sensibilidad y viceversa. Específicamente, para el caso de eventos regionales, un valor típico de STA se encuentra entre 1 y 2 segundos y para eventos locales es más común emplear valores entre 0.3 y 0.5 segundos. Por otro lado, un valor de LTA corto puede mitigar los disparos falsos debido al ruido generado por fuentes artificiales (v.g. personas, vehículos). Un valor común para este parámetro fue de 60 segundos. Además, se debe considerar que los algoritmos disponibles en ObsPy reciben los parámetros NSTA (*Length of short time average window in samples*) y NLTA (*Length of long time average window in samples*), por lo que los valores recomendados se deben convertir de tiempo a número de muestras considerando la frecuencia de muestreo de los registros.

Por otra parte, el umbral de activación (*trigger*) determina en, mayor medida, qué eventos se registraron y cuáles no. Cuanto mayor es el valor que se establece, un mayor número de eventos no son registrados, pero se produce menos disparos falsos. Un valor típico inicial para el nivel de umbral de activación STA/LTA es de 4. De

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

similar manera, un nivel de umbral de desactivación (*detrigger*) STA/LTA demasiado bajo no es recomendado. Lo último puede ocasionar la detección de eventos sísmicos de tamaño indefinido. Un valor inicial típico del nivel de umbral de *detrigger* es de 2 a 3. A partir de las consideraciones descritas, así como de un considerable número de pruebas realizadas, a continuación, se indican los parámetros empleados para los experimentos con los distintos algoritmos.

Parámetros utilizados

Los experimentos realizados con diferentes parámetros se presentan en la Tabla 1, donde se observa que se realizaron 10 experimentos denominados desde P1 hasta P10. Cabe destacar que los parámetros de STA y LTA se encuentran en segundos.

En cuanto al algoritmo Classic STA/LTA, se comenzó con unos parámetros iniciales (P1) y se generaron los resultados comparando con los eventos reales. Luego, se modificaron los parámetros y se determinaron si los resultados mejoraron. Este procedimiento de ajuste de parámetros se repitió hasta obtener los mejores resultados. Por lo tanto, se realizaron más pruebas de las que se muestran en la Tabla 1, en este caso, solo se publicaron las más relevantes.

En el caso del método Recursive STA/LTA y Delayed STA/LTA se procedió de igual forma que con el algoritmo anterior. Se definió un valor inicial y se fue iterando a base de los resultados obtenidos. Sin embargo, como se mencionaba, estos valores no van a ser adecuados para todos los casos, sino que dependen de otros factores tales como el nivel de ruido promedio, sensibilidad de los sensores sísmicos y magnitud de los eventos.

El algoritmo Z Detector no recibe como entrada el parámetro LTA, como los métodos basados en STA/LTA, pero de igual forma se realizaron las pruebas variando los parámetros en cada una. Algo que se debe considerar en este algoritmo es el tiempo de procesamiento. En los algoritmos anteriores el tiempo de procesamiento promedio fue menor a 1 minuto. Pero en el método Z Detector, conforme aumenta el valor de STA, el tiempo en el que se ejecuta el algoritmo aumenta, llegando incluso a 30 minutos de procesamiento.

Tabla 1

Parámetros de los distintos algoritmos

ALGORITMO	PARÁMETRO	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Classic STA/LTA	Trigger on	1.5	1.3	1.2	1.15	1.1	1.15	1.15	1.15	1.15	5
	Trigger off	0.5	0.7	0.8	0.85	0.85	0.85	0.85	0.85	0.5	0.5
	Sta	5	5	50	50	500	40	40	40	10	0.5
	Lta	10	10	100	100	1000	70	200	500	70	70
Delayed STA/LTA	Trigger on	5	1	1	18	1.4	103	2	103	124	101
	Trigger off	10	5	20	30	1.5	100	3	100	121	100
	Sta	2.1	5.2	21	1.7	80	20	300	4	10	5
	Lta	2.3	5.5	23	2	90	2000	350	400	1200	500
Recursive STA/ LTA	Trigger on	1.5	1.15	2	1.3	1.2	1.1	1.1	1.1	1.1	1.1
	Trigger off	0.5	0.5	0.5	0.5	0.7	0.8	0.8	0.8	0.8	0.95
	Sta	8.5	100	1000	10	20	40	40	20	6	20
	Lta	14	200	3000	70	50	80	120	100	12	40
Z Detector	Trigger on	1	2	1	1	0.5	0.5	0.5	0.2	0.1	0.1
	Trigger off	0.5	1	0.5	0.5	0.1	0.1	0.1	0.1	0	0
	Sta	10	50	200	2	0.5	1	7	2	20	8
Baer-and Kradolfer-picker	Trigger on	1E+08	7	1E+08	1E+08	7	10	7	1E+08	10	20
	Trigger off	2E+06	12	2E+06	2E+06	12	12	12	2E+06	12	30
	Thr1	10	7	10	300	10	7	10	10	7	7
	Thr2	2	12	200	2	20	12	20	2	12	15
	Tdownmax	1	20	1	1	10	20	10	1	20	20
	Tupevent	640	60	640	640	2	60	97	640	60	60
	Preset len	640	100	640	640	384	100	100	6	200	100

AmeliCA PDF generado a partir de XML-JATS4RModelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

	P dur	6	100	6	6	100	100	100	6	200	100
AR-AIC	F1	1	1	1	1	1	1	1	1	-	-
	F2	20	20	20	20	8	20	20	20	-	-
	Lta p	1	2	10	100	1	8	8	8	-	-
	Sta p	0.1	0.2	1	10	0.1	2	2	2	-	-
	Lta s	4	8	40	400	4	6	6	6	-	-
	Sta s	1	2	10	100	1	3	3	3	-	-
	M_p	2	4	20	200	2	2	10	2	-	-
	M_s	8	16	80	800	8	8	15	8	-	-
	L_p	0.1	0.2	1	10	0.1	0.1	0.1	1	-	-
	L_s	0.2	0.4	2	20	0.2	0.2	0.2	3	-	-

Para los algoritmos Baer and Kradolfer picker y AR-AIC se tienen más parámetros que para los métodos basados en STA/LTA. Estos algoritmos también se probaron con valores disponibles en Trigger/Picker Tutorial—ObsPy Documentation (2021) y a base de los resultados obtenidos se modificaron los diferentes parámetros.

Evaluación y resultados

En esta sección se describe la herramienta de *software* RSADE implementada para la evaluación de algoritmos de detección de eventos sísmicos. Además, se presentan los resultados obtenidos con los diferentes algoritmos.

Software para la detección automática de eventos sísmicos

La herramienta para la evaluación de los algoritmos de detección de eventos se implementa en Python. Entre las funcionalidades incorporadas se encuentran las capacidades de graficar eventos en diferentes intervalos de tiempo, exportar un archivo miniSeed con los datos de los intervalos de tiempo seleccionados y además guardar todos los eventos obtenidos con hora de inicio y fin en un archivo de texto. La Figura 2 describe la estructura de la herramienta mostrando las clases del sistema, sus atributos y métodos. Por ejemplo el método graficar_eventos() de la clase Graficar es público y permite visualizar los registros sísmicos. El método obtener_eventos() permite guardar todos los eventos sísmicos del análisis, con hora de inicio y fin del sismo en un archivo de texto. El método obtener_miniSeed() permite segmentar el archivo de análisis dependiendo de la hora de inicio y fin seleccionada. Y el método seleccionar_metodo() permite elegir el algoritmo de detección . En la Figura 3 se presenta una captura del sistema RSADE donde se resaltan sus principales funcionalidades. Cabe indicar que el *software* se libera para su uso y experimentación. La funcionalidad y configuración en detalle de la herramienta se puede revisar en Armijos (2021).

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

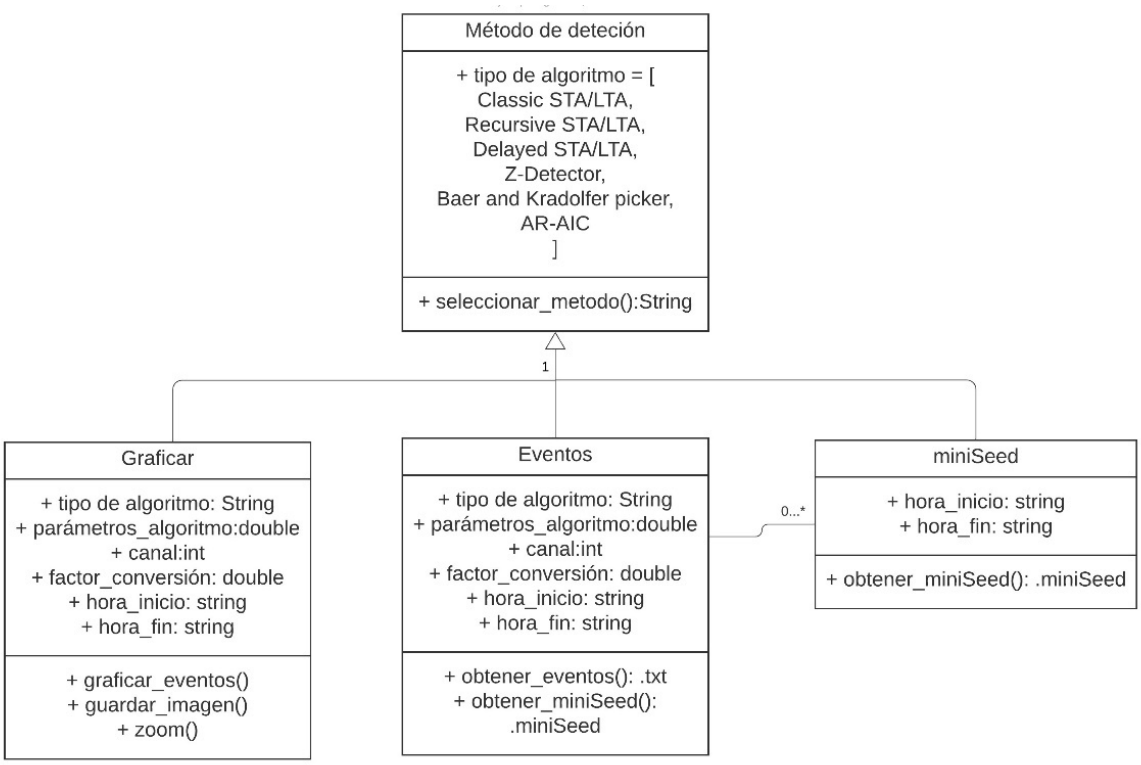


Figura 2
Diagrama de clases de la herramienta RSADE

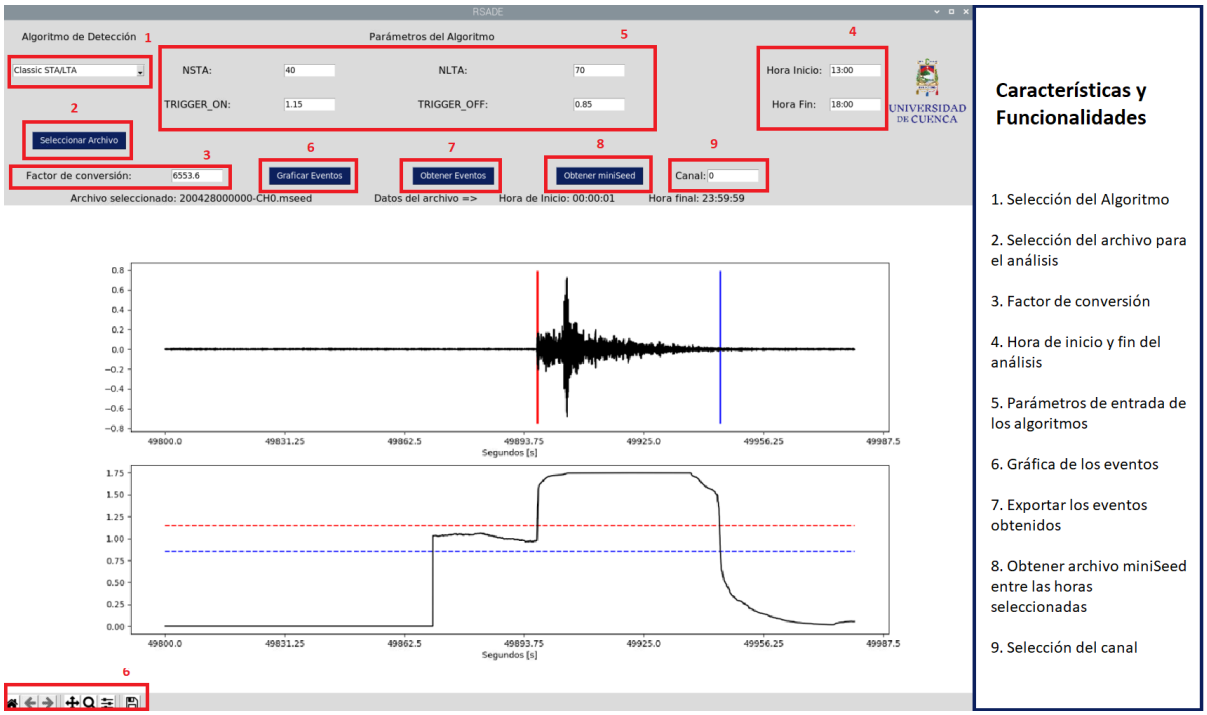


Figura 3

Características y Funcionalidades de la Herramienta RSADE

Los casos de uso de la herramienta se muestran en la Figura 4. Aquí la herramienta es el sistema y el actor el usuario. Para seleccionar el algoritmo únicamente interviene el usuario, pero para los otros casos de uso como graficar, obtener eventos y miniSeed también interviene la herramienta ya que es la que "prepara" todo para dar los resultados al usuario.

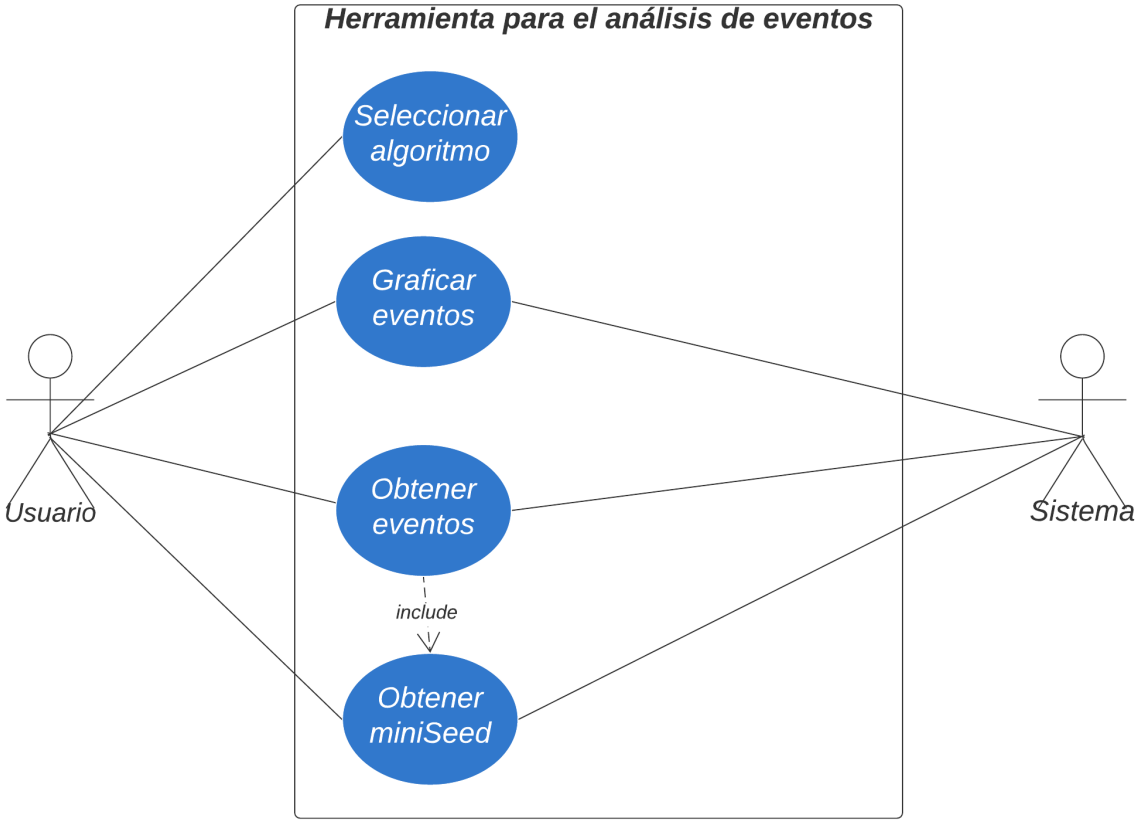


Figura 4
Casos de uso de la herramienta RSADE

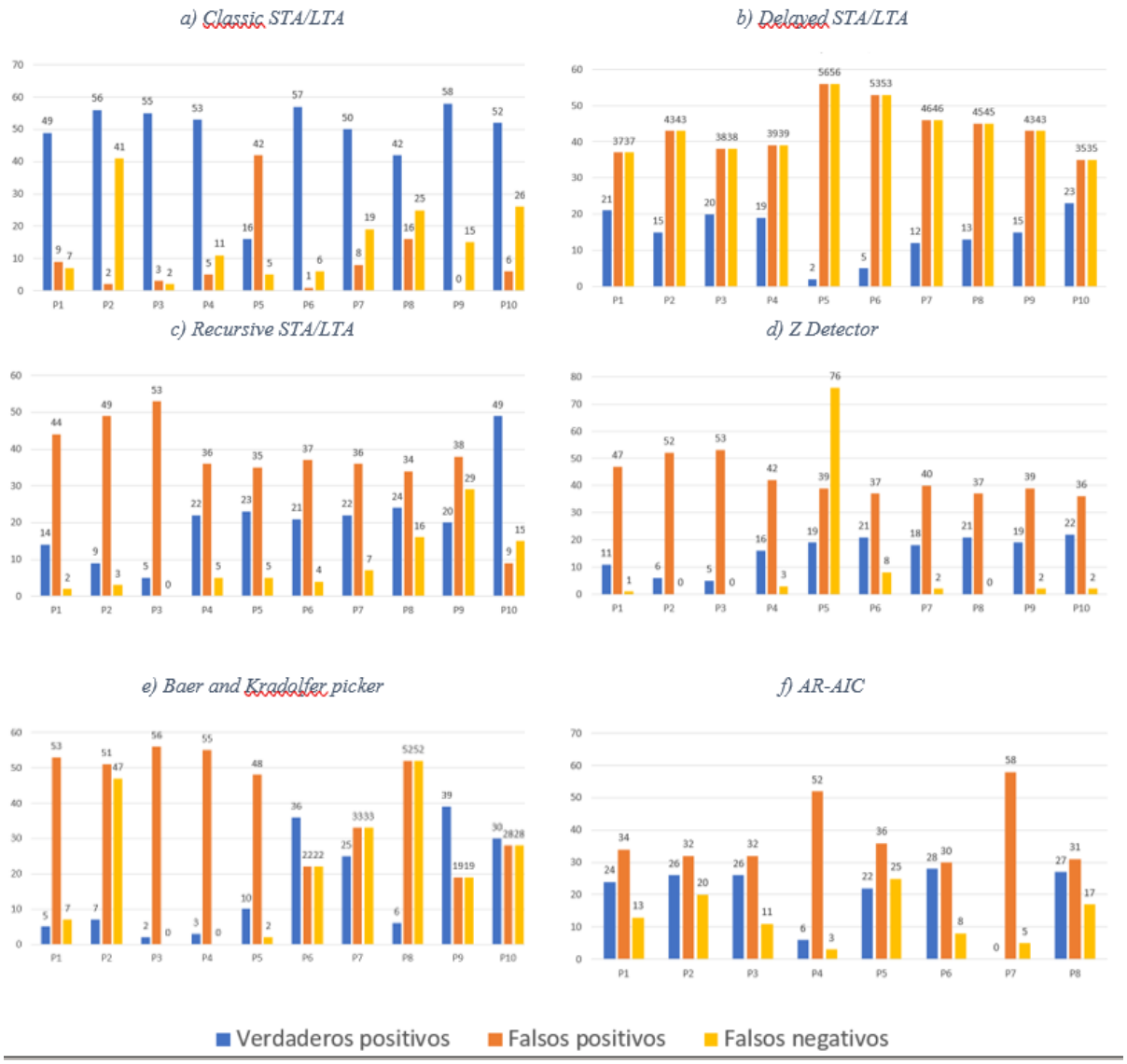


Figura 5
Resultados obtenidos con los diferentes algoritmos

Comparación de algoritmos de detección de eventos sísmicos

Para la comparación de los algoritmos se utiliza un conjunto de 58 registros con eventos sísmicos y 58 sin eventos. Los parámetros utilizados son: Verdaderos Positivos (VP) que es la predicción de un evento que corresponde a un evento real, Falsos Positivos (FP) que son los eventos que no se detectaron, Falsos Negativos (FN), que es el ruido y Verdaderos Negativos (VN). Luego, para evaluar los resultados se emplea la matriz de confusión que es una herramienta que permite visualizar el desempeño de un algoritmo. Una de las métricas de la matriz de confusión es la precisión, que viene a ser la

proporción de VP dividido entre todos los resultados positivos (tanto VP como FP).

Los resultados obtenidos en cada experimento para todos los algoritmos se presentan en la Figura 5. En primer lugar, los resultados del algoritmo Classic STA/LTA se muestran en la Figura 5a. Como se puede observar, con la prueba 9 (P9) se detectan todos los eventos reales y se tiene un total de 15 falsos negativos, lo cual no es conveniente. Con este análisis, la prueba que mejores resultados ofrece es la P6, en la cual solo un evento no se detecta, se obtienen 6 falsos positivos y se logra un total de 57 verdaderos positivos.

El algoritmo Delayed STA/LTA genera los resultados mostrados en la Figura 5b. El mejor resultado se obtiene en el experimento P10, donde se detectan un total de 58 eventos, de los cuales 23 son verdaderos positivos y 35 falsos positivos. En consecuencia, los resultados obtenidos con el método Delayed STA/LTA no son los mejores.

Con el último método basado en STA/LTA, el Recursive, se consiguieron los resultados de la Figura 5c. A base de los valores utilizados en P1 se ajustan los parámetros hasta mejorar el rendimiento. Se observa claramente que P10 fue la prueba con mejores resultados, ya que se lograron 64 eventos en total, de los cuales 49 fueron verdaderos positivos y 9 falsos positivos.

Continuando con la evaluación experimental, los resultados obtenidos con el algoritmo Z Detector se observan en la Figura 5d. En general, todas las pruebas tienen un porcentaje de eventos detectados bajo. La máxima cantidad de verdaderos positivos (22) de un total de 24 eventos obtenidos, fue en P10.

Para el algoritmo Baer and Kradolfer picker se generan los resultados de la Figura 5e. El mejor resultado con este método se obtiene en P9. En esta prueba se detectan 39 verdaderos positivos de un total de 58 eventos. Finalmente, para el algoritmo AR-AIC se puede observar que se realizan 8 pruebas, esto debido a que los resultados son parecidos a pesar de modificar los parámetros, a excepción de P4 y P7. El mejor resultado se obtiene en P6 (Figura 5f) ya que se logran 28 verdaderos positivos. Por lo tanto, se tiene 48 % de efectividad.

Selección de los mejores resultados

La matriz de confusión con la clasificación de evento y no evento para el mejor resultado de cada algoritmo se presenta en la Figura 6. De los 6 métodos comparados, el algoritmo Classic STA/LTA obtiene el mayor rendimiento, ya que como se muestra en la Figura 6a con los resultados de la matriz de confusión se tiene una precisión del 98,2%. Cabe destacar que el algoritmo Classic STA/LTA detecta

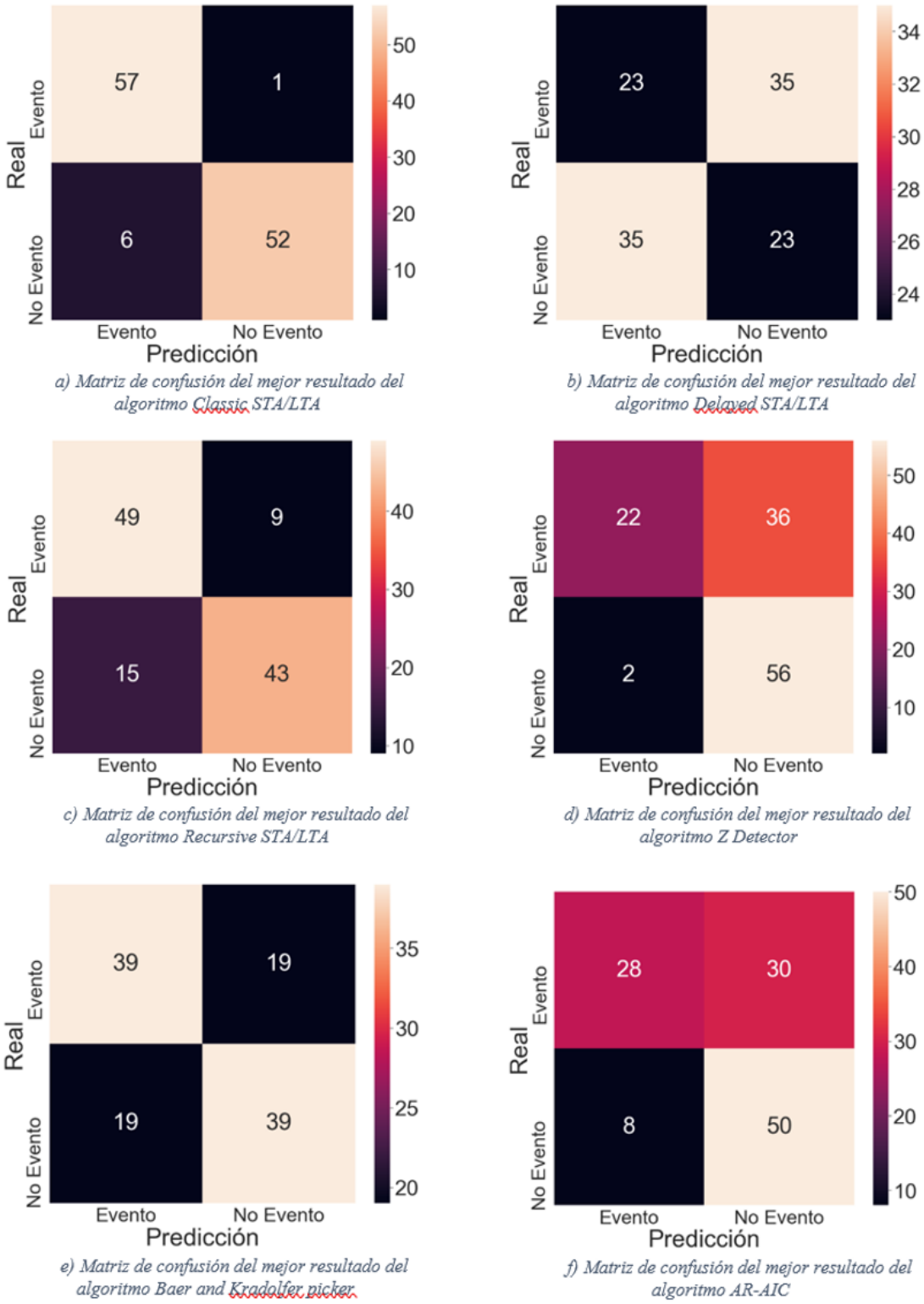
sismos comparando los niveles de movimiento del suelo a corto plazo con los niveles de movimiento del suelo a largo plazo, de esta manera se mejora significativamente el registro de sismos de baja magnitud en comparación con los otros algoritmos. También reduce el número de registros falsos provocados por el ruido sísmico natural y provocado por el hombre, como se detalla en Sevilla (2019).

Por otro lado, el método Recursive STA/LTA genera un 84,4% de precisión, como se observa en la Figura 6c. De igual forma en la Figura 6e, se describen los resultados para el método Baer and Kradolfer picker, el cual alcanza una precisión del 67,2%.

En cuanto a los métodos restantes, los resultados indican niveles inferiores de precisión. En concreto un 48.2% para el mecanismo AR-AIC (Figura 6f), 37.9% para el algoritmo Z Detector (Figura 6d). Finalmente, para el caso del método Delayed STA/LTA descrito en la Figura 6b, el nivel de precisión alcanza tan solo el 35.9%.

En la Figura 7 se presenta uno de los eventos sísmicos más relevantes, obtenido con el algoritmo Classic STA/LTA. En la parte superior de la figura se puede observar la llegada de la onda P (primaria) con la línea vertical de color rojo y el final de la onda S (secundaria) con la línea azul. En la parte inferior se muestra la función característica y el umbral *trigger/ detrigger* configurado.

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica



a) Matriz de confusión del mejor resultado del algoritmo Classic STA/LTA

b) Matriz de confusión del mejor resultado del algoritmo Delayed STA/LTA

c) Matriz de confusión del mejor resultado del algoritmo Recursive STA/LTA

d) Matriz de confusión del mejor resultado del algoritmo Z Detector

e) Matriz de confusión del mejor resultado del algoritmo Baer and Kradolfer picker

f) Matriz de confusión del mejor resultado del algoritmo AR-AIC

Figura 6
Mejores resultados de cada Algoritmo

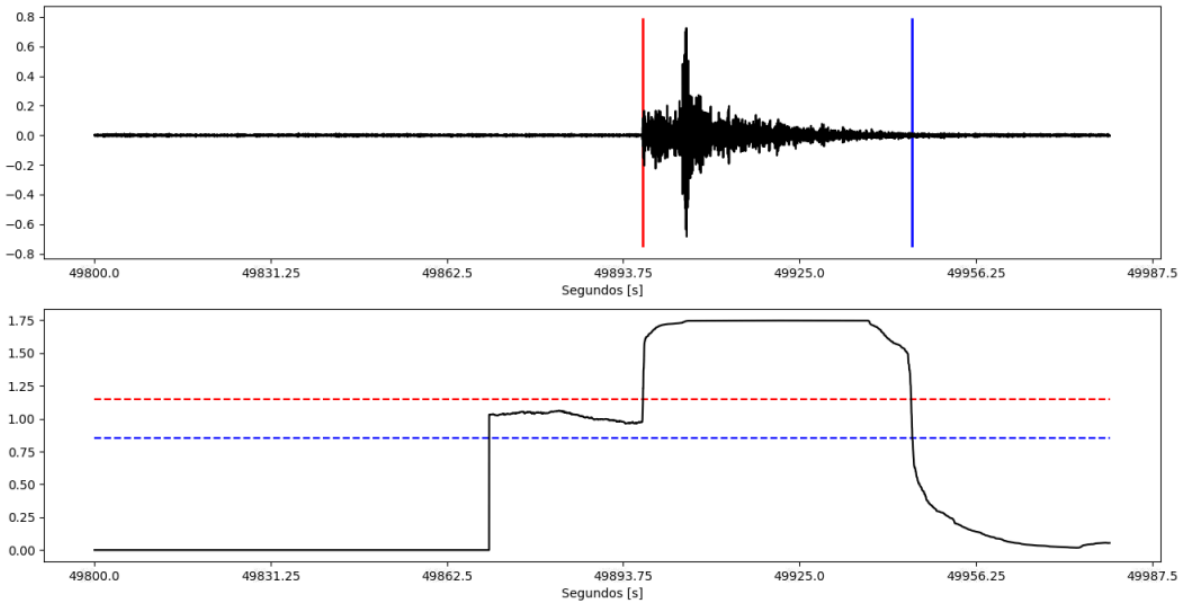


Figura 7
Evento sísmico obtenido con el Algoritmo Classic STA/LTA

Evaluación del rendimiento del dispositivo SBC

Como se describe previamente, la implementación de la herramienta de software y la comparación de los métodos se realizan sobre una plataforma SBC. En tal sentido, se diseñan experimentos para evaluar el rendimiento de la plataforma al ejecutar los diferentes algoritmos. Específicamente, se determina el uso de CPU del dispositivo SBC para cada método de detección como se muestra en la Figura 8.

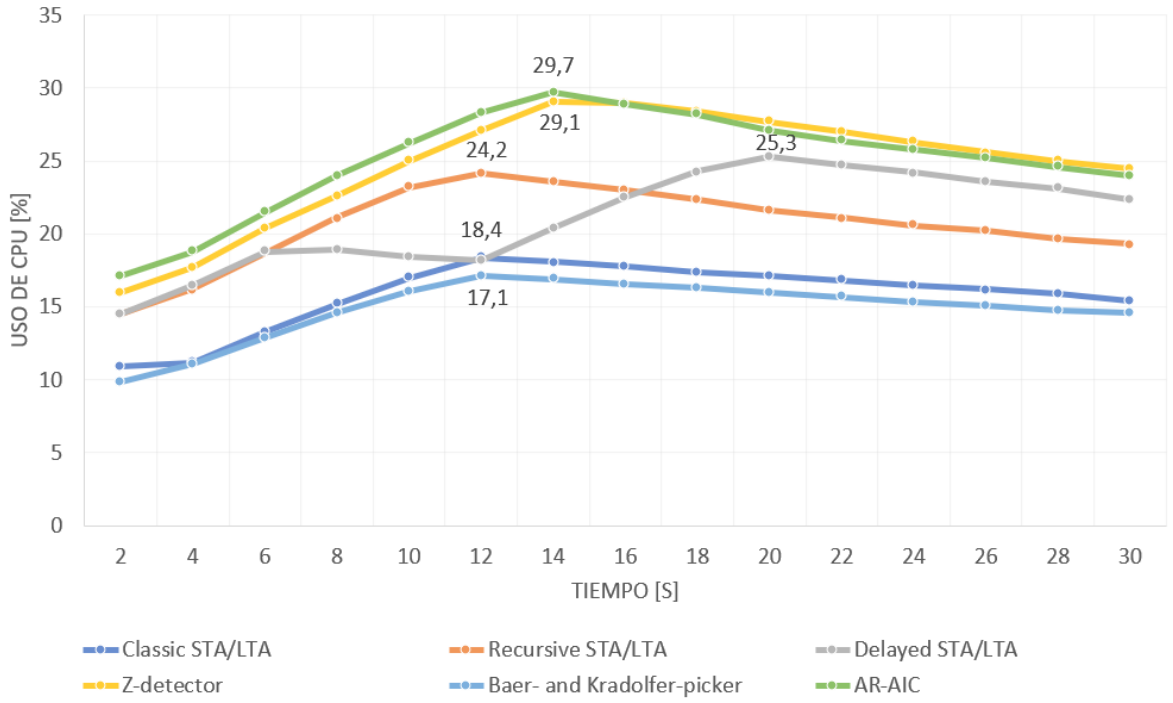


Figura 8
Uso de CPU por cada algoritmo

Con el objetivo de obtener estos resultados, se emplea un registro sísmico de 30 minutos para cada algoritmo. Además, se utilizan los mejores parámetros obtenidos en la subsección anterior y, finalmente, se captura el uso de la CPU de la Raspberry Pi durante 30 segundos ya que en promedio el tiempo de procesamiento de este registro es de 10 segundos. Como se puede apreciar en las curvas, los algoritmos que más costo computacional demandan son el Z Detector y AR-AIC. Mientras que el método Classic STA/LTA, que genera los mejores resultados, presenta un pico máximo de 18 % de uso de CPU.

En cuanto al método que menor costo computacional requiere es el de Baer and Kradolfer picker, en este caso el máximo valor obtenido es de 17.1 %. Finalmente, cabe destacar que en ningún caso se presentan sobrecargas en la CPU, por consiguiente, el uso de la plataforma SBC es adecuado para el estudio descrito.

Conclusiones

En este artículo se presenta un estudio de evaluación y comparación de algoritmos para la detección de eventos sísmicos. Con tal propósito se desarrolla una herramienta de *software* que permite leer un registro sísmico, detectar con diversos algoritmos los eventos sísmicos de dicho registro y la comparación de resultados de

forma rápida y sencilla. Además, dispone de las funcionalidades de análisis en intervalos de tiempo, obtener gráficas de los resultados, exportación de los resultados y creación de nuevos archivos miniSeed con los datos de los intervalos de interés. La decisión de liberar el *software* RSADE tiene la finalidad de que siga creciendo y mejorando con aportes de personas interesadas en este ámbito y que pueda ser usada para desarrollar más estudios sobre la detección de eventos sísmicos.

Con respecto a los experimentos realizados para cada algoritmo, se obtiene como resultado que el método Classic STA/LTA es el que mejores resultados genera, bajo las condiciones de los registros sísmicos de la Red Sísmica del Austro (nivel de ruido promedio, sensibilidad de los sensores sísmicos, distancia a la que se encuentran las estaciones sísmicas, magnitud). Con este algoritmo se obtiene una precisión del 98,2%. Por otro lado, para otros algoritmos como el Baer and Kradolfer picker se logra una baja tasa de detección debido a que varios de los eventos sísmicos tienen baja magnitud.

Adicionalmente, se realiza la evaluación del uso de CPU de la plataforma SBC. Como resultado se obtiene que los algoritmos que más costo computacional requieren son el Z Detector y AR-AIC, por lo que no serían óptimos para una implementación en tiempo real. Mientras que el método con menor uso de CPU es el de Baer and Kradolfer picker.

Como trabajos futuros se plantea aplicar técnicas de inteligencia artificial como Redes Neuronales o *Support Vector Machines* para la detección automática de eventos sísmicos. Como datos de entrada se plantea emplear registros sísmicos reales de varios institutos de sismología de forma similar a los utilizados en el presente trabajo. Además, con base en los resultados obtenidos, se va a utilizar el algoritmo Classic STA/LTA para implementar en tiempo real el sistema de detección automática de eventos sísmicos.

Cabe recalcar que al tener implementado el sistema en una Raspberry Pi 3 se puede replicar el *software* en diferentes dispositivos de forma rápida y económica, ya que al ser una plataforma de bajo costo permite que el sistema sea escalable, portable y fácil de manejar.

Reconocimientos

Este trabajo se enmarca dentro del Proyecto de Investigación Tecnologías IoT y Redes Inalámbricas de Sensores Aplicados a la Monitorización de Salud Estructural en Edificios Esenciales de la Ciudad de Cuenca y forma parte de la Tesis Evaluación y Comparación de Algoritmos para la Detección Automática de Ondas Sísmicas.

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

Los autores desean expresar su agradecimiento a la Dirección de Investigación de la Universidad de Cuenca (DIUC) y a la Red Sísmica del Austro (RSA), por el apoyo y facilidades brindadas para la consecución del presente trabajo.

Referencias

Akram, J., Peter, D., & Eaton, D. (2018). A k-mean characteristic function to improve STA/LTA detection. *Proceedings of the Geoconvention, Calgary, AB, Canada*, 7-11.

Armijos, A. (2021). *Alvaroarmijos/RSADE* [Python]. https://github.com/alvaroarmijos/RSADE (Original work published 2020)

Baer, M., & Kradolfer, U. (1987). An automatic phase picker for local and teleseismic events. *Bulletin of the Seismological Society of America, 77*(4), 1437-1445.

Baraschino, R., Baltzopoulos, G., & Iervolino, I. (2020). R2R-EU: Software for fragility fitting and evaluation of estimation uncertainty in seismic risk analysis. Soil Dynamics and Earthquake Engineering, 132, 106093. https://doi.org/10.1016/j.soildyn.2020.106093

Beauval, C., Yepes, H., Bakun, W. H., Egred, J., Alvarado, A., & Singaucho, J.-C. (2010). Locations and magnitudes of historical earthquakes in the Sierra of Ecuador (1587–1996). *Geophysical Journal International, 181*(3), 1613-1633.

Beauval, C., Yepes, H., Palacios, P., Segovia, M., Alvarado, A., Font, Y., Aguilar, J., Troncoso, L., & Vaca, S. (2013). An earthquake catalog for seismic hazard assessment in Ecuador. *Bulletin of the Seismological Society of America, 103*(2A), 773-786.

Bueno, A., Zuccarello, L., Díaz-Moreno, A., Woollam, J., Titos, M., Benítez, C., Álvarez, I., Prudencio, J., & De Angelis, S. (2020). PICOSS: Python Interface for the Classification of Seismic Signals. Computers & Geosciences, 142, 104531. https://doi.org/10.1016/j.cageo.2020.104531

Chamberlain, C. J., & Townend, J. (2018). Detecting real earthquakes using artificial earthquakes: On the use of synthetic waveforms in matched-filter earthquake detection. *Geophysical Research Letters, 45*(21), 11,641-11,649.

Choubik, Y., Mahmoudi, A., Himmi, M. M., & El Moudnib, L. (2020). STA/LTA trigger algorithm implementation on a seismological dataset using Hadoop MapReduce. *IAES International Journal of Artificial Intelligence, 9*(2), 269.

Eguez, A., Alvarado, A., Yepes, H., Machette, M. N., Costa, C., Dart, R. L., & Bradley, L. A. (2003). Database and map of Quaternary faults and folds of Ecuador and its offshore regions. *US Geological Survey Open-File Report, 3*, 289.

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

Elkady, A., & Lignos, D. G. (2020). EaRL—Software for Earthquake Risk, Loss and Lifecycle Analysis. SoftwareX, 12, 100607. https://doi.org/10.1016/j.softx.2020.100607

Ghorbani, S., Barari, M., & Hoseini, M. (2018). Presenting a new method to improve the detection of micro-seismic events. *Environmental monitoring and assessment, 190*(8), 1-13.

Jiménez, J., Cabrera, J., Sánchez, J., & Avilés, F. (2018). Vulnerabilidad sísmica del patrimonio edificado del Centro Histórico de la Ciudad de Cuenca: Lineamientos generales y avances del proyecto. *Maskana, 9*(1), 59-78.

Jones, J. P., & van der Baan, M. (2015). Adaptive STA–LTA with outlier statistics. *Bulletin of the Seismological Society of America, 105*(3), 1606-1618.

Küperkoch, L., Meier, T., & Diehl, T. (2012). Automated event and phase identification. En *New Manual of Seismological Observatory Practice 2 (NMSOP-2)* (pp. 1-52). Deutsches GeoForschungsZentrum GFZ.

Lara-Cueva, R. A., Benítez, D. S., Carrera, E. V., Ruiz, M., & Rojo-Álvarez, J. L. (2016). Automatic recognition of long period events from volcano tectonic earthquakes at cotopaxi volcano. *IEEE Transactions on Geoscience and Remote Sensing, 54*(9), 5247-5257.

Liu, H., & Zhang, J. Z. (2014). STA/LTA algorithm analysis and improvement of Microseismic signal automatic detection. *Progress in Geophysics, 29*(4), 1708-1714.

Mosher, S. G., & Audet, P. (2020). Automatic Detection and Location of Seismic Events From Time-Delay Projection Mapping and Neural Network Classification. *Journal of Geophysical Research: Solid Earth, 125*(10), e2020JB019426.

Narvekar, N. (2018). Distinguishing earthquakes and noise using random forest algorithm. Master's Projects. https://doi.org/10.31979/etd.6kxb-c9tu

Obspy. *12. Trigger/Picker Tutorial—ObsPy Documentation (1.2.0)*. (2021). Recuperado de https://docs.obspy.org/tutorial/code_snippets/trigger_tutorial.html

Otero, D. Z. (2018). ONDAS SÍSMICAS, su importancia para la geofísica y la humanidad. *Universidad Industrial de Santander, 3*(4).

Reynen, A., & Audet, P. (2017). Supervised machine learning on a network scale: Application to seismic event classification and detection. *Geophysical Journal International, 210*(3), 1394-1409.

Rojas, O., Otero, B., Alvarado, L., Mus, S., & Tous, R. (2019). Artificial neural networks as emerging tools for earthquake detection. *Computación y Sistemas, 23*(2), 335-350.

Romero, J. E., Titos, M., Bueno, Á., Álvarez, I., García, L., Torre, Á. de la, & Benítez, M. C. (2016). APASVO: A free software tool for automatic P-phase picking and event detection in seismic traces. Computers & Geosciences, 90, 213-220. https://doi.org/10.1016/j.cageo.2016.02.004

Sánchez, D., Howard, R., & Tobón, G. (2016). Trabajo 2 ONDAS SISMICAS. Recuperado 28 de agosto de 2021, de https://www.academia.edu/24984185/Trabajo_2_ONDAS_SISMICAS

Sevilla Echeverría, I. (2019). Improving automatic earthquake detection in the Krafla seismic network [Thesis]. https://skemman.is/handle/1946/34507

Sharma, B. K., Kumar, A., & Murthy, V. M. (2010). Evaluation of seismic events detection algorithms. *Journal of the Geological Society of India, 75*(3), 533-538.

Sleeman, R., & Van Eck, T. (1999). Robust automatic P-phase picking: An on-line implementation in the analysis of broadband seismogram recordings. *Physics of the earth and planetary interiors, 113*(1-4), 265-275.

Toledo Peña, P. A. (2014). Algoritmo de detección de ondas P invariante de escala: Caso de réplicas del sismo del 11 de marzo de 2010. http://repositorio.uchile.cl/handle/2250/131361

Trnkoczy, A. (2009). Understanding and parameter setting of STA/LTA trigger algorithm. En *New Manual of Seismological Observatory Practice (NMSOP)* (pp. 1-20). Deutsches GeoForschungsZentrum GFZ.

Vaezi, Y., & Van der Baan, M. (2015). Comparison of the STA/LTA and power spectral density methods for microseismic event detection. *Geophysical Supplements to the Monthly Notices of the Royal Astronomical Society, 203*(3), 1896-1908.

Zhang, J., Tang, Y., & Li, H. (2018). STA/LTA Fractal Dimension Algorithm of Detecting the P-Wave ArrivalSTA/LTA Fractal Dimension Algorithm of Detecting the P-Wave Arrival. *Bulletin of the Seismological Society of America, 108*(1), 230-237.

Zhu, L., Peng, Z., McClellan, J., Li, C., Yao, D., Li, Z., & Fang, L. (2019). Deep learning for seismic phase detection and picking in the

AmeliCA
PDF generado a partir de XML-JATS4R
Modelo de publicación sin fines de lucro para conservar la naturaleza académica y abierta de la comunicación científica

aftershock zone of 2008 Mw7. 9 Wenchuan Earthquake. *Physics of the Earth and Planetary Interiors, 293*, 106261.

AmeliCA

Disponible en:
https://portal.amelica.org/ameli/journal/
844/8445114004/8445114004.pdf

Cómo citar el artículo

Número completo

Más información del artículo

Página de la revista en portal.amelica.org

AmeliCA
Ciencia Abierta para el Bien Común

Álvaro Eduardo Armijos Sarango,
Iván Santiago Palacios Serrano,
Santiago Renán González Martínez

Evaluación y comparación de algoritmos para la detección automática de eventos sísmicos
Evaluation and comparison of algorithms for the automatic detection of seismic events

Revista Tecnológica ESPOL - RTE
vol. 33, núm. 2, Esp. p. 58 - 74, 2021
Escuela Superior Politécnica del Litoral, Ecuador
rte@espol.edu.ec

ISSN: 0257-1749
ISSN-E: 1390-3659

DOI: https://doi.org/10.37815/rte.v33n2.830

CC BY-NC 4.0 LEGAL CODE
Licencia Creative Commons Atribución-NoComercial 4.0 Internacional.