

Detección automática de primeros arribos usando redes neuronales artificiales en trazas terrestres reales del catálogo sísmico colombiano.

Juan José Mantilla Parra y Miguel Rodrigo Vargas Blanco

Trabajo de investigación para optar al título de Ingeniero Electrónico.

Director

Sergio Alberto Abreo Carrillo

Doctor en Ingeniería

Codirector

Ana Beatriz Ramírez Silva

Doctor en Ingeniería Eléctrica

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Ingeniería Electrónica Bucaramanga

2023

Tabla de Contenido

Introducción	12
1 Objetivos	16
1.1 Objetivo General	16
1.2 Objetivos Específicos	16
2 Marco de referencia	17
2.1 Contexto geofísico	17
2.1.1 Eventos sismológicos	18
2.2 Algoritmos actuales de detección	18
2.2.1 Algoritmo Short-Time Window/Long-Time Window	19
2.2.2 Aprendizaje automático	21
2.2.2.1 Redes neuronales Artificiales	21
2.3 PhaseNet	22
2.4 Reentrenamiento	24
2.4.1 Estrategia Ensemble	25
2.4.2 Algoritmos Multi Armed Bandit	25
2.4.2.1 Replicado de memoria usando algoritmos MAB	26
2.4.2.2 Optimización de pesos usando algoritmos MAB	27

DETECCIÓN AUTOMÁTICA DE PRIMEROS ARRIBOS.	3
2.5 Entornos de trabajo	28
2.5.1 Hardware	28
2.5.2 Software	28
2.5.2.1 Anaconda / Conda	29
2.5.2.2 TensorFlow	29
3 Conjuntos de datos	30
3.1 Conjunto de datos Catalogo Sísmico Colombiano	30
3.2 STanford EArthquake Dataset (STEAD)	35
4 Red convolucional para la detección de primeros arribos	39
4.1 Pre-procesamiento de las grabaciones sismológicas	42
4.2 Post-procesamiento de la respuesta de la red entrenada	44
5 Experimentación	46
5.1 Caracterización de la red	46
5.2 Diseño de experimentos	50
5.2.1 Comparación de resultados de experimentos.	52
5.2.1.1 Fase experimental 1	52
5.2.1.2 Fase experimental 2	55
5.2.1.3 Fase experimental 3	62
5.2.1.4 Fase experimental 4	66

DETECCIÓN AUTOMÁTICA DE PRIMEROS ARRIBOS.	4
5.3 Sistema resultado	69
6 Conclusiones	74
7 Recomendaciones	77
Referencias Bibliográficas	79
Apéndices	82

Lista de Figuras

Figura 1	Modelo simple de una red neuronal artificial	22
Figura 2	Clúster del Grupo CPS	28
Figura 3	Vista geográfica de la zona de interés	31
Figura 4	Tipos de grabaciones en el conjunto de datos usados para el entrenamiento	35
Figura 5	Evento de ruido	36
Figura 6	Ejemplo de los tres tipos de etiquetas de la base de datos STEAD.	37
Figura 7	Arquitectura CNN para la detección de primero arribos	41
Figura 8	Etapa de pre-procesamiento	43
Figura 9	Etapa de post-procesamiento de la respuesta de la red entrenada	45
Figura 10	Gráficas del comportamiento del experimento: Entrenamiento SGC	47
Figura 11	Gráficas del comportamiento del experimento: Entrenamiento SGC + STEAD	48
Figura 12	Diagramas de barras de la predicción para el modelo preentrenado	48
Figura 13	Diagramas de barras de la predicción del experimento: Entrenamiento SGC	49
Figura 14	Diagramas de barras de la predicción del experimento: Entrenamiento SGC + STEAD	50
Figura 15	Fase Experimental 1: Entrenamientos con Dropout 10%	53
Figura 16	Fase Experimental 1: Entrenamientos con Dropout 30%	54

Figura 17	Diagramas de barras Fase Experimental 1: Entrenamientos con Dropout 10%	55
Figura 18	Fase Experimental 1: Entrenamientos con Dropout 30%	56
Figura 19	Fase Experimental 2: Entrenamiento Entrada 1	59
Figura 20	Fase Experimental 2: Entrenamiento Entrada 2	59
Figura 21	Fase Experimental 2: Entrenamiento Entrada 3	59
Figura 22	Fase Experimental 2: Entrenamiento Entrada 4	60
Figura 23	Fase Experimental 2: Entrenamiento Entrada 5	60
Figura 24	Fase Experimental 2: Entrenamiento Entrada 6	61
Figura 25	Fase Experimental 2: Entrenamiento Entrada 7	61
Figura 26	Fase Experimental 3: Entrenamiento STEAD Entrada 0	63
Figura 27	Fase Experimental 3: Reentrenamiento Entrada 0	63
Figura 28	Fase Experimental 3: Entrenamiento STEAD Entrada 1	64
Figura 29	Fase Experimental 3: Reentrenamiento Entrada 1	65
Figura 30	Fase Experimental 3: Entrenamiento STEAD Entrada 2	65
Figura 31	Fase Experimental 3: Reentrenamiento Entrada 2	66
Figura 32	Fase Experimental 4: Entrenamiento Final Entrada 0	67
Figura 33	Fase Experimental 4: Entrenamiento Final Entrada 1	68
Figura 34	Fase Experimental 4: Entrenamiento Final Entrada 2	68
Figura 35	Arquitectura CNN final para la detección de primeros arribos	70
Figura 36	Ejemplo de predicción de la red neuronal final para datos del STEAD	71
Figura 37	Ejemplo de predicción de la red neuronal final para datos óptimos del SGC	72

Figura 38 Ejemplo de predicción de la red neuronal final para datos problema del SGC

73

Lista de Tablas

Tabla 1	Descripción de las columnas contenidas en cada DataFrame.	32
Tabla 2	Cantidad de datos de los diferentes grupos de datos después de aplicar los filtros	37
Tabla 3	Conjuntos de datos utilizados en esta fase de caracterización	46
Tabla 4	Experimentos realizados en la fase de caracterización	47
Tabla 5	Caracterización de la red	50
Tabla 6	Conjuntos de datos utilizados en esta fase experimental 1	52
Tabla 7	Experimentos realizados en la fase experimental 1	53
Tabla 8	Resultados Fase 1	55
Tabla 9	Conjuntos de datos utilizados en esta fase experimental 2	58
Tabla 10	Experimentos realizados en la fase experimental 2	58
Tabla 11	Resultados Fase 2	62
Tabla 12	Conjuntos de datos utilizados en esta fase experimental 3	62
Tabla 13	Experimentos realizados en la fase experimental 3	63
Tabla 14	Resultados Fase 3	66
Tabla 15	Conjuntos de datos utilizados en esta fase experimental 4	67
Tabla 16	Experimentos realizados en la fase experimental 4	67
Tabla 17	Resultados Fase 4	69

Lista de Apéndices

Apéndice A. Códigos de programación implementados.

Resumen

Título: Detección automática de primeros arribos usando redes neuronales artificiales en trazas terrestres reales del catálogo sísmico colombiano.

*

Autores: Juan José Mantilla Parra, Miguel Rodrigo Vargas Blanco. **

Palabras Clave: Redes Neuronales, Picado de primeros arribos, Geofísica, Trazas Sismológicas, TensorFlow, Keras.

Descripción: Este trabajo se basa en la detección de primeros arribos en trazas sismológicas registradas en el Catálogo Sísmico Colombiano utilizando redes neuronales artificiales. Dada la gran cantidad de eventos sísmicos que se generan y registran diariamente; y el largo proceso que implica el análisis manual de los eventos para poder determinar el primer arribo, se han desarrollado algoritmos y herramientas que permiten la extracción de información de estos eventos de manera automática. En este proyecto, se consideran las diferentes condiciones en las que se obtienen los datos y lo que implica la presencia de ruido en la traza, que dificulta la correcta detección del primer arribo. Adicionalmente, se realiza una descarga, organización y clasificación de eventos sismológicos reportados por el Servicio Geológico Colombiano y publicados en el Catálogo Sísmico Colombiano, para el entrenamiento, validación y evaluación de la red neuronal. La red neuronal utilizada tiene como entrada los canales del evento sismológico, un filtro pasa bajas y una serie de transformadas que luego son ingresadas a una arquitectura que fusiona dos estrategias dando como resultado una secuencia de capa convolucional 1D, maxpooling 1D, capa flatten y dos capas densas, en donde se adicionan capas de dropout. En este trabajo se evalúan los resultados de la predicción de la red para diferentes escenarios de entrenamiento, en los cuales se varían: tamaño de la entrada y salida de la red, número de transformadas Wavelet, diferentes tipos de etiquetas para identificar el primer arribo en los datos de entrenamiento, además de evaluar diferentes conjuntos de datos de tanto para entrenamiento como para reentrenamiento, que incluyen tanto trazas del Catálogo Sísmico Colombiano como del STanford EArthquake Dataset (STEAD). Como resultado, se obtiene un error promedio mínimo en el número de muestras en la predicción para el conjunto de datos crudos reales del Servicio Geológico Colombiano de 227 muestras, incluido el error intrínseco de los datos que provee el Catálogo Sísmico Colombiano como resultado de la precisión de la estampa de tiempo empleada para identificar el primer arribo en los eventos registrados.

* Trabajo de Grado.

** Facultad de Ingeniería Fisicomecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Sergio Alberto Abreo Carrillo. Codirectora: Ana Beatriz Ramírez Silva

Abstract

Title: Automatic detection of first arrivals using artificial neural networks on real land traces from the Catálogo Sísmico Colombiano. *

Authors: Juan José Mantilla Parra, Miguel Rodrigo Vargas Blanco. **

Keywords: Neural Networks, First Arrival Picks, Geophysics, TensorFlow, Keras.

Description: This work is based on detecting the first breaks in seismological traces recorded in the Colombian Seismic Catalogue (Catálogo Sísmico Colombiano CSC) using artificial neural networks. The large number of seismic events that are generated and recorded daily, coupled with the lengthy process involved in the manual analysis of the events to determine the first break, has led to the development of algorithms and tools that allow the extraction of information from these events automatically. This project considers the different conditions in which the data are obtained and the presence of noise in the trace, which makes the correct detection of the first break difficult. Additionally, a download, organization, and classification of seismological events reported by the Colombian Geological Service (Servicio Geológico Colombiano SGC) and published in the Catálogo Sísmico Colombiano is performed for the training, validation, and evaluation of the neural network. The neural network used has as input the channels of the seismological event, a low-pass filter, and a series of transforms that are later entered into an architecture that merges two strategies resulting in a sequence of 1D convolutional layer, 1D maxpooling, flatten layer and two dense layers, where dropout layers are added. This work evaluates the results of the network prediction for different training scenarios, in which we vary: the size of the input and the output of the network, the number of Wavelet transforms, and different types of labels to identify the first break in the training data, in addition to evaluating different data sets for both training and retraining, which include both traces of the CSC and the STanford EArthquake Dataset (STEAD). As a result, a minimum average error in the number of samples in the prediction is obtained for the actual raw data set of the SGC of 227 samples, including the intrinsic error of the data provided by the CSC as a result of the accuracy of the time stamp used to identify the first break in the recorded events.

* Undergrad Work

** Faculty of Engineering Physicomechanics. School of Electrical, Electronic and Telecommunications Engineering. Advisor: Sergio Alberto Abreo Carrillo; Co-Advisor: Ana Beatriz Ramirez Silva

Introducción

Para el año 2021, fueron reportados más de 22.000 eventos sísmicos dentro del territorio colombiano¹ y alrededor del mundo, se aproxima que diariamente 50 eventos sísmicos son lo suficientemente fuertes para ser percibidos por el hombre (eventos sísmicos con magnitudes mayores a 2,5 en la escala de Richter) y otros que sólo son captados por estaciones sismológicas (eventos sísmicos con magnitudes menores a 2,5 en la escala de Richter) (Mousavi *et al.*, 2019). Esta adquisición masiva de datos ha promovido el desarrollo de estrategias, algoritmos y herramientas que permiten la manipulación y extracción de información sobre un evento sísmico de manera automática.

El enfoque principal de este proyecto es facilitar el procesamiento de trazas sismológicas reportadas y publicadas por el Servicio Geológico Colombiano en el Catálogo Sísmico Colombiano (Servicio Geológico Colombiano, 2022) y detectar en ella el tiempo de llegada de ondas primarias (ondas P) de forma automática, haciendo uso de aprendizaje automático (machine learning o ML por sus siglas en inglés).

La detección de las ondas P presenta una serie de retos debidos, entre otros, al ruido cap-

¹ Catálogo Sísmico Colombiano. <https://sismo.sgc.gov.co/>. Consultado: 10 de septiembre de 2022.

tado en las estaciones sismológicas; dependiendo de la zona y la sensibilidad que estas estaciones posean, pueden captar perturbaciones provenientes de otras fuentes (como puede ser el tráfico, explosiones, derrumbes, entre otras) (Mousavi *et al.*, 2019). Se puede recurrir a la inspección visual de las trazas sismológicas por parte de un experto o se puede recurrir a diferentes métodos para realizar la detección, de los cuales se destacan los métodos de realizaciones de tiempo corto (STA) y tiempo largo (LTA), el AMPA (Romero *et al.*, 2016) o el método de la transformada wavelet continua (o CWT por sus siglas en inglés), siendo su principal objetivo mejorar el rendimiento de la predicción del primer arribo.

El proceso de inspección visual de las trazas sísmicas puede tomar entre un 20% y un 50% del tiempo total empleado para su análisis y gran parte de estos datos que han sido analizados se mantiene como un secreto de negocio (Tsai *et al.*, 2020). Por esta razón, este proyecto pretende detectar los tiempos de arribo de las ondas P y S de forma automática en las trazas sismológicas reportadas en el Catálogo Sísmico Colombiano, usando la base de datos STanford EArthquake Dataset (STEAD)² como punto de referencia de entrenamiento y validación de una red neuronal diseñada para ese fin.

La arquitectura propuesta para esta tarea corresponde a una modificación de las redes convolucionales con la adición de una serie de capas densas a la salida expuesta en Zambrano y Cacho-

² STEAD, corresponde a una base de datos creada por la universidad de Stanford con datos sísmicos adaptados para facilitar su uso en aplicaciones de ML

po (2022). Con el fin de aprovechar las diferentes características de los datos de interés, se evalúa el efecto de cambiar las características de entrada de la red para aprovechar los datos captados por las estaciones sismológicas en los tres ejes (dirección este-oeste, dirección norte-sur y dirección vertical). Adicionalmente, teniendo en cuenta que la salida de la red se diseñó como una etiqueta de la misma longitud de las trazas de entrada, se define un algoritmo de post-procesamiento para identificar la muestra que se estima más probable como correspondiente al primer arribo de la traza analizada.

En este trabajo se realizaron pruebas para evaluar cómo se comporta la predicción de la red utilizando dos conjuntos de datos de diferente origen, cambiando la cantidad de transformadas que se le ingresan a la red, cambiando la dimensión de entrada de la red, cambiando el tipo de etiqueta para el entrenamiento de la red y revisando cómo ésta se comporta al agregar un grupo de eventos de ruido.

El presente documento se organiza de la siguiente manera: Capítulo 1, objetivo general y objetivos específicos del trabajo. Capítulo 2, revisión previa de conceptos básicos de la geofísica, aprendizaje automático y detección de primeros arribos. Capítulo 3, se expone el proceso realizado para la creación del conjunto de datos del Servicio Geológico Colombiano, los retos que este implica y cómo se organiza el mismo, también se expone el conjunto de datos STEAD, cómo éste se organiza, qué contiene y los diferentes tipos de datos que son utilizados en este trabajo. Capítulo 4, en éste se presenta con detalle la arquitectura de red utilizada, los diferentes procesos que se hacen con los datos para ingresarlos a la misma y el proceso que se realiza con la predicción

de la red para encontrar la muestra del primer arribo en la traza del evento sísmológico. Capítulo 5, en este se muestran los experimentos realizados para mejorar la predicción de la red, se exponen los diferentes escenarios que se usaron y se comparan las métricas obtenidas para así encontrar la red solución que representa las mejores métricas. Capítulo 6, en este se presentan las conclusiones y los características de arquitectura de la red neuronal que tuvo el mejor resultado. Por último el capítulo 7, donde se encuentran las recomendaciones que se basan en lo que se evidenció en la fase experimental para así tenerlas en futuros trabajos.

1. Objetivos

1.1. Objetivo General

Diseñar un sistema basado en redes neuronales para la detección automática de primeros arribos en trazas sismológicas terrestres reales del catálogo sísmico colombiano.

1.2. Objetivos Específicos

- Preparar un conjunto de datos a partir de trazas sismológicas del catálogo sísmico colombiano para ser empleado en el entrenamiento y validación del modelo neuronal a proponer.
- Definir una arquitectura de red neuronal orientada a la detección de primeros arribos en trazas sismológicas.
- Proponer una estrategia de entrenamiento para la arquitectura de red previamente definida para la detección de primeros arribos
- Medir el desempeño de la red entrenada utilizando la base de datos anteriormente establecida.

2. Marco de referencia

En este capítulo se abordan: el contexto geofísico de los datos procesados, los algoritmos que se usan actualmente para la detección del primer arribo; las redes neuronales artificiales; las metodologías de reentrenamiento utilizadas: estrategia Ensemble y algoritmos Multi Armed Bandit MAB; y los entornos de trabajo empleados en este proyecto.

2.1. Contexto geofísico

La geofísica es una de las ramas principales de la ciencia que estudia la Tierra, en esta se aplican conceptos de la física para el estudio del comportamiento y las propiedades físicas del suelo. En esta se incluyen métodos para caracterizar el subsuelo, determinar condiciones físicas o determinar la historia evolutiva. Adicionalmente, esta ciencia usa métodos cuantitativos físicos que aprovechan eventos naturales como sismos para recolectar información, interpretar y predecir las características de nuestro planeta (Mexicano, sf; Krohn, 1984).

En el contexto sismológico se encuentran las ondas elásticas que son perturbaciones sobre un medio material que se propagan mediante un movimiento uniforme en el medio y la rapidez de esta depende de las propiedades físicas, como la densidad y la temperatura del medio material que se perturba. Este tipo de ondas, se pueden descomponer en una componente longitudinal en la

dirección de la propagación y en una onda transversal al movimiento, estas son catalogadas como la onda primaria (onda P) y la onda secundaria (onda S) respectivamente.

2.1.1. Eventos sismológicos

Los eventos sismológicos son movimientos inesperados al interior de la tierra, como puede ser el movimiento de las placas tectónicas, que se transmiten como ondas elásticas al rededor de un medio material. Las erupciones de volcanes, avalanchas, terremotos, pequeños sismos generan estos eventos que dependiendo a su magnitud son detectados por estaciones especializadas y, posteriormente, almacenados como trazas sismológicas en diferentes bases de datos.

Estas trazas son almacenadas generalmente en tres trazas diferentes que representan como se ve un evento específico en tres ejes que facilitan su análisis. Adicionalmente, estas trazas se guardan con información acerca del origen (distancia, latitud y longitud), un código de identificación, información sobre la magnitud, frecuencia en la cual fue registrado, información sobre la estación sismológica que hizo el registro, entre otros datos que pueden variar dependiendo la base de datos.

2.2. Algoritmos actuales de detección

La adquisición de datos sísmicos de manera digital se ha convertido en un problema, ya que, una sola estación sismológica operando de manera continua genera grandes cantidades de datos, que regularmente presentan un reto importante al momento de ser almacenados y procesados.

Esta problemática ha motivado el desarrollo de algoritmos de detección de eventos sísmicos, que permitan procesar en tiempo real (o lo más cercano a este) y que decidan si los datos captados son almacenados de forma permanente. A continuación, se discutirán los principales algoritmos de detección actuales implementados en las estaciones sismológicas encargadas de esta tarea.

2.2.1. Algoritmo Short-Time Window/Long-Time Window

Este algoritmo de disparo, se basa en un promedio de ventanas de corto y largo plazo (STA/LTA por sus siglas en inglés). Es el más usado en aplicaciones de sismología de débil movimiento. Este calcula de forma continua el promedio de las amplitudes absolutas de una señal sísmica con dos ventanas de tiempo en movimiento y trata de grabar la mayor cantidad de eventos posibles.

Este algoritmo requiere que se ajusten una serie de parámetros, siendo los más importantes la duración en tiempo de las ventanas, los valores para los cuales la relación entre estas dos inician y detienen la grabación de un posible evento sísmico y adicionalmente el ajuste de otras dos ventanas de tiempo. Estas dos últimas son las encargadas de almacenar información antes, ventana previa al evento sísmico (PEM por sus siglas en inglés), y después del evento sísmico, ventana post evento sísmico (PET por sus siglas en inglés).

La ventana de tiempo corto (STA) es sensible a los eventos sísmicos, mientras que la ventana de tiempo larga (LTA) provee información temporal sobre el ruido que rodea la estación sismológica. Cuando la relación entre estas dos ventanas de tiempo excede un valor predeterminado,

se declara que un evento ha ocurrido y su información es grabada y almacenada.

Generalmente la STA mide el valor considerado como instantáneo de la señal o de su envolvente, en la mayoría de los casos su duración es ajustada para que sea más pequeña en tiempo que los eventos que se esperan capturar, donde se suelen utilizar ventanas de tiempo entre 0.3 y 2 segundos. Por otra parte, la LTA es la encargada de medir el promedio de amplitudes del ruido captado, donde se debe ajustar para que sea unos periodos más larga que las fluctuaciones irregulares de ruido sísmico o puede ser continuamente ajustada.

También es necesario ajustar los valores para los cuales este algoritmo inicia la grabación del evento y en cuales se detiene. Entre más bajo sea el límite para la cual el algoritmo activa la grabación, este será más sensible, por ende puede generar la grabación de falsos positivos o por otra parte, si dicho valor es muy elevado, también es posible que una buena parte de eventos sísmicos no sean grabados.

En cuanto a la ventana PEM, esta se selecciona procurando que la grabación de dicho evento incluya todas las fases del sismo y en cuanto a la ventana PET, también es ajustada para captar la llegada de las ondas coda las cuales su amplitud suele ser mucho menor a las causantes de la activación de la grabación y marcan el final del evento sísmico.(Musson, 2012)

2.2.2. Aprendizaje automático

El aprendizaje automático (o Machine Learning, ML por sus siglas en inglés), es una rama de la ciencia en la cual consiste darle la habilidad a un sistema o computador la habilidad de aprender características de un conjunto de datos, sin la necesidad de generar reglas netamente explícitas. Se busca mejorar el rendimiento de una tarea a partir de la experiencia al desarrollar esta tarea (Geron, 2017).

Este tipo de algoritmos permiten dar soluciones a problemas complejos y en constante cambio, los cuales requerirían generar grandes listas de reglas de forma manual para llegar a una solución esperada. La principal debilidad de abordar un problema con estas técnicas se centra en la cantidad y la calidad de datos disponibles. Para hacer posible una detección de patrones, muchas veces es necesario cientos o miles de datos de ejemplo para la puesta a punto de un algoritmo basado en ML.

2.2.2.1. Redes neuronales Artificiales. Las redes neuronales artificiales (o Artificial Neural Networks, ANN por sus siglas en inglés), son una de las propuestas más usadas e implementadas en los acercamientos de ML. Su funcionamiento consiste en una abstracción simple del sistema nervioso central, donde las unidades más básicas de su funcionamiento son denominadas neuronas y son organizadas en distintas capas. Una red neuronal suele tener principalmente tres partes: Una capa de entrada, encargada de recibir y presentar los datos; una o múltiples capas ocultas, en su mayoría encargadas de la extracción y detección de características; y una capa de salida,

encargada de presentar el resultado obtenido por esta red. 1

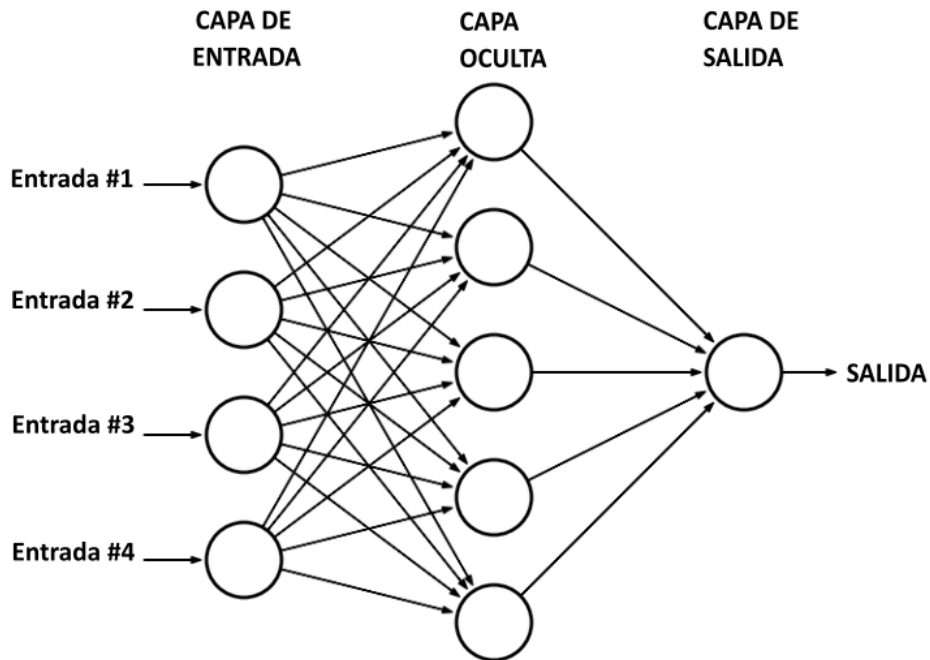


Figura 1. Modelo simple de una red neuronal artificial.

Nota: En la figura se observa una abstracción de un modelo simple de una Red Neuronal Artificial (ANN por sus siglas en inglés). Cada círculo representa una neurona y cada conjunto de neuronas se denomina capa. Se observan 3 capas: la capa de entrada, a la cual se le entrega el conjunto de datos; una capa oculta (puede haber más de una capa); y una capa de salida, que para este caso está representada con una sola neurona y ésta nos entrega la respectiva salida.

2.3. PhaseNet

PhaseNet corresponde a una arquitectura de red propuesta para la detección del momento de llegada de la onda P, la onda S o detección de ruido. Su arquitectura está basada en unas modificaciones de las redes U-Net. Esta arquitectura de red recibe como entrada las tres formas de onda sin ningún proceso de filtrado y esta nos entrega la probabilidad de la ubicación de la onda P, la onda S o ruido.

PhaseNet es una modificación de U-Net (Ronneberger *et al.*, 2015), que trabaja con series de datos de tiempo 1-D, mientras que las redes U-Net son un acercamiento de redes neuronales usado en el procesamiento de imágenes biomédicas, que busca localizar propiedades en una imagen; La entrada de la arquitectura PhaseNet se componen de los 3 componentes de una grabación de un evento sismológico y la salida son las distribuciones de probabilidad de las clases ondas P, ondas S y ruido. El autor busca localizar propiedades en las tres series de tiempo y calcular lo probabilidad de las tres clases: picados P, picados S y ruido. (Zhu y Beroza, 2019).

En sus experimentos, se presentaron ventanas de tiempo de 30s, muestreadas una frecuencia de 100 Hz. Las entradas pasan por 4 etapas de submuestreo (down-sampling) y 4 etapas de sobre muestreo (up-sampling), cada etapa con convoluciones 1-D y función de activación ReLU (Zhu y Beroza, 2019).

Las etapas de convoluciones 1-D tienen stride de 7 muestras, la etapa de down-sampling de 4 muestras, mientras la etapa de up-sampling expande a 4 muestras, para en cada etapa ir recuperando el tamaño y como salida una función softmax, en cargada de calcular la probabilidad de cada clase. Cada etapa tiene skip conexiones que concatenan las capas del lado izquierdo, con las del lado derecho, con el fin de ayudar en la divergencia durante el entrenamiento (Zhu y Beroza, 2019).

2.4. Reentrenamiento

El proceso de reentrenamiento de una Red Neuronal es importante cuando se le quieren presentar datos nuevos al modelo ya entrenado rescatando lo aprendido en el entrenamiento para así generar un modelo más robusto que el original frente a diferentes situaciones o características en los datos. El proceso consiste en utilizar el modelo preentrenado como punto de partida para realizar un entrenamiento en el que solo se le presenta el conjuntos de datos nuevos.

En este proceso se pretende evitar que el modelo original olvide las características del primer conjunto de datos presentado, normalmente para evitar este problema, el modelo es reentrenado utilizando todos los datos antiguos con los que fue entrenado al principio y agregando a estos los nuevos que no han sido presentados a la red, lo que representa un aumento en la cantidad de datos que se le presentan a la red haciendo de este método algo inviable debido al costo computacional que representa entrenar con todo el conjunto de datos al mismo tiempo (Klabjan y Zhu, 2020).

Para el proceso se puede optar por presentarle los datos recientes u optar por otro tipo de aproximación al proceso. Asimismo, existen otras estrategias que representan un aumento en la eficiencia del entrenamiento y que sirven para prevenir que la red olvide lo que aprendió con los datos antiguos y producir un modelo más efectivo que el original, de estas podemos rescatar las estrategias de Ensemble o los algoritmos Multi Armed Bandit.

2.4.1. Estrategia Ensemble

La estrategia de reentrenamiento Ensemble se basa en el uso de varios modelos entrenados para realizar una predicción final. Esta aproximación utiliza diversos conjuntos de datos para así generar diferentes modelos que pueden o no contar con la misma Red Neuronal. Luego, estos se utilizan para realizar predicciones individuales de un grupo específico de datos y dichas salidas luego se promedian para estimar un resultado final (Brownlee, 2021).

Esta estrategia permite utilizar diferentes tipos de datos, además de poder realizar actualizaciones al modelo sin perder lo que este había extraído de la información que se le había suministrado antiguamente. Adicionalmente, si se utiliza con diferentes algoritmos o arquitecturas de red se pueden generar modelos que logren extraer distintos parámetros clave de un mismo conjunto de datos, llegando a mejorar la precisión de en la predicción final (Education, 2021).

2.4.2. Algoritmos Multi Armed Bandit

Los algoritmos Multi Armed Bandit (MAB, por sus siglas en inglés), son los encargados de tomar decisiones a lo largo del tiempo bajo incertidumbre (Slivkins, 2019). Estos algoritmos tienen K posibles acciones para escoger, a estas acciones se les denomina brazos (arms, término en inglés) y un número T de rondas. En cada ronda t , se escoge un brazo y se evalúa la recompensa obtenida.

En un funcionamiento básico de estos algoritmos se observa únicamente la recompensa del

brazo escogido en una ronda t , pero los otros brazos no pueden ser escogidos. Por lo tanto, se hace una exploración para probar diferentes brazos y de esta forma adquirir nueva información.

El termino Multi Armed Bandit proviene de una analogía asociada con los juegos de azar y las maquinas traga monedas, la cual consiste en realizar la misma jugada en diferentes maquinas traga monedas (denominadas brazos) y al mismo tiempo y observar cual de todas estas nos arroja la mejor ganancia (o recompensa).

Este tipo de algoritmos proveen estrategias para el reentrenamiento de redes neuronales artificiales, actuando como un sistema de recompensas, de las cuales destacamos principalmente dos estrategias de reentrenamiento de redes ANN, descritas a continuación.

2.4.2.1. Replicado de memoria usando algoritmos MAB. Esta técnica consiste en seleccionar mini-batches óptimos en la sección de entrenamiento $m+1$ para la sección de entrenamiento $m+2$ y se crea un sistema de recompensas basado en las actualizaciones de las perdidas para las diferentes muestras del entrenamiento. Se considera que cada mini-batch usado en el entrenamiento como un brazo y cada actualización del gradiente como una acción para conocer la recompensa (Klabjan y Zhu, 2020).

La sesión de reentrenamiento consiste en seleccionar un subset de datos o muestras usados en el entrenamiento actual, que posteriormente serán usado en el subset de datos de la siguiente sesión de entrenamiento. Formalmente mientras el entrenamiento en $D^m \cup D^{m+1}$. D^m , denota el conjunto de datos viejos y D^{m+1} denota el conjunto de datos nuevo. El objetivo es seleccionar un

subset S e igualar $D^{m+1} = S$ para el siguiente entrenamiento en $D^{m+1} \cup D^{m+2}$. La idea principal consiste en realizar un entrenamiento sobre un número de épocas con una técnica de optimización y luego hacer un cambio a una estrategia que haga uso de un mini-batch en cada paso de entrenamiento basado en MAB, y en cada paso se registra cual mini-batch ha sido seleccionado haciendo uso de algoritmos MAB. (Klabjan y Zhu, 2020)

2.4.2.2. Optimización de pesos usando algoritmos MAB. Esta estrategia consiste en realizar la actualización de una agrupación (cluster, termino en inglés) de pesos/neuronas, las cuales con respecto al entrenamiento muestran tener una correlación entre las mismas. Es decir, el cluster seleccionado de pesos debe converger en sincronía y se juntan para ser reentrenados (Klabjan y Zhu, 2020)

Inicialmente, se itera sobre cada cluster y únicamente se actualiza el que fue seleccionado sobre una sola época (para esta sesión de entrenamiento se usan todos los datos). Se registra el cambio en la función de pérdida para cada cluster (en este caso, cada uno representa un brazo), seguido de esto, se restablecen los valores iniciales de cada uno. (Klabjan y Zhu, 2020)

Luego de este proceso de inicialización, los mini-batch se procesan de la manera habitual basada en épocas y para cada uno se selección un cluster (o brazo) de pesos usando algoritmos MAB, se registra la pérdida obtenida al optimizar únicamente el grupo de pesos seleccionado, mientras que por otra parte, el resto de grupos no son actualizados (Klabjan y Zhu, 2020)

2.5. Entornos de trabajo

Para el desarrollo de pruebas e implementaciones hechas durante este proyecto, se emplearon herramientas de hardware y software descritas a continuación.

2.5.1. Hardware

Las tareas desarrolladas para la organización del conjunto de datos usados durante el desarrollo de este proyecto, junto con las tareas de entrenamiento y reentrenamiento de modelos de red propuestos y generados, se realizaron en el clúster del grupo de investigación en Conectividad y Procesamiento de Señales (CPS), el cual cuenta con las características mostradas en la figura 2

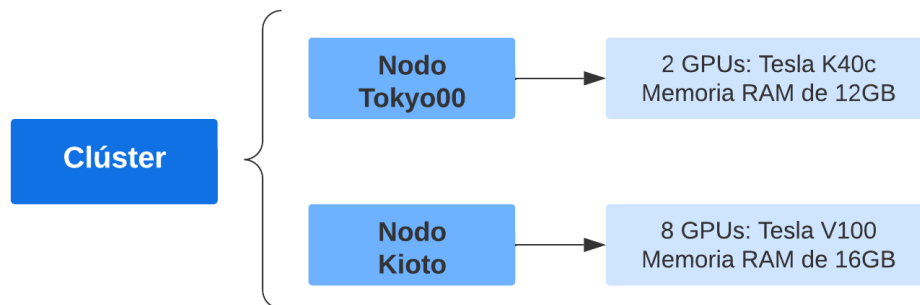


Figura 2. Clúster del Grupo CPS.

Nota: En la figura se muestra la organización del clúster del grupo de investigación en Conectividad y Procesamiento de Señales (CPS) y las unidades de procesamiento gráfico (GPU) disponibles en cada nodo.

2.5.2. Software

A continuación, se entrará en detalle de las herramientas de software y librerías o módulos del lenguaje de programación Python usados durante el desarrollo de este proyecto.

2.5.2.1. Anaconda / Conda. Esta herramienta consiste en un potente administrador de paquetes y entornos de programación para el lenguaje Python, desarrollado para Windows, macOS o Linux. Esta herramienta permite crear entornos únicos y separados, con versiones de módulos y librerías específicas dependiendo de la tarea que se quiera realizar. Anaconda (2022)

2.5.2.2. TensorFlow. Es una plataforma de código libre enfocada en aplicaciones de ML, que presenta un ecosistema, herramientas y librerías flexibles para el desarrollo e implementación de redes neuronales y ML en grandes conjuntos de datos. Permite hacer uso de unidades de procesamiento gráfico (GPU's por sus siglas en inglés), para acelerar los tiempos de entrenamiento y tiempos de inferencias. Abadi *et al.* (2015)

3. Conjuntos de datos

3.1. Conjunto de datos Catalogo Sísmico Colombiano

La creación de un conjunto de datos provenientes del Catalogo Sísmico Colombiano (CSC), corresponde a uno de los objetivos principales del desarrollo de este proyecto; este conjunto de datos está conformado por información de eventos sísmológicos del valle medio del Magdalena, delimitados por las coordenadas en longitud desde -74° a -72° y en latitud desde 6° a 7.5° , que fueron ajustados para emplearlos en entrenamientos, predicciones y validaciones de problemas que hagan uso de Machine Learning (ML). La Figura 3. presenta una vista geográfica de la zona de interés mencionada previamente.

Los eventos sísmicos registrados por el Servicio Geológico Colombiano (SGC) y publicados en el Catálogo Sísmico Colombiano³ de forma masiva fueron consultados con los parámetros de filtrado necesarios (región de interés, magnitudes y profundidad), permitiendo condensar la búsqueda de los eventos sísmicos en un archivo Excel. Para la construcción del conjunto de datos se

³ Enlace de acceso al Catálogo Sísmico Colombiano: <http://bdrsnc.sgc.gov.co/paginas1/catalogo/index.php>

La consulta de datos se limitó a eventos surgidos a partir del 1 de marzo de 2018 hasta el 30 de abril de 2022, con magnitudes máximas de 9 y un rango de profundidad máxima de 200 km⁵. Todas las grabaciones obtenidas fueron filtradas, teniendo en cuenta sus metadatos (ver Tabla 1), con estampas de tiempo registradas por el Servicio Geológico Colombiano y suministradas al grupo de investigación CPS en un archivo CSV.

Tabla 1

Descripción de las columnas contenidas en cada DataFrame.

Columna	Descripcion	Ejemplo
Network	Red Sísmica Colombiana	CM
Station	Estación que captó el evento	BAR2, VMM09
LatStation	Latitud de la estación sísmológica	6.77, 7.36
LongStation	Longitud de la estación sísmológica	-73.72, 73.85
Location	Código de locación	00
Channel	Canal que percibió la traza	HHE, HHN ó HHZ
Starttime	Tiempo de inicio de la traza sísmica	2018-03-03T03:43:01.840000Z
Endtime	Tiempo de fin de la traza sísmica	2018-03-03T03:54:48.410000Z
Samplingrate	Tasa de muestreo	100Hz
Delta	Distancia de muestra en segundos	0.01
Ntps	Numero de muestras tomadas	70658,67890
Eventime	Momento en el que sucede el evento sísmico (String)	2018-03-03T03:44:19
DepthEvent	Profundidad del evento sísmico en Km	102.44
LongEvent	Longitud estimada de donde sucedió el evento sísmico	-73.66
LatEvent	Latitud estimada de donde sucedió el evento sísmico	6.426
MagEvent	Magnitud del evento sísmico	2.2
Data	Array de la traza sísmica	[3911, 3882, ...]
publicID	ID publico del evento sísmico generado por el SGC	SGC2018ehtc
phase_code	Fase de la cual se tiene información	P ó S
pick_time_value	Momento de la llegada de la onda P ó S determinada por la columna anterior	2018-03-03 03:44:34
time_value_ms	Valor en milisegundos de la llegada de la onda P ó S	35349.0

Nota. La información es proveniente del Servicio Geológico Colombiano y del Catálogo Sísmico Colombiano.

⁵ Los eventos que suceden dentro de este rango de profundidad son conocidos como eventos locales.

Posterior al proceso de descarga, se construyó un archivo en formato HDF5 (The HDF Group, 2010), y se almacenó en el clúster del grupo de investigación CPS. El archivo contiene 16203 eventos con información relacionada a la llegada de ondas P y ondas S, y 29390 eventos sísmicos sin información sobre la llegada de las ondas P y S. Cada uno de estos eventos se compone de una grabación en sentido Este-Oeste, denominada canal HHE, una grabación en sentido Norte-Sur denominada como canal HHN y por último una grabación en sentido vertical, denominada como canal HHZ y estos se encuentran almacenados respectivamente. De acuerdo con los metadatos (ver Tabla 1) obtenidos por el proceso de descarga, todas las grabaciones de este conjunto de datos generalmente son muestreadas a 100Hz. Para el etiquetado de la onda P, las estampas de tiempo suministradas por el SGC poseen una precisión de 1 segundo, lo cual define una precisión de 100 muestras para el momento exacto de la llegada de la onda P.

Los eventos para los cuales se posee información de la onda P y onda S, fueron etiquetados de la siguiente manera: **publicID_station**, correspondiendo a la unión del ID público generado por el Servicio Geológico Colombiano (SGC) y la estación a la cual pertenece la grabación del evento. Por otro lado, el subconjunto de eventos sin información de las onda P y S se les otorgo la siguiente etiqueta: **eventtime_station**, correspondiente a la unión del metadato *eventtime*, el cual indica la fecha en la cual sucedió el evento sísmico y la estación que genero la grabación de las formas de onda.

Como fue mencionado anteriormente, este conjunto de datos toma 16203 eventos suministrados por el Servicio Geológico Colombiano, los cuales poseen información de llegada de la

onda P. Para este conjunto de eventos realizamos un proceso de caracterización (filtrado) y como resultado se obtiene un subconjunto de 4067 eventos para ser usados en los experimentos que se llevaron a cabo en este proyecto.

La reducción del subconjunto escogido para llevar a cabo los experimentos fue realizada teniendo como criterio la magnitud de los eventos a considerar, por lo tanto, sólo se tuvieron en cuenta eventos con magnitudes mayores a 2.5 en la escala de Richter. Al realizar este filtrado, se obtiene en total 4060 eventos sísmicos. Dentro de este subconjunto se identificaron principalmente 4 tipos de grabaciones: grabación ruidosa, grabación con problema en uno o dos canales, grabaciones no ruidosas, grabaciones con desfase en los canales. La identificación de los tipos de grabaciones se realizó mediante una inspección visual del conjunto de datos, por lo tanto, no se logra enumerar con certeza cuántas grabaciones totales se obtienen de cada tipo.

También se evidencian grabaciones donde se presume que su frecuencia difiere al estándar de grabación de 100Hz utilizado en las estaciones sismológicas. Cabe recordar que cada estación ha sido previamente sintonizada para el entorno donde se encuentra ubicada, por ende, cada grabación posee niveles de offset diferentes, que pueden ser visualizados también en la Figura 4. Para cada uno de los experimentos llevados a cabo en este proyecto, como parte del procesamiento de los datos, se realiza una normalización de estos niveles de offset.

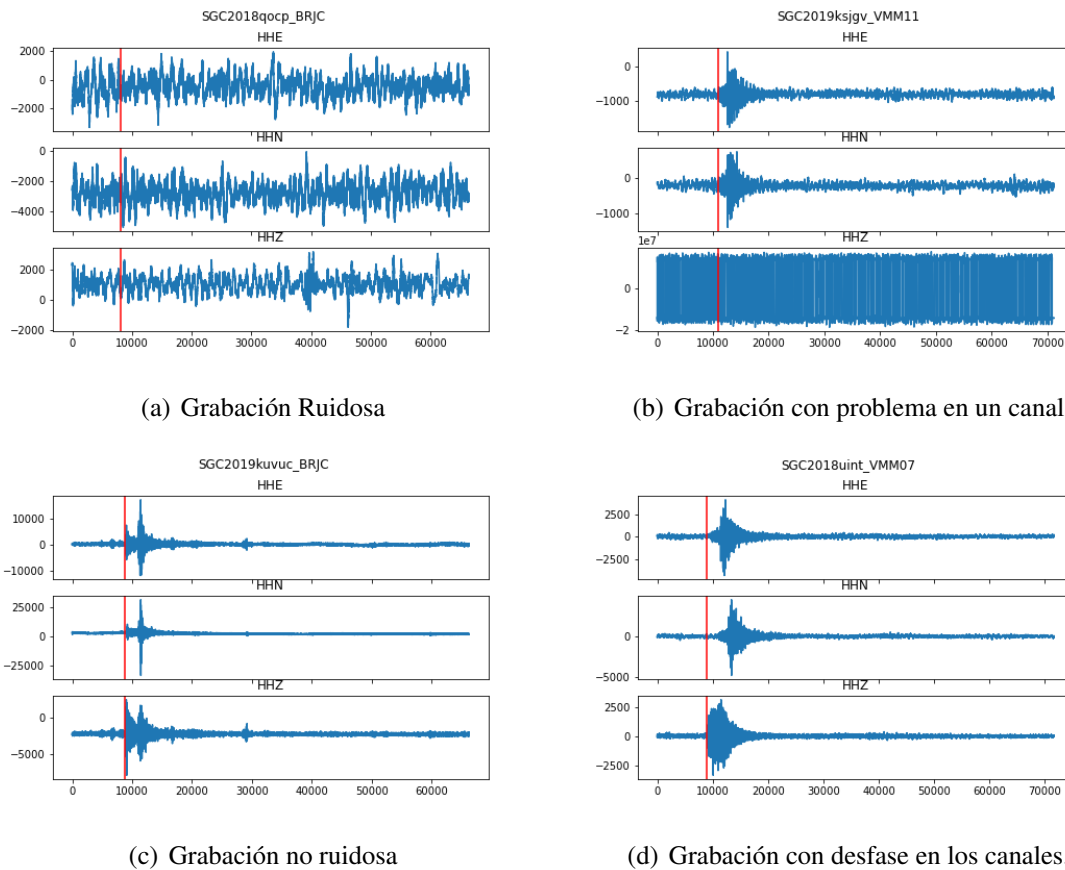


Figura 4. Tipos de grabaciones en el conjunto de datos usados para el entrenamiento.

Nota: La línea vertical roja ilustra la etiqueta del momento de llegada de la onda P.

3.2. STanford Earthquake Dataset (STEAD)

El conjunto de datos de la Universidad de Stanford llamado STEAD (por sus siglas en inglés) (Mousavi *et al.*, 2019), presenta una cantidad útil de eventos sísmológicos para realizar el entrenamiento y validación de una red neuronal. Ésta nos entrega dos archivos: un archivo de tipo HDF5 con las trazas de cada evento y un archivo CSV con diferentes datos adicionales acerca de cada uno de estos. Adicionalmente, todos los eventos cuentan con el mismo formato: trazas

de 6000 muestras a 100 Hz, haciendo que el proceso de tratamiento de datos para ingresarlos a la red sea más sencillo. La frecuencia de muestreo es una de las diferencias de este conjunto de datos con el del SGC, lo que implica un reto en el pre-procesamiento de la señal debido a que es necesario que todas las trazas que se le presentan a la red tengan la misma frecuencia de muestreo, esto también afecta en las predicciones de la misma ya que para el post-procesamiento también se tiene en cuenta que todas las trazas cuenten con una frecuencia de muestreo de 100Hz.

STEAD contiene dos tipos de señales, ambas captadas por equipos sísmicos. Uno de estos tipos son señales de ruido, que son datos reales que no son considerados un evento sísmológico. El segundo tipo de señal, son los eventos sísmológicos clasificados en dos tipos de eventos: locales y globales.

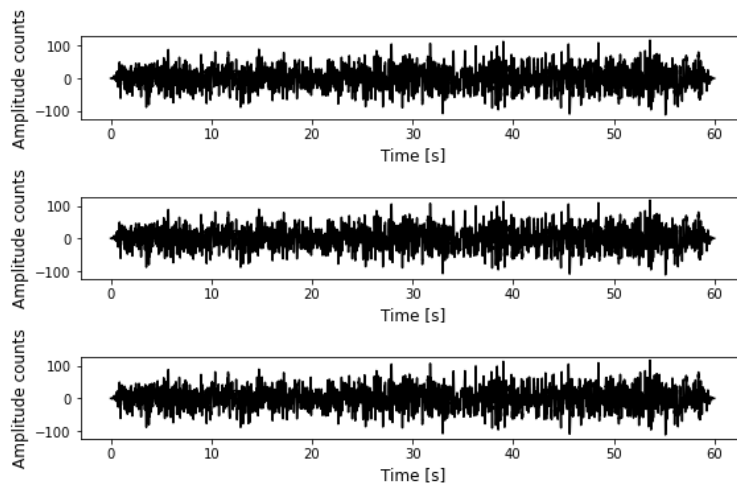


Figura 5. Ejemplo de evento de ruido de la base de datos STEAD.

Este trabajo se enfocó en los eventos clasificados como locales, con una profundidad en el origen menor o igual a 200 km, de magnitud mayor o igual a 2 y con etiquetas de picado manual y

automático, que consisten en eventos picados por la red neuronal de la Universidad de Stanford y por métodos algorítmicos (Mousavi *et al.*, 2019). Esto con el fin de tener un conjunto de datos para entrenar una red neuronal con datos etiquetados de onda P que permita detectar el primer arribo de un evento sísmológico.

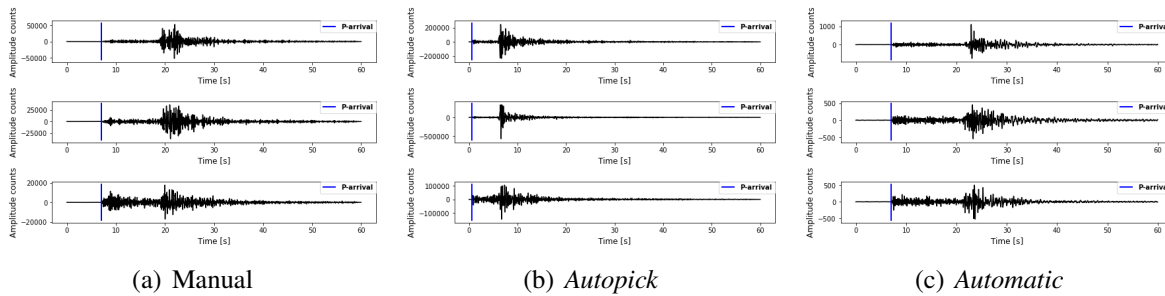


Figura 6. Ejemplo de los tres tipos de etiquetas de la base de datos STEAD.

Para ingresar los datos a red, se hizo un proceso de filtrado en los datos que consiste en extraer la información del archivo CSV y extraer sólo los eventos de interés. Luego de esto, se almacena en un archivo de datos en Formato de Datos Jerárquicos (H5) los diferentes eventos y las etiquetas de los picados P.

Tabla 2

Cantidad de datos de los diferentes grupos de datos después de aplicar los filtros

Tipo de evento	Cantidad de datos
Picado P Manual	195698
Picado P <i>Autopick</i>	34696
Picado P <i>Automatic</i>	17434
Ruido	235426

Finalmente, después de los filtros aplicados, se tienen dos grupos de datos finales: el conjunto de ruido con 235426 eventos y el conjunto de interés para este trabajo son los datos sísmo-

lógicos con información acerca de 247828 eventos. De esta cantidad de eventos sísmológicos, se separan para entrenamiento, validación y evaluación de la red neuronal junto con los datos del SGC.

4. Red convolucional para la detección de primeros arribos

La tarea de detectar el momento de llegada de la onda P suele ser realizada por un experto, el cual analiza la grabación del evento sísmico y se encarga de efectuar el proceso de etiquetado de forma manual. Este proceso toma entre el 20% y el 50% del tiempo empleado por el analista al observar una traza sismológica (Tsai *et al.* (2020) , Al-Mashhor *et al.* (2019)). Esta detección manual recae netamente en la perspectiva de la persona encargada de realizar el etiquetado, generando inconsistencias al momento de determinar el tiempo de ruptura de la onda P.

Para optimizar el proceso de detección se han desarrollado diferentes técnicas para realizar el etiquetado de forma automática o semiautomática, como lo es el algoritmo STA/LTA⁶. Para usar este algoritmo es necesario hacer ajustes en sus parámetros de funcionamiento, dependiendo de la zona donde el instrumento de grabación sea ubicado. Se requiere el ajuste de dos ventanas de tiempo, una ventana de tiempo de corta duración que aporta la información sobre el suceso de un evento sísmico y otra ventana de tiempo de larga duración encargada de aportar información sobre el ruido captado en la grabación. A partir de la relación de estas dos ventanas de tiempo se fija un umbral para el cual se va a considerar la llegada de la onda P.

⁶ Para más detalles sobre este algoritmo dirigirse a la sección 2.2.1

Este proyecto abordó la detección del primer arribo aprovechando la capacidad de extracción de características que pueden tener las redes neuronales convolucionales (CNN), en contraste con el algoritmo STA/LTA, los parámetros de una red CNN son ajustados durante las etapas de entrenamiento y no dependen de la zona de ubicación de la estación sísmológica.

Se usó una arquitectura de red neuronal convolucional CNN previamente utilizada para la detección de primeros arribos en trazas sísmicas sintéticas del valle medio del Magdalena (Zambrano y Cachopo, 2022), esta red fue entrenada con grabaciones sísmicas de 1251 muestras. Previo al entrenamiento, cada grabación fue sometida a un filtro pasa bajas con el objetivo de eliminar componentes de alta frecuencia, para posteriormente aplicar 31 transformadas Wavelet. La arquitectura propuesta en el documento *Detección automática de primeros arribos en trazas sintéticas terrestres del valle medio del magdalena usando redes neuronales artificiales* (Zambrano y Cachopo, 2022) consiste en una capa convolucional con 10 filtros (Kernels) y un tamaño de 20 muestras, cumpliendo como función principal la búsqueda de patrones temporales en las grabaciones que se le presentan. Seguido de esta capa, se presenta una capa de Maxpooling, la cual se encarga de tomar únicamente los máximos valores obtenidos luego de la aplicación del kernel sobre la entrada.

La CNN propuesta (Zambrano y Cachopo, 2022) continúa el procesamiento con una capa densamente conectada, siguiendo con un aplanamiento en una dimensión, para luego continuar con un par de capas densamente conectadas encargadas de la clasificación de las muestras. Este proyecto modificó la arquitectura de referencia (Zambrano y Cachopo, 2022) agregando dos capas de Dropout previo a la entrada de las dos últimas capas densamente conectadas. La función de las

capas Dropout ignoran aleatoriamente un porcentaje de neuronas en las capas densamente conectadas durante el proceso de entrenamiento con el objetivo de ayudar a mejorar la generalización sobre los datos de entrenamiento. La Figura 7 presenta la arquitectura implementada y previamente descrita.

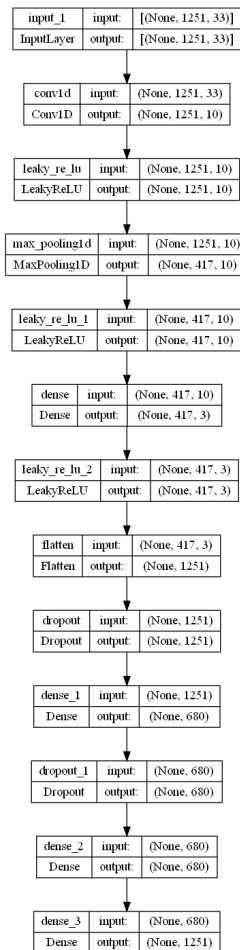


Figura 7. Arquitectura CNN para la detección de primeros arribos.

Nota: La red cuenta con una capa convolucional 1D, con función de activación Leaky ReLu; seguida de una capa Maxpooling 1D, nuevamente con función de activación Leaky ReLu; seguida de una capa densamente conectada y un aplanamiento en una dimensión para ingresar a un par de capas densamente conectadas con capas Dropout.

A continuación, esta sección describe detalladamente los procesos de pre-procesamiento y post-procesamiento de la red neuronal modificada por los autores de este proyecto.

4.1. Pre-procesamiento de las grabaciones sismológicas

El pre-procesamiento de las grabaciones sismológicas inicia al generar recortes sobre la longitud total en muestras de los eventos. En el caso del conjunto de datos proveniente del Servicio Geológico Colombiano, la longitud es variable y en todas las grabaciones es mayor a 50000 muestras. Se realizaron 2 grupos de recortes para cumplir con el tamaño requerido por la entrada de la red que se utilizó: un primer grupo con recorte total de 1251 muestras, ubicando aleatoriamente la llegada de la onda P entre la muestra 200 y 1100, y el segundo grupo con un total de 6000 muestras, nuevamente ubicando el momento de ruptura de la onda P entre la muestra 200 y 2000.

El proceso continuó con la aplicación de un filtro pasa bajas para la eliminación de componentes de alta frecuencia no deseados en las grabaciones, éste filtro se aplica sobre los 3 canales que componen las grabaciones. Posterior a este filtrado, se generaron una serie de transformadas Wavelet, donde el número de estas depende del experimento propuesto para determinar el número adecuado de transformadas a aplicar. Esta transformada permite seccionar las grabaciones en componentes de escala frecuencia-tiempo, permitiendo representar la mayoría de la energía de una señal en coeficientes de expansión y ayudando así a la eliminación del ruido en las señales (Burrus *et al.*, 1998). En este caso se decidió aplicar la transformada Wavelet continua (CWT), la cual se define como:

$$F(a,b) = \int f(t)w\left(\frac{t-a}{b}\right) dt \quad (1)$$

Donde $w(t)$ representa la ondícula madre que para este caso es la ondícula Morlet, siendo a y b variables de desplazamiento y escalamiento en tiempo para dicha ondícula. Estos escalamientos permiten revelar información general de la señal, así como también, permite observar detalles más finos de la misma. La implementación de esta transformada se realizó con la librería PyWavelets (Lee *et al.*, 2019). Como último paso, se elaboró una etiqueta en forma de escalón para el proceso de entrenamiento, en la cual se ha asignado un valor de 0 para las muestras previas al momento de ruptura de la onda P y un valor de 1 para todas las muestras posteriores a la ruptura.

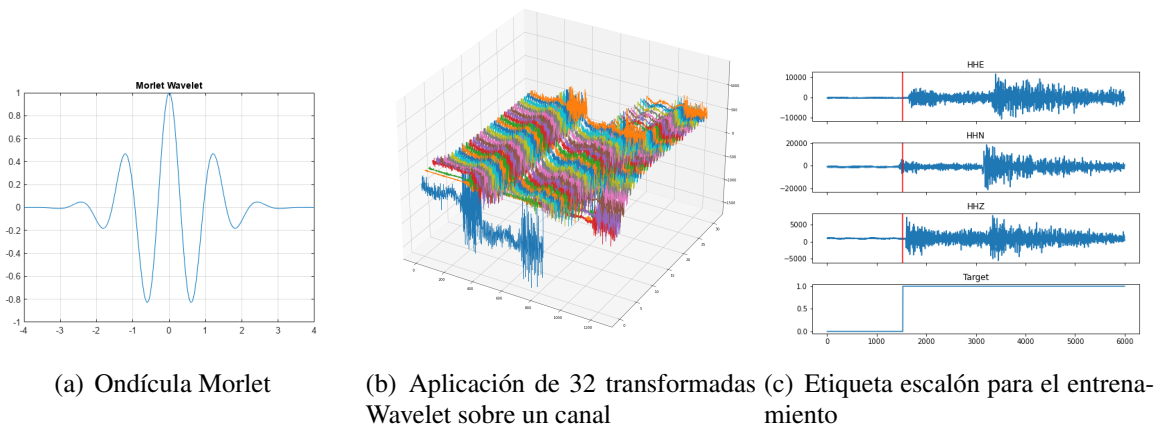
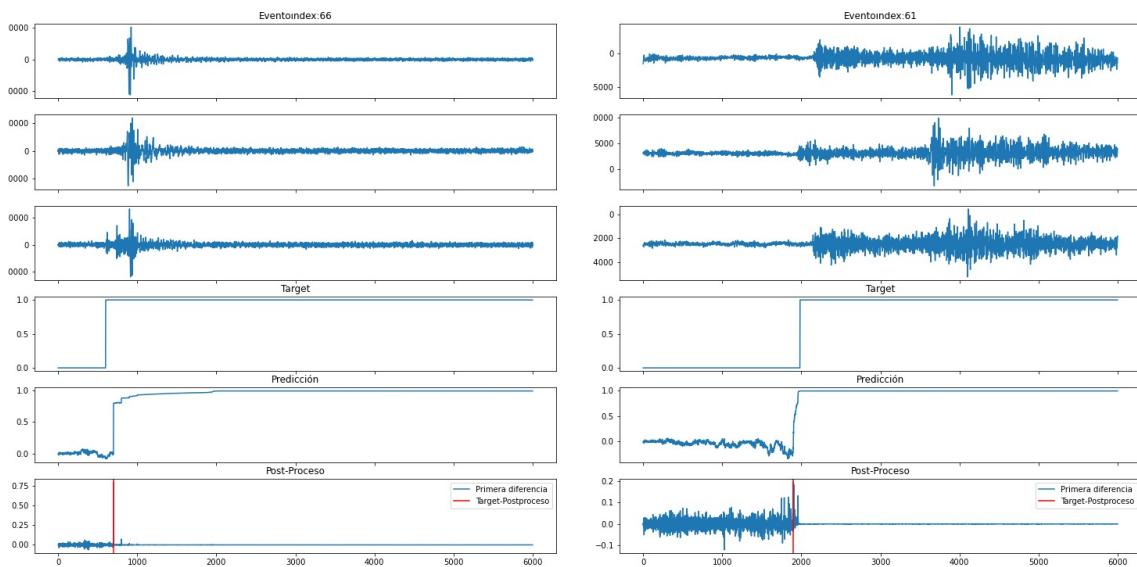


Figura 8. Etapa de pre-procesamiento

Nota: 8(a) Corresponde a la ondícula madre Morlet, 8(b) Es el resultado de aplicar 32 transformadas Wavelet sobre un canal con la ondícula Morlet, 8(c) Ejemplo de la etiqueta escalón implementada para el entrenamiento

4.2. Post-procesamiento de la respuesta de la red entrenada

La selección del primer arribo se realizó tomando el vector de respuesta generado por la red propuesta para este trabajo, el cual arroja un vector de 1251 muestras o 6000 muestras, dependiendo del experimento en cuestión. El vector busca tener la forma de la etiqueta escalón propuesta para el entrenamiento y sobre este se aplicó un proceso de primera diferencia con el fin de obtener la máxima diferencia que exista en la salida de la predicción. En la figura 9 se ilustra el proceso de post-procesamiento sobre una predicción realizada en una grabación, recopilada por la Universidad de Stanford (figura 9(a)), así mismo, se ilustra el proceso sobre una grabación del conjunto de datos proveniente del Servicio Geológico Colombiano (figura 9(b)). Por último, el cálculo del error absoluto y la desviación estándar sobre las predicciones se realizó tomando el índice con el que se obtuvo la mayor diferencia y fue restado con la posición en la cual se encuentra la etiqueta original de la onda P, donde es importante tener en cuenta la precisión de las etiquetas originales de las estampas de tiempo otorgadas por el Servicio Geológico Colombiano que adicionan aproximadamente más o menos 100 muestras de error.



(a) Post-procesamiento grabación STEAD

(b) Post-procesamiento grabación proveniente del Servicio Geológico Colombiano

Figura 9. Etapa de post-procesamiento de la respuesta de la red entrenada.

Nota: La Figura 9(a) muestra el proceso de primera diferencia sobre una predicción en el conjunto de datos de la universidad de Stanford; y la Figura 9(b) muestra el proceso de post-procesamiento sobre una predicción en una grabación del Servicio Geológico Colombiano.

5. Experimentación

5.1. Caracterización de la red

Debido a que este proyecto parte del uso de una red neuronal diseñada para un conjunto de datos diferente al de interés, para los datos del Servicio Geológico Colombiano la primera etapa realizada es de caracterización. Esta etapa permitió tener un punto de partida en términos de comportamiento de la red al procesar los datos del Servicio Geológico Colombiano. En las tablas 3 y 4 se encuentran respectivamente las características de los conjuntos de datos y la información de los experimentos realizados en esta fase.

Esta etapa contó con tres diferentes acercamientos: primero, una predicción utilizando datos del SGC y STEAD para poder obtener métricas del modelo entrenado previamente en Zambrano y Cachopo (2022); segundo, un entrenamiento de la misma arquitectura con datos del SGC y una predicción del modelo entrenado de la misma forma que el primero; tercero, un entrenamiento

Tabla 3
Conjuntos de datos utilizados en esta fase de caracterización

	Características	Número de eventos(entrenamiento, validación y evaluación)
SGC	Datos reales crudos.	12963, 1620, 1620
STEAD	Eventos locales con magnitud mayor o igual a 2, con picado manual, autopick y automatic.	12963, 1620, 1620

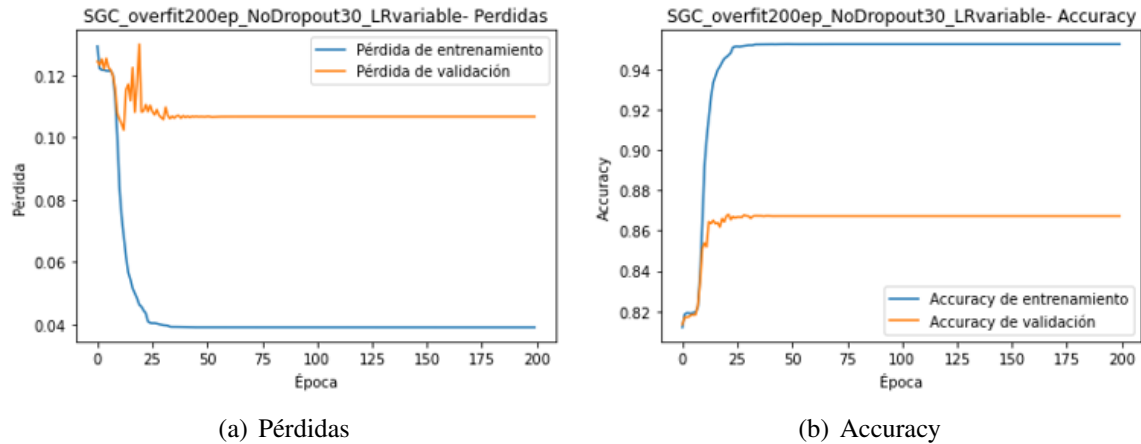


Figura 10. Gráficas del comportamiento del experimento: Entrenamiento SGC

Nota: Estos diagramas representan los valores de pérdida y accuracy en el entrenamiento y validación.

Tabla 4

Experimentos realizados en la fase de caracterización

Nombre del experimento	Set de datos utilizado
Modelo preentrenado	Modelo final de Zambrano y Cachopo (2022)
Entrenamiento SGC	SGC
Entrenamiento SGC+STEAD	SGC y STEAD

de la red neuronal utilizando un conjunto de datos conformado por SGC y STEAD que entrega un nuevo modelo que es evaluado de la misma forma que los dos anteriores. Finalmente, es importante resaltar que, en esta etapa, como en la arquitectura de referencia solo se tenían eventos con un solo canal y los datos con los que se trabajó contienen tres señales por evento. Se realizó un proceso de selección aleatorio entre los tres canales de información de cada evento.

Los dos entrenamientos realizados en esta fase, mencionados anteriormente, contaron con parámetros de entrenamiento similares, como lo son: *Learning Rate* variable, sin *dropout* y 200 épocas. En esta etapa también se buscó analizar el comportamiento del entrenamiento, dónde se

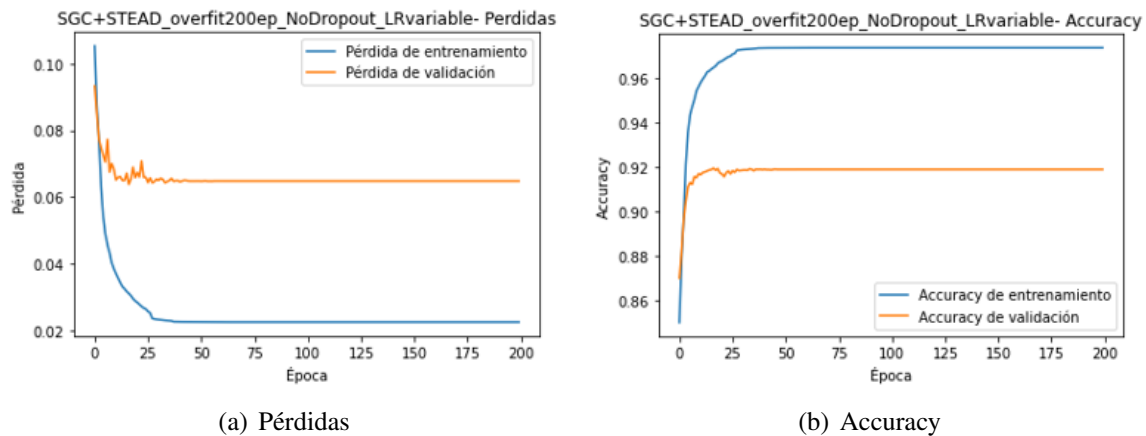


Figura 11. Gráficas del comportamiento del experimento: Entrenamiento SGC + STEAD

logra ver un sobre entrenamiento y principalmente verificar si la arquitectura funciona para encontrar los primeros arribos en los eventos de interés. En las figuras 10 y 11, se ven las gráficas de entrenamiento de los dos acercamientos mencionados anteriormente, donde se resalta un mejor resultado en los valores de pérdida y *accuracy* del entrenamiento donde se utilizan los dos conjuntos de datos SGC y STEAD.

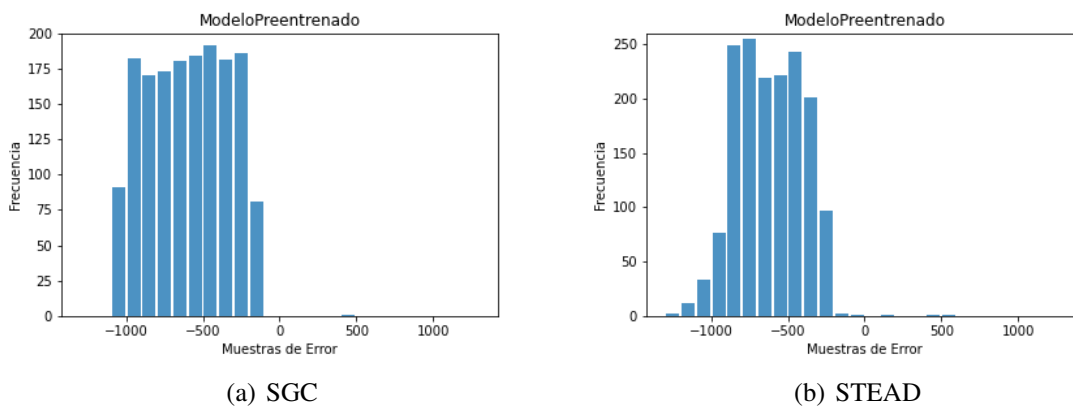


Figura 12. Diagramas de barras de la predicción para el modelo preentrenado.

Nota: Estos diagramas representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Adicionalmente, con la predicción realizada, se calculan unas muestras de error en los valores del primer arribo que consisten en la resta entre el valor original de la etiqueta y el valor calculado con la predicción del modelo. En las figuras 12,13 y 14, se muestra el comportamiento de las predicciones de los tres modelos que se evalúan en esta etapa, en donde se busca que tengan un aspecto similar a una distribución normal con media en cero, ya que esta refleja la mayor cantidad de frecuencia en la media lo que en este caso significa error de 0 muestras.

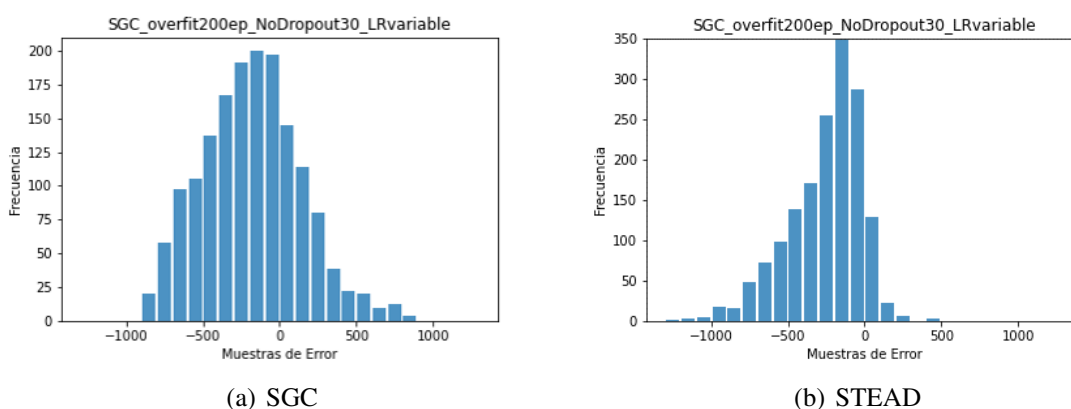


Figura 13. Diagramas de barras de la predicción del experimento: Entrenamiento SGC.

Nota: Estos diagramas representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Revisando los resultados de error promedio absoluto y desviación estándar en muestras expuestos en la tabla 5 y los diagramas de barras mencionados anteriormente, se evidencia: primero, que los datos con los que se entrenó la red neuronal anteriormente, no comparten muchas características con los del SGC y STEAD como lo indica el error calculado; segundo, que el entrenamiento con los datos sí presenta una mejora en las métricas; y tercero, que el uso de los dos conjuntos de datos para el entrenamiento afectan el error absoluto promedio para el SGC pero

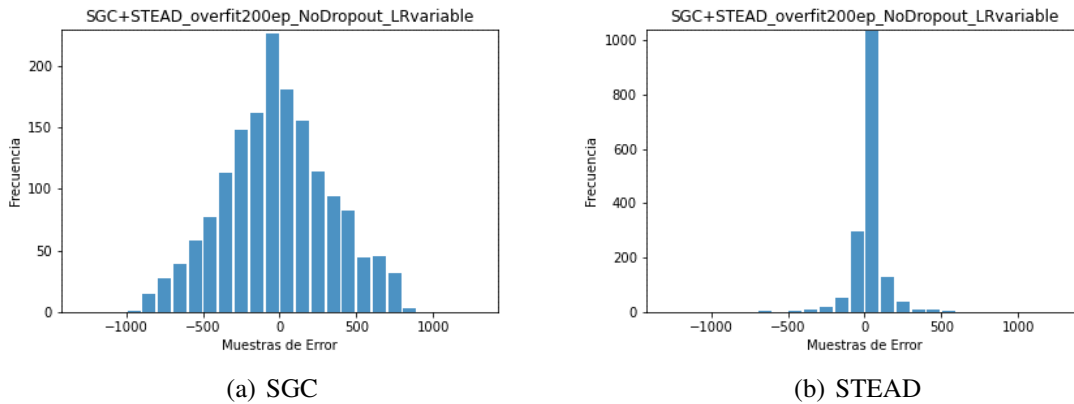


Figura 14. Diagramas de barras de la predicción del experimento: Entrenamiento SGC + STEAD.

Nota: Estos diagramas representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

disminuye la desviación estándar, esto también se ve en el diagrama 14(a).

Tabla 5

Caracterización de la red

	SGC Error absoluto promedio	SGC Desviación estándar	STEAD Error absoluto promedio	STEAD Desviación estándar
Modelo Preentrenado	602	261	628	215
Entrenamiento SGC	305	250	347	222
Entrenamiento SGC+STEAD	322	247	43	92

Nota. Todos los valores numéricos son en número de muestras.

5.2. Diseño de experimentos

Para esta fase de experimentación, se revisó el comportamiento de la predicción de la red para el conjunto de datos del SGC realizando variaciones a la red, a la entrada de la red, al proceso de entrenamiento y finalmente a la cantidad de datos presentados a la red. Para este proceso se trabajó con las siguientes hipótesis:

- I. El uso de capas de *dropout* con porcentaje de 10% y de 30%, permite que el modelo no

- alcance el sobre entrenamiento lo que produce que el entrenamiento presente una mejoría en las métricas de la predicción.
- II. La utilización de la información completa en tres dimensiones de cada evento en los procesos de entrenamiento, validación y evaluación genera mejoría en los resultados de la predicción.
 - III. El uso de una mayor cantidad de transformadas y el orden en el que se le presentan a la red neuronal en los procesos de entrenamiento, validación y evaluación afectan en los resultados de la predicción.
 - IV. El uso de más muestras por canal en los procesos de entrenamiento, validación y evaluación genera mejoría en los resultados de la predicción.
 - V. Tener un proceso de entrenamiento con el conjunto de datos STEAD sumado con un reentrenamiento utilizando los datos del SGC mejora los resultados de predicción de la red.
 - VI. Realizar el proceso de entrenamiento y validación utilizando un grupo con el mismo número de datos para cada uno de los conjuntos de datos al mismo tiempo implica una mejoría en los resultados de la predicción de la red.
 - VII. Realizar los procesos de entrenamiento, validación y evaluación utilizando un grupo preseleccionado de 4060 eventos para cada uno de los conjuntos de datos al mismo tiempo implica una mejoría en los resultados de la predicción de la red.

Tabla 6
Conjuntos de datos utilizados en esta fase experimental 1

	Características	Número de eventos(entrenamiento, validación y evaluación)
SGC	Datos reales crudos.	12963, 1620, 1620
STEAD	Eventos locales con magnitud mayor o igual a 2, con picado manual, autotpick y automatic.	12963, 1620, 1620

5.2.1. Comparación de resultados de experimentos.

A continuación, se presentan las 4 fases de experimentación que se realizaron y como estas ayudaron a comprobar las hipótesis mencionadas anteriormente.

5.2.1.1. Fase experimental 1. En esta fase se realizaron los experimentos para comprobar como responde la red neuronal cuando se le agregan capas de *dropout*, se utilizan 12963 eventos del SGC y la misma cantidad de STEAD, se utilizó una entrada con un solo canal de información seleccionado al azar entre los tres disponibles, un filtro pasa bajas y 31 transformadas wavelet de esta canal previamente seleccionado, dando como resultado una dimensión de entrada de la forma: (cantidad de eventos, 1251, 33). La arquitectura de la red utilizada en esta fase se encuentra en la figura 7. En las tablas 6 y 7 se encuentran respectivamente las características de los conjuntos de datos y la información de los experimentos realizados en esta fase.

En las figuras 15 y 16 se observa que los entrenamientos que se realizaron con un *dropout* de 30%, alcanzaron un valor más bajo de pérdidas y un valor más alto en accuracy en comparación con los resultados del entrenamiento y validación en los que se usó un 10% de *dropout*.

Tabla 7

Experimentos realizados en la fase experimental 1

Nombre del experimento	Set de datos utilizado
Entrenamiento SGC Dropout 10%	SGC
Entrenamiento STEAD Dropout 10%	STEAD
Entrenamiento SGC+STEAD Dropout 10%	SGC y STEAD
Entrenamiento SGC Dropout 30% SGC	SGC
Entrenamiento STEAD Dropout 30% STEAD	STEAD
Entrenamiento SGC+STEAD Dropout 30% SGC+STEAD	SGC y STEAD

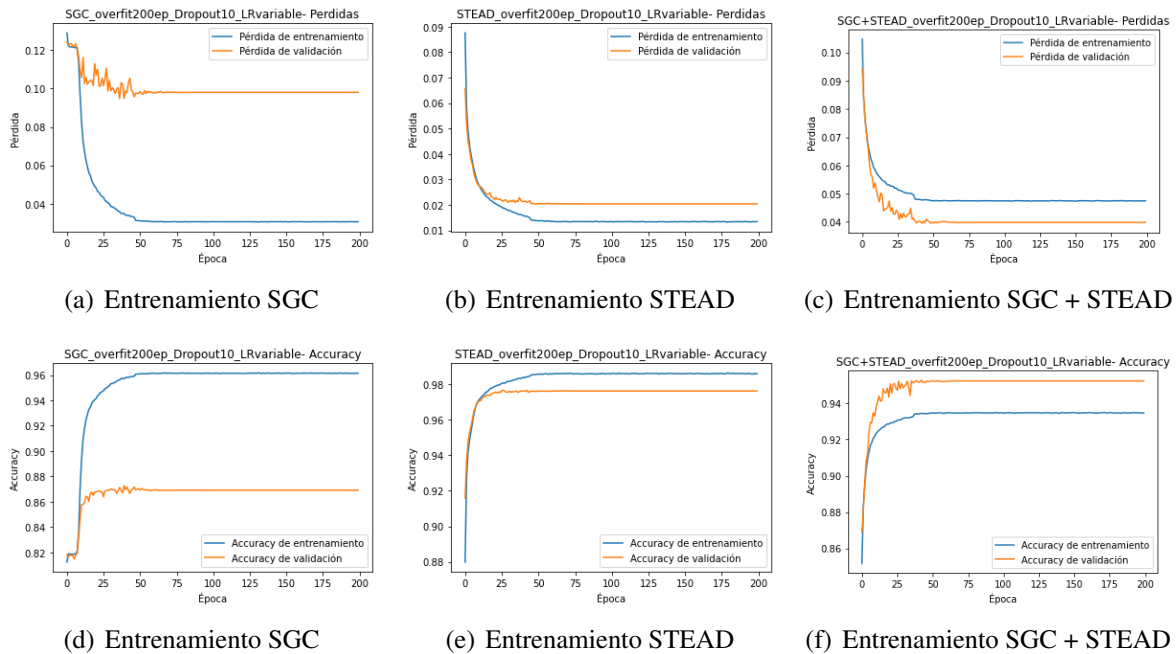


Figura 15. Fase Experimental 1: Entrenamientos con Dropout 10%

Analizando la tabla 8, se observa que el resultado en el error absoluto promedio del entrenamiento con *dropout* de 10% y solamente utilizando el conjunto de datos STEAD es el de menor valor para esta fase, asimismo se evidencia que el entrenamiento con *dropout* de 30% y solamente utilizando el conjunto de datos SGC y STEAD tiene mejores resultados en la desviación estándar. Para definir cuál es el mejor proceso de entrenamiento en esta fase se revisó la forma del diagrama

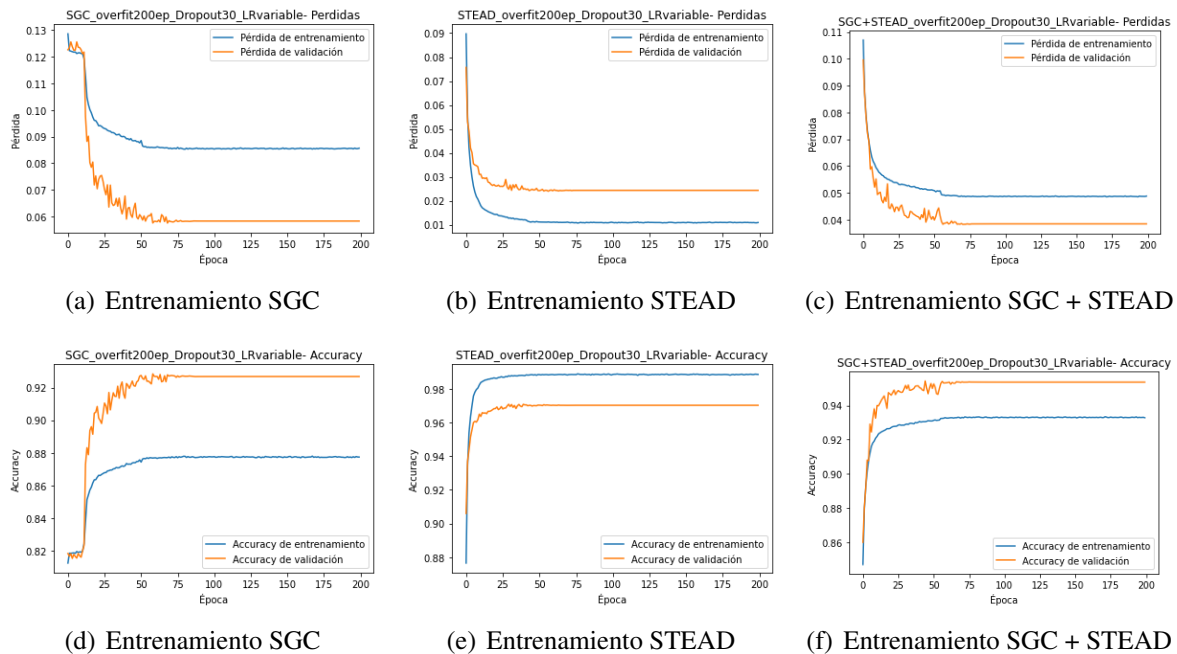
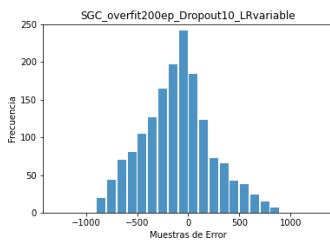


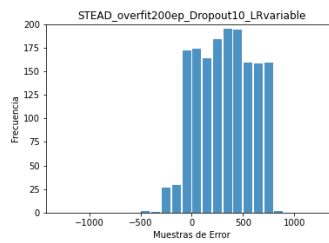
Figura 16. Fase Experimental 1: Entrenamientos con Dropout 30 %

de barras en las figuras 17(b) y 18(c), encontrando que en la segunda se describe una distribución normal con media en cero que nos representa un mejor comportamiento del primer arribo predicho.

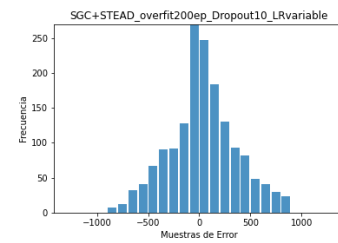
En conclusión, para esta fase, se comprobó la hipótesis I donde se expone que agregar capas de *dropout* mejora los resultados en el modelo y se encuentra que el uso de 30 % de este resulta en un mejor rendimiento de la red. Adicionalmente, se comprueba revisando la forma de las figuras 17(c) y 18(c) que el uso de los dos conjuntos de datos en los procesos de entrenamiento, validación y evaluación implica un mejor comportamiento en la predicción de la red. Teniendo en cuenta lo mencionado anteriormente, se resalta en verde en la tabla 8 el mejor experimento de esta fase el cual determina un punto de partida para la siguiente fase.



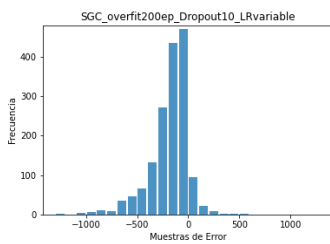
(a) Métricas SGC
Entrenamiento SGC



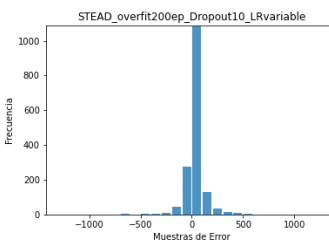
(b) Métricas SGC
Entrenamiento STEAD



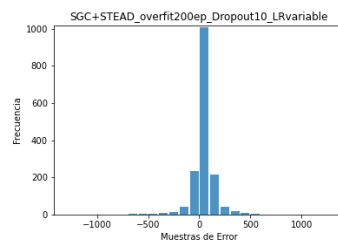
(c) Métricas SGC
Entrenamiento SGC + STEAD



(d) Métricas STEAD
Entrenamiento SGC



(e) Métricas STEAD
Entrenamiento STEAD



(f) Métricas STEAD
Entrenamiento SGC + STEAD

Figura 17. Diagramas de barras Fase Experimental 1: Entrenamientos con Dropout 10%.

Nota: Estos diagramas representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Tabla 8

Resultados Fase 1

	SGC Error absoluto promedio	SGC Desviación estándar	STEAD Error absoluto promedio	STEAD Desviación estándar
Entrenamiento SGC Dropout 10%	248	212	30	88
Entrenamiento STEAD Dropout 10%	248	213	39	96
Entrenamiento SGC+STEAD Dropout 10%	253	215	50	98
Entrenamiento SGC Dropout 30%	364	243	48	94
Entrenamiento STEAD Dropout 30%	365	243	49	90
Entrenamiento SGC+STEAD Dropout 30%	250	205	28	79

Nota. Todos los valores numéricos son en número de muestras.

5.2.1.2. Fase experimental 2. En las tablas 9 y 10 se evidencian respectivamente las características de los conjuntos de datos y la información de los experimentos realizados en esta fase. Esta fase experimental parte con el uso de capas de *dropout* de 30% y el uso de ambos conjun-

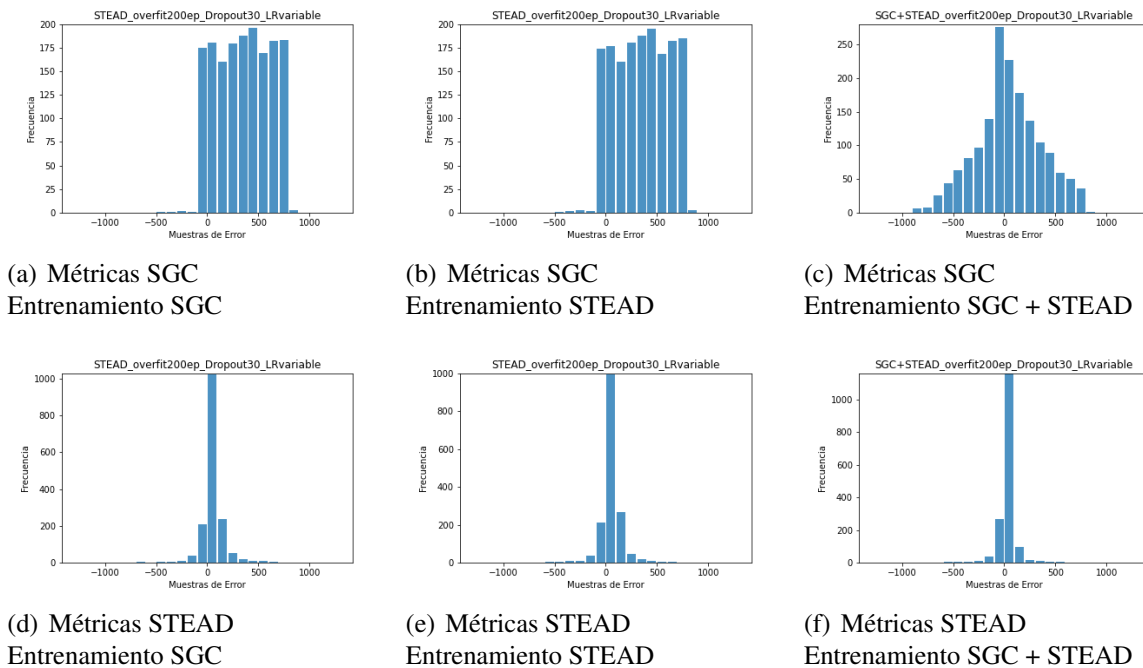


Figura 18. Fase Experimental 1: Entrenamientos con Dropout 30%.

Nota: Estos diagramas representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

tos de datos con una cantidad de 16203 cada uno en los procesos de entrenamiento, validación y evaluación; y busca comprobar cómo el orden en el que se le presentan los canales y las transformadas a la red sumado con la variación en la cantidad de transformadas afectan los resultados en la predicción de esta. Para esta fase se determinan 7 entradas diferentes todas con 1251 muestras, las cuales se definen de la siguiente forma:

1. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 31 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 31 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 31 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 1251, 99)

2. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 9 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 9 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 9 Transformadas del Canal HHZ].
Dimensión total de: (Cantidad de eventos, 1251, 33)
3. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 49 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 49 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 49 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 1251, 153)
4. Entrada de la red organizada de la siguiente forma: [Canal HHE, 1 filtro pasa bajas en el Canal HHE, 31 Transformadas del Canal HHE , Canal HHN, 1 filtro pasa bajas en el Canal HHN, 31 Transformadas del Canal HHN, Canal HHZ, 1 filtro pasa bajas en el Canal HHZ, 31 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 1251, 99)
5. Entrada de la red organizada de la siguiente forma: [Canal HHE, 1 filtro pasa bajas en el Canal HHE, 9 Transformadas del Canal HHE, Canal HHN, 1 filtro pasa bajas en el Canal HHN, 9 Transformadas del Canal HHN, Canal HHZ, 1 filtro pasa bajas en el Canal HHZ, 9 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 1251, 33)
6. Entrada de la red organizada de la siguiente forma: [Canal HHE, 1 filtro pasa bajas en el Canal HHE, 49 Transformadas del Canal HHE, Canal HHN, 1 filtro pasa bajas en el Canal HHN, 49 Transformadas del Canal HHN, Canal HHZ, 1 filtro pasa bajas en el Canal HHZ,

Tabla 9

Conjuntos de datos utilizados en esta fase experimental 2

	Características	Número de eventos(entrenamiento, validación y evaluación)
SGC	Datos reales crudos.	12963, 1620, 1620
STEAD	Eventos locales con magnitud mayor o igual a 2, con picado manual, autpick y automatic.	12963, 1620, 1620

Tabla 10

Experimentos realizados en la fase experimental 2

Nombre del experimento	Set de datos utilizado
Entrenamiento Entrada 1	SGC y STEAD
Entrenamiento Entrada 2	SGC y STEAD
Entrenamiento Entrada 3	SGC y STEAD
Entrenamiento Entrada 4	SGC y STEAD
Entrenamiento Entrada 5	SGC y STEAD
Entrenamiento Entrada 6	SGC y STEAD
Entrenamiento Entrada 7	SGC y STEAD

49 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 1251, 153)

7. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 49 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 49 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 49 Transformadas de Canal HHZ]. Adicionalmente, se le realiza la transpuesta, lo que resulta en una dimensión total en la entrada de la forma: (Cantidad de eventos, 153, 1251)

Con la excepción de las dimensiones en la entrada y la salida, la arquitectura utilizada para esta fase es la misma de la figura 7. Se utilizaron las mismas condiciones: entrenamiento por 100 épocas, *Learning Rate* variable y se utilizaron los mismos datos en los procesos de entrenamiento,

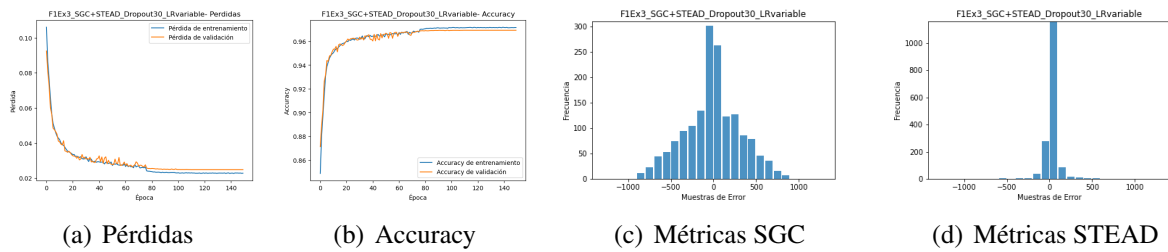


Figura 19. Fase Experimental 2: Entrenamiento Entrada 1.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

validación y evaluación.

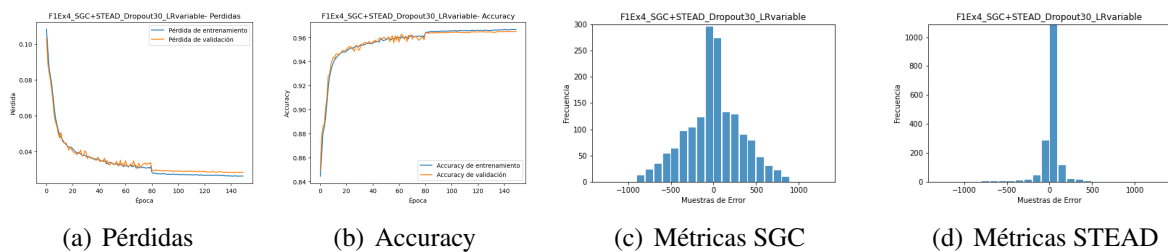


Figura 20. Fase Experimental 2: Entrenamiento Entrada 2.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

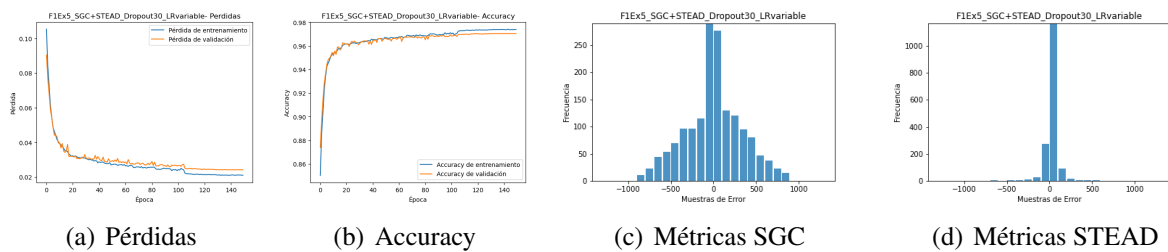


Figura 21. Fase Experimental 2: Entrenamiento Entrada 3.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

En las figuras 19 - 25, se observa en las graficas del entrenamiento y validación, que en la

mayoría de los casos el comportamiento de estas dos etapas fue similar con la excepción de las figuras 25(a) y 25(b) en las que se evidencia que el proceso de validación no llega a los niveles de los otros seis casos.

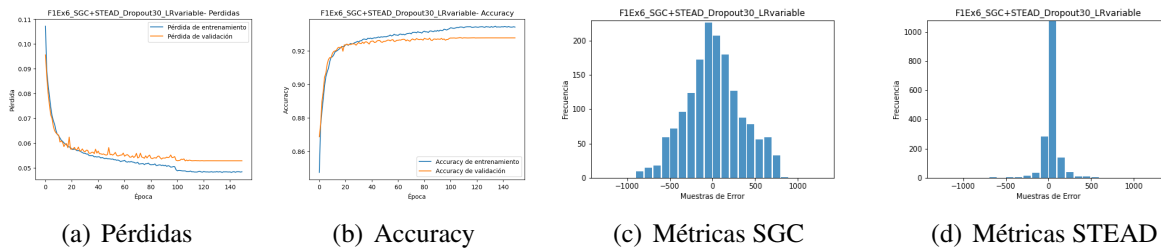


Figura 22. Fase Experimental 2: Entrenamiento Entrada 4.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

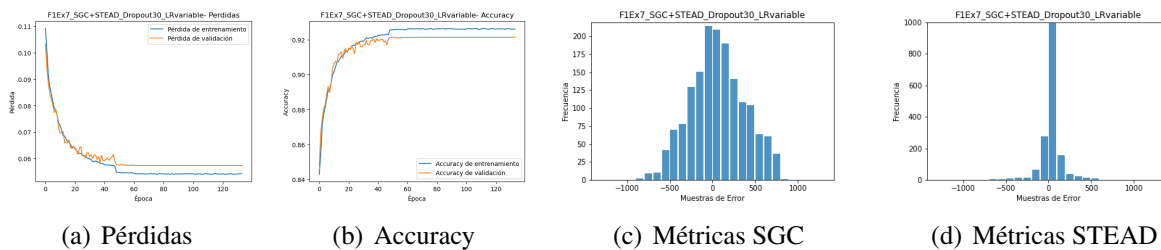


Figura 23. Fase Experimental 2: Entrenamiento Entrada 5.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Analizando los resultados numéricos presentados en la tabla 11, se evidencia que el uso de una mayor cantidad de transformadas no beneficia el comportamiento de la red y sí implica un mayor uso de memoria. Adicionalmente, se observa que presentarle las tres dimensiones del evento al inicio implica una mejora con respecto a la presentación intercalada de canales con sus respectivos filtros y transformadas. También es importante resaltar el bajo resultado del experimento con la entrada 7 que indica que la dimensión de entrada de la red sí importa y que la forma en la que se le

están presentando los datos a la red es la óptima en este caso. Finalmente, se puede evidenciar que los experimentos que usan la entrada 1 y 2, resaltados en color verde, cuentan con unos valores similares.

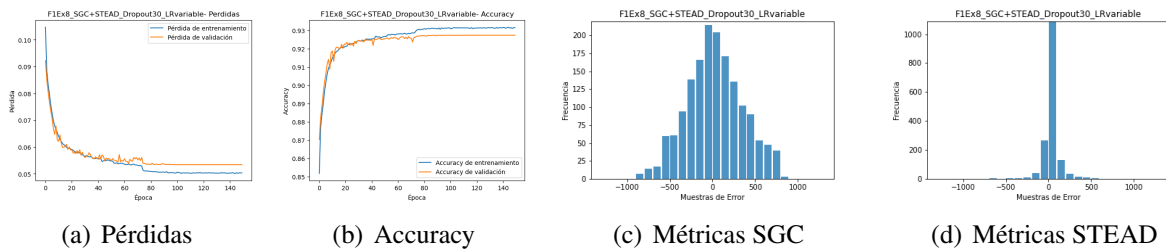


Figura 24. Fase Experimental 2: Entrenamiento Entrada 6.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

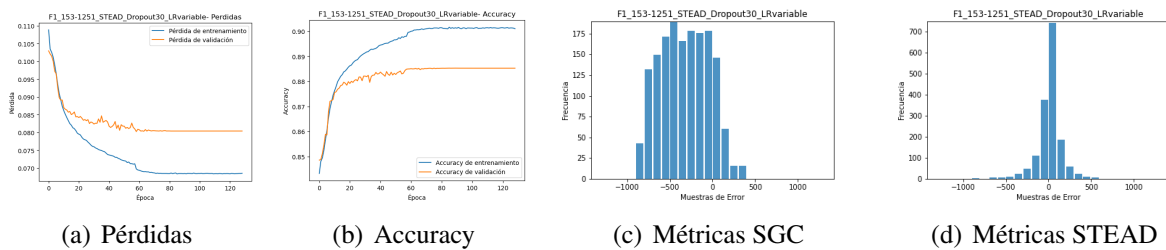


Figura 25. Fase Experimental 2: Entrenamiento Entrada 7.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

En conclusión, observando la tabla 11 y las figuras 19 y 20, se evidencia que el comportamiento es similar en estos dos experimentos, donde se resalta el resultado de desviación estándar de la entrada 1. Teniendo en cuenta lo mencionado anteriormente, se decidió continuar el proceso de experimentación en las siguientes fases realizando pruebas con la entrada 1 y la 2.

Tabla 11
Resultados Fase 2

	SGC Error absoluto promedio	SGC Desviación estándar	STEAD Error absoluto promedio	STEAD Desviación estándar
Entrada 1	248	212	30	88
Entrada 2	248	213	39	96
Entrada 3	255	218	27	79
Entrada 4	262	205	38	86
Entrada 5	260	198	55	112
Entrada 6	264	202	38	86
Entrada 7	354	243	84	122

Nota. Todos los valores numéricos son en número de muestras.

Tabla 12
Conjuntos de datos utilizados en esta fase experimental 3

	Características	Número de eventos(entrenamiento, validación y evaluación)
SGC	Datos reales crudos con magnitud mayor o igual a 2.5.	2436, 812, 812
STEAD	Eventos locales con magnitud mayor o igual a 2, con picado manual, autopick y automatic.	12963, 1620, 1620

5.2.1.3. Fase experimental 3. Para esta fase, se utilizan las mismas condiciones de arquitectura y parámetros de entrenamiento y se parte de los resultados de las fases anteriores. Con el fin de mejorar los resultados, en esta parte de la experimentación se propuso: primero, reducir el conjunto de datos del SGC utilizado anteriormente a 4060 eventos que tienen magnitud igual o mayor a 2.5 con el fin de reducir el efecto que tiene el ruido en este set de datos. Segundo, utilizar una cantidad de muestras de 6000 en lugar de las 1251. Por último, realizar un proceso de entrenamiento con datos del STEAD y con el modelo obtenido realizar un reentrenamiento con el conjunto de datos de SGC. En las tablas 12 y 13 se encuentran respectivamente las características de los conjuntos de datos y de los experimentos realizados en esta fase experimental.

Tabla 13
Experimentos realizados en la fase experimental 3

Nombre del experimento	Set de datos utilizado
Entrenamiento STEAD Entrada 0	STEAD
Reentrenamiento Entrada 0	SGC
Entrenamiento STEAD Entrada 1	STEAD
Reentrenamiento Entrada 1	SGC
Entrenamiento STEAD Entrada 2	STEAD
Reentrenamiento Entrada 2	SGC

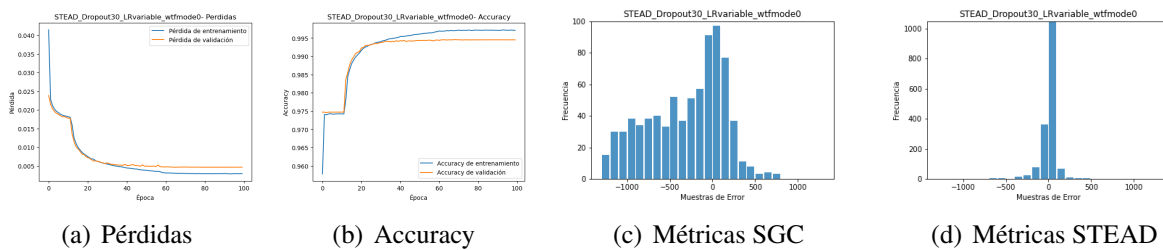


Figura 26. Fase Experimental 3: Entrenamiento STEAD Entrada 0.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

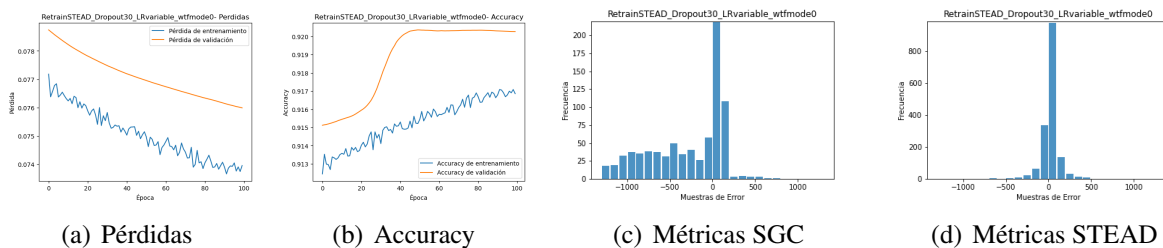


Figura 27. Fase Experimental 3: Reentrenamiento Entrada 0.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

En esta sección, se definen unas entradas que está relacionadas con las mencionadas anteriormente:

- 0. Entrada de la red organizada de la siguiente forma: [1 Canal seleccionado al azar, 1 filtro

pasa bajas en el Canal seleccionado, 31 Transformadas Canal seleccionado]. Dimensión total de: (Cantidad de eventos, 6000, 33).

1. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 31 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 31 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 31 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 6000, 99).
2. Entrada de la red organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 9 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 9 Transformadas de Canal HHE, 1 filtro pasa bajas en el Canal HHZ, 9 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 6000, 33).

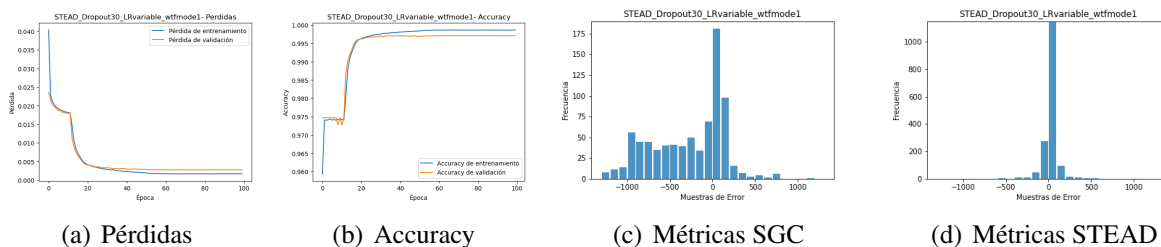


Figura 28. Fase Experimental 3: Entrenamiento STEAD Entrada 1.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Al revisar las gráficas de entrenamiento en las figuras 26 - 31, se observa que las gráficas tienen una forma diferente a las revisadas en los experimentos anteriores aunque en el paso de las épocas estas tienden a resultados similares a los de las etapas experimentales anteriores. Con

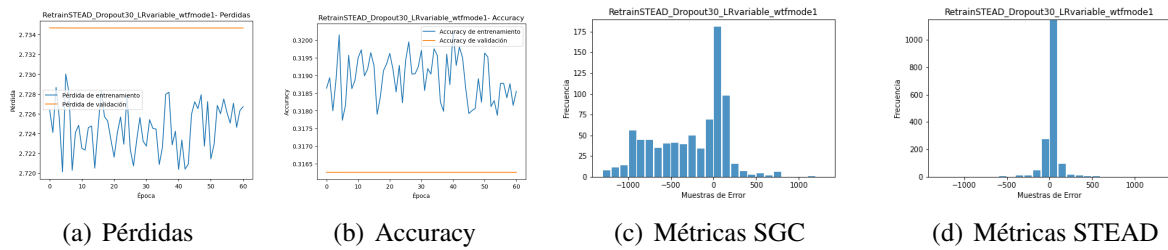


Figura 29. Fase Experimental 3: Reentrenamiento Entrada 1.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

respecto a los diagramas de barras de las figuras mencionadas anteriormente, todos tienen la misma forma que no se asimila a una distribución normal de media cero.

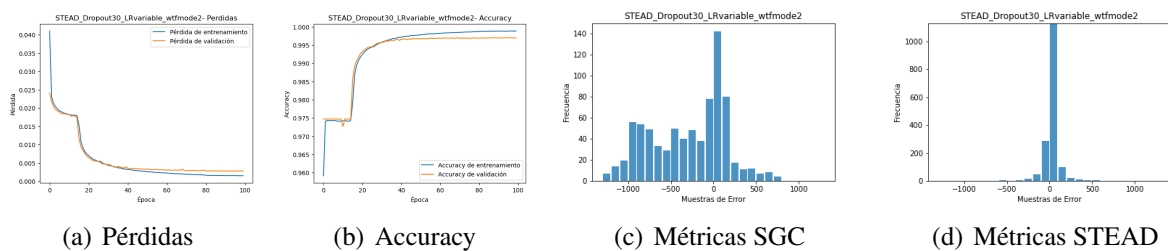


Figura 30. Fase Experimental 3: Entrenamiento STEAD Entrada 2.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Finalmente, luego de analizar los gráficos mencionados previamente, se revisan los resultados numéricos de error absoluto promedio y desviación estándar en muestras de la tabla 14, se observa que estos no implican una mejoría con respecto a los resultados expuestos en las fases de experimentación 1 y 2. Adicionalmente el reentrenamiento no presenta una mejoría notable para los datos de evaluación del SGC demostrando que este tratamiento no es suficiente para que la red aprenda correctamente las características del conjunto de datos utilizado para el reentrenamiento

del modelo.

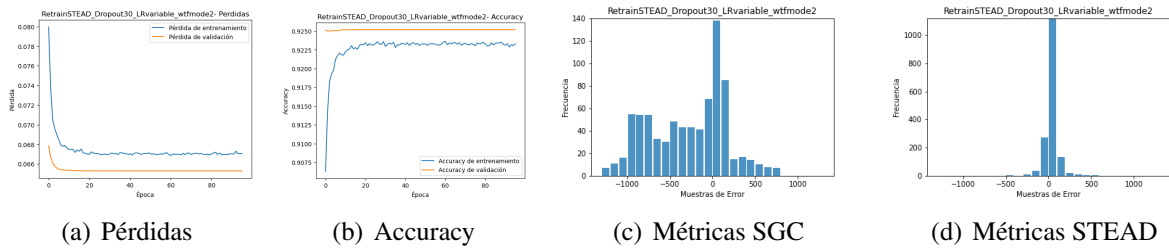


Figura 31. Fase Experimental 3: Reentrenamiento Entrada 2.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

En conclusión, realizar un proceso de entrenamiento con datos del STEAD sumado con un reentrenamiento utilizando datos del SGC no representa una mejoría en el proceso de predicción de la red neuronal. Por esta razón, se decide no seguir utilizando la estrategia de entrenamiento mencionada e implementada en esta fase para el siguiente proceso de experimentación.

Tabla 14

Resultados Fase 3

	SGC Error absoluto promedio	SGC Desviación estándar	STEAD Error absoluto promedio	STEAD Desviación estándar
Entrenamiento STEAD Entrada 0	444	394	37	82
Reentrenamiento Entrada 0	391	390	46	91
Entrenamiento STEAD Entrada 1	382	355	28	72
Reentrenamiento Entrada 1	382	355	28	71
Entrenamiento STEAD Entrada 2	422	369	29	71
Reentrenamiento Entrada 2	424	361	29	70

Nota. Todos los valores numéricos son en número de muestras.

5.2.1.4. Fase experimental 4. Para esta última fase experimental, se utiliza la misma arquitectura de red de las fases anteriores y se traen las características que representaron una mejoría en las secciones: se usan capas de *dropout* de 30% y un único proceso de datos utilizando los dos conjuntos con la misma cantidad de eventos en cada uno. Adicionalmente, se realiza la prueba en

Tabla 15
Conjuntos de datos utilizados en esta fase experimental 4

	Características	Número de eventos(entrenamiento, validación y evaluación)
SGC	Datos reales crudos con magnitud mayor o igual a 2.5.	2436, 812, 812
STEAD	Eventos locales con magnitud mayor o igual a 2, con picado manual, autpick y automatic.	2436, 812, 812

Tabla 16
Experimentos realizados en la fase experimental 4

Nombre del experimento	Set de datos utilizado
Entrenamiento Final Entrada 0	SGC y STEAD
Entrenamiento Final Entrada 1	SGC y STEAD
Entrenamiento Final Entrada 2	SGC y STEAD

las tres entradas implementadas en la fase anterior con 6000 muestras por evento, se utilizan los 4060 datos preseleccionados del SGC y 4060 eventos de STEAD que recogen todos los tipos de eventos sísmológicos locales que este conjunto de datos contiene. En las tablas 15 y 16 se presentan respectivamente las características de los conjuntos de datos y los datos utilizados en los procesos de entrenamiento, validación y evaluación de los experimentos realizados en esta fase.

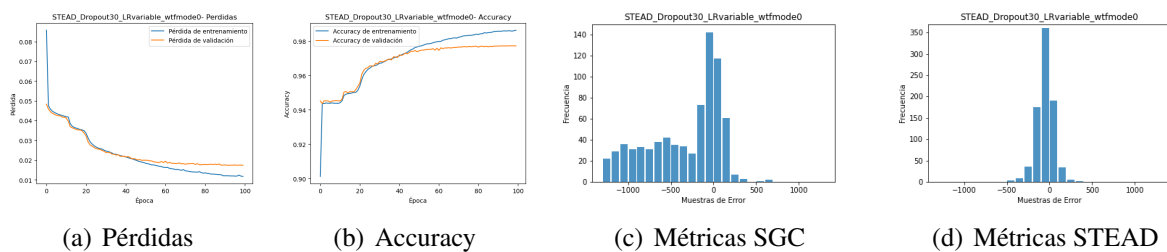


Figura 32. Fase Experimental 4: Entrenamiento Final Entrada 0.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Las gráficas de entrenamiento presentadas en las figuras 32 - 34, se comparan con los

mejores experimentos de las fases anteriores, que son los expuestos en la fase 2 de la entrada 1 y 2, y se observa que consiguen llegar a menores valores en la pérdida y mayores valores de accuracy tanto para la línea de entrenamiento como la de validación.

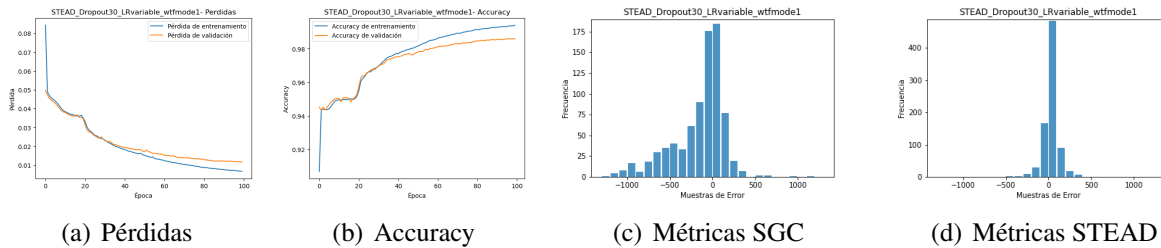


Figura 33. Fase Experimental 4: Entrenamiento Final Entrada 1.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

En las figuras mencionadas previamente se observan diagramas de barras que describen el comportamiento del error, encontrando que la mayoría de los errores se encuentran en el lado negativo de la gráfica, esto implica que la red, con el método de post-procesamiento implementado en este trabajo, está prediciendo los primeros arribos para el conjunto de datos del SGC antes que los reales.

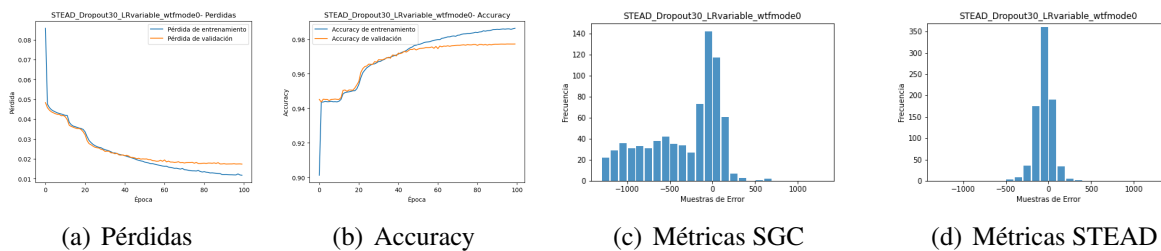


Figura 34. Fase Experimental 4: Entrenamiento Final Entrada 2.

Nota: Los diagramas de barras representan las muestras de error entre la predicción y la etiqueta real, cada barra representa 100 muestras.

Los datos de la tabla 17 indican que el menor error absoluto promedio en muestras de todo el trabajo ocurre cuando se utiliza la entrada 1, si se compara este con los otros resultados de esta misma fase, se confirma que el uso de 31 transformadas wavelet produce un resultado óptimo comparado con el uso de menos transformadas. Adicionalmente reafirma lo comprobado en las fases previas evidenciando que el uso de los tres canales de información en los procesos de entrenamiento, validación y evaluación implica una mejora con respecto al uso de un solo canal seleccionado al azar.

Tabla 17
Resultados Fase 4

	SGC Error absoluto promedio	SGC Desviación estándar	STEAD Error absoluto promedio	STEAD Desviación estándar
Entrenamiento final Entrada 0	455	456	112	87
Entrenamiento final Entrada 1	227	254	53	80
Entrenamiento final Entrada 2	356	397	60	89

Nota. Todos los valores numéricos son en número de muestras.

5.3. Sistema resultado

En esta sección se presenta la arquitectura final de la red neuronal que tuvo mejores resultados en la fase de experimentación y que recoge las características que probaron mejorar la predicción de la red para el conjunto de datos del SGC, ver figura 35. Adicionalmente, los resultados numéricos son los presentados en la tabla 17 en la fila resaltada en color verde. Finalmente, se presentan unos ejemplos de predicción de la red de los datos de evaluación.

A continuación se listan las características de esta red y del proceso de entrenamiento:

- El set de datos utilizado es una suma de los 4060 datos preseleccionados del SGC que cuen-

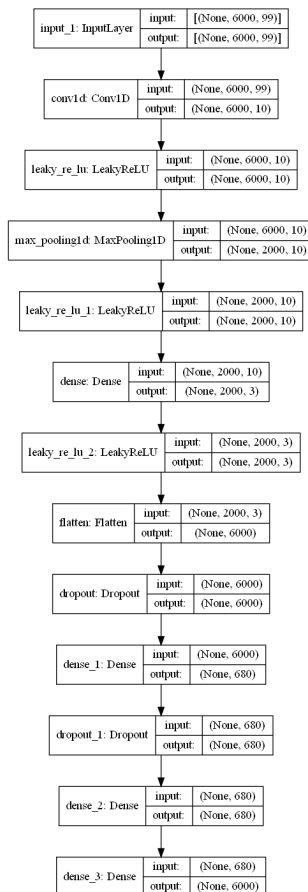


Figura 35. Arquitectura CNN final para la detección de primeros arribos

tan con una magnitud mayor a 2.5, junto con una selección de la misma cantidad de datos de STEAD que recogen las tres características de los eventos sísmológicos locales registrados en la base de datos.

- La entrada de la red está organizada de la siguiente forma: [3 canales, 1 filtro pasa bajas en el Canal HHE, 31 Transformadas del Canal HHE, 1 filtro pasa bajas en el Canal HHN, 31 Transformadas del Canal HHN, 1 filtro pasa bajas en el Canal HHZ, 31 Transformadas del Canal HHZ]. Dimensión total de: (Cantidad de eventos, 6000, 99).

- Capas de *dropout* de 30%, 100 épocas y *Learning Rate* variable.
- La salida de la red es un vector de 6000 muestras.

En la figura 36(a) se presenta un caso óptimo donde el primer arribo real y el predicho con la red son el mismo, asimismo en la figura 36(b) evidencia un caso problema para el conjunto de datos de STEAD. Analizando estos casos los dos eventos tienen una apariencia similar, es por esto que los resultados para este grupo de datos tienen valores más pequeños comparados con los del SGC.

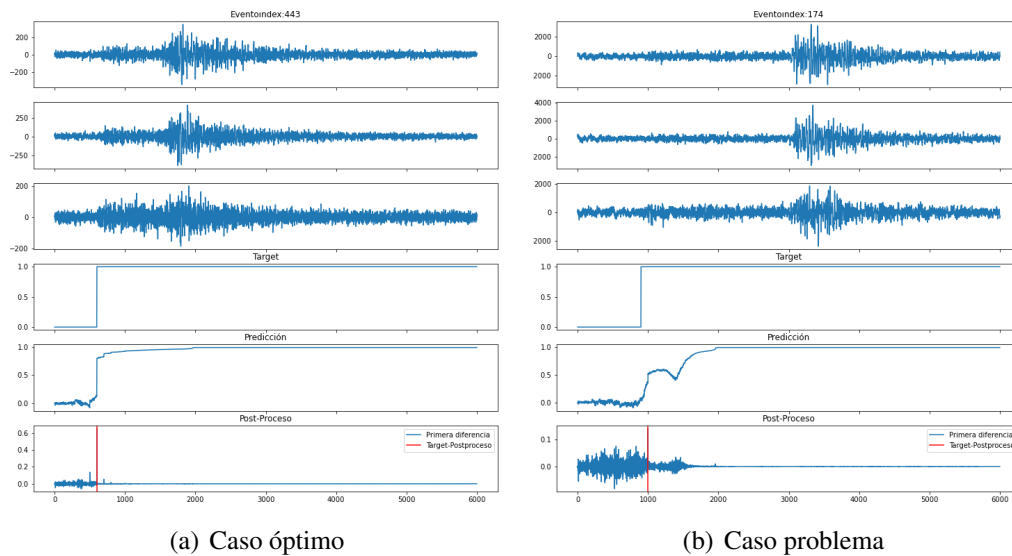


Figura 36. Ejemplo de predicción de la red neuronal final para datos del STEAD

Finalmente, se observa unos casos óptimos en la figura 37 y unos casos problema en la figura 38 para el conjunto de datos del SGC. Al comparar estos dos eventos con los revisados anteriormente para STEAD, se encuentra que estos no siempre tienen la misma apariencia y que en algunos casos, como se puede ver en las figuras 38(a), 38(b) y 38(c), alguno o varios de los canales

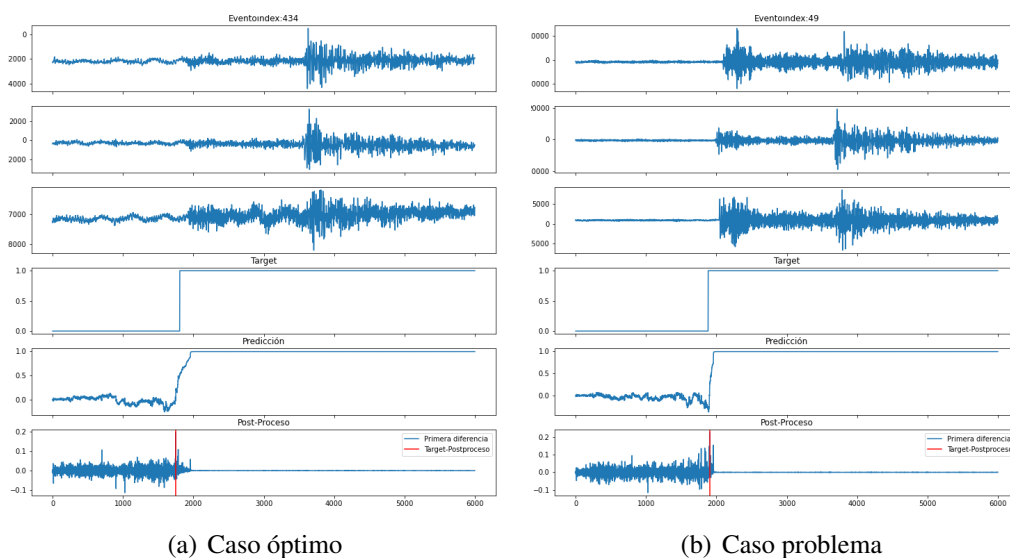


Figura 37. Ejemplo de predicción de la red neuronal final para datos óptimos del SGC

no presenta una forma de onda sencilla de procesar lo que genera una predicción prematura del primer arribo.

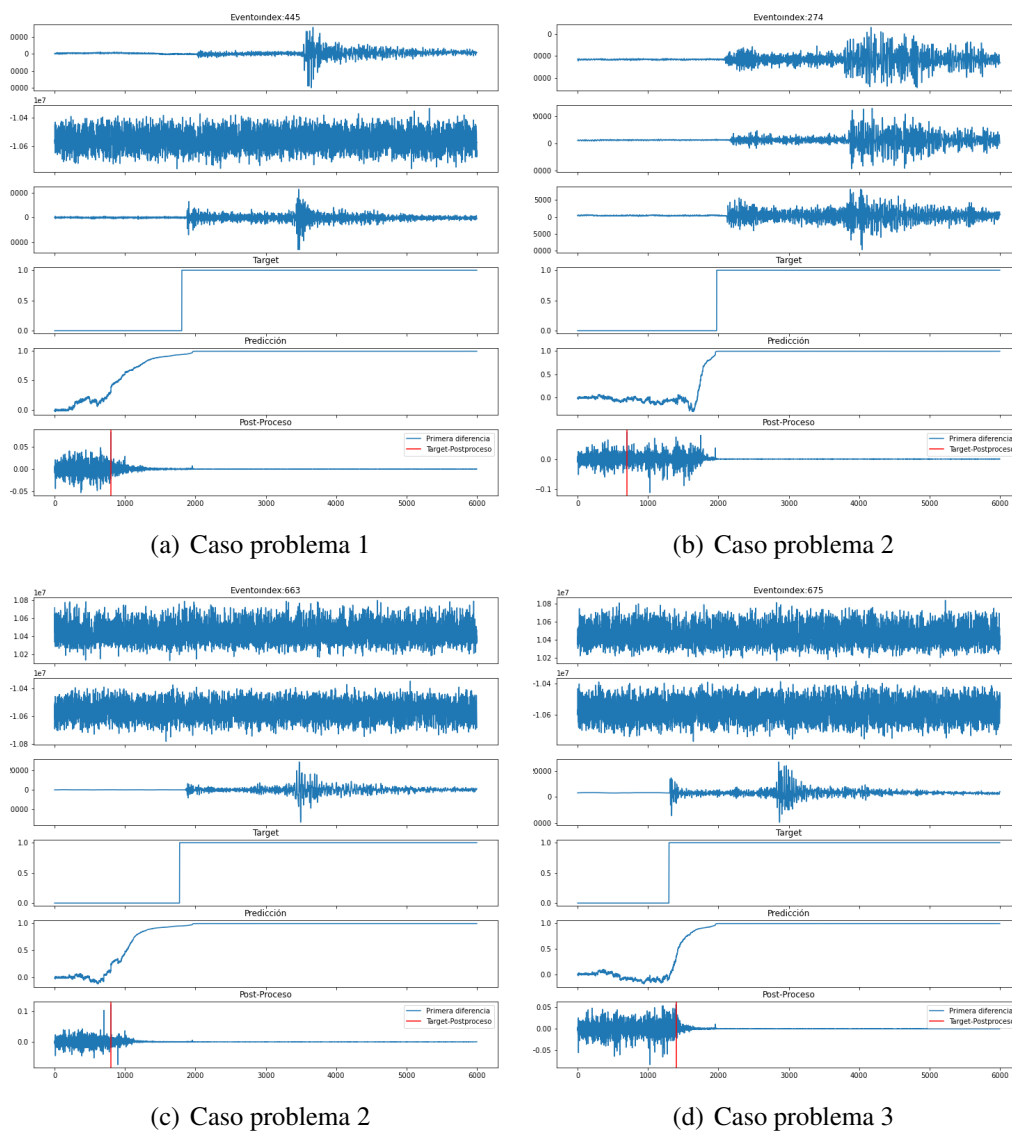


Figura 38. Ejemplo de predicción de la red neuronal final para datos problema del SGC

6. Conclusiones

En este documento se presentaron los acercamientos realizados a la arquitectura de la red neuronal expuesta en Zambrano y Cachopo (2022) para la detección automática de primeros arribos en trazas sismológicas terrestres reales del catálogo sísmico colombiano. Con este fin se entrenaron, validaron y evaluaron diferentes modelos generados bajo diferentes condiciones de entrenamiento, en las que se variaron las dimensiones de la entrada y la salida, se cambió el número de transformadas Wavelet generadas, se implementó el uso de eventos sísmicos de tres dimensiones y se comprobó cómo la calidad y cantidad de datos utilizados para este proceso afecta la predicción de los primeros arribos.

Estas variaciones dieron como resultado una red neuronal con una entrada de 6000 muestras con 99 series de datos en los cuales se incluyen: 3 canales del evento, 1 filtro pasa bajas por cada canal y 31 transformadas wavelet por canal. Una secuencia de capas ocultas conformadas por una capa convolucional 1D, maxpooling 1D, capa flatten y dos capas densas, en donde se adicionan unas capas de dropout del 30%. Los resultados obtenidos con la implementación mencionada anteriormente, para el conjunto de datos del Catálogo Sísmico Colombiano, son de 227 muestras de error absoluto promedio y 254 muestras de desviación estándar, donde se contemplan las 100 muestras de error intrínsecas que suman la precisión de la estampa de tiempo del etiquetado del

primer arribo por parte del Servicio Geológico Colombiano, esto implica un error absoluto promedio en la predicción de del primer arribo de aproximadamente 2.3 segundos con una variación de más o menos 2.5 segundos.

Se recopiló un conjunto de grabaciones sismológicas reales a partir de eventos sismológicos reportados en el Catálogo Sísmico Colombiano en la zona del valle medio del Magdalena, este conjunto de datos fue construido mediante una descarga masiva de eventos que luego fueron correlacionados a partir de sus meta-datos con estampas de tiempo suministradas por el Servicio Geológico Colombiano que permitieron añadir el etiquetado del momento de ruptura de la onda P. Este proceso fue parte necesaria para este proyecto ya que no se contaba con una recopilación de grabaciones reales de la zona de interés que pudieran ser usadas en procesos de entrenamiento, validación y evaluación de aplicaciones de *Machine Learning*.

Con el fin de mejorar los resultados, se incluyó el conjunto de datos STEAD en cada etapa, al cual se le agregó un piso de ruido (SNR) entre 6 a 10 dB generado aleatoriamente, para incrementar el número de datos disponibles que contaran con unas características similares a los seleccionados en el conjunto de datos del SGC. La implementación de estos datos en las etapas de entrenamiento y validación permiten mejorar los resultados obtenidos reduciendo la desviación estándar ya que aportan características que no son sencillas de identificar en los datos crudos del SGC.

En la fase experimental se comprobó que el uso de tres dimensiones de información de

mayor cantidad de muestras sumado con una serie de transformadas wavelet mejora la predicción del modelo, adicionalmente, se verificó que el uso de 31 transformadas por cada canal de información es un número adecuado debido a que cuando se utilizaron cantidades diferentes, mayores y menores, se evidenció una afectación negativa en las métricas calculadas para el conjunto de datos de evaluación. Finalmente, se evidenció que la selección y uso de eventos de mayor magnitud para el conjunto de datos del SGC genera beneficios en los procesos de entrenamiento, validación y evaluación de la arquitectura de red estudiada en este documento.

La solución de problemas utilizando redes neuronales implica un amplio campo de posibilidades y acercamientos con respecto a lo que se busca solucionar, estas dependen de los diferentes recursos físicos y temporales, las perspectivas y los ambientes en los que cada diseño se conforma. Teniendo en cuenta lo mencionado anteriormente y recogiendo los resultados expuestos previamente en este documento, el sistema basado en redes neuronales artificiales resultante en este documento (que se conforma por pre-procesamiento, arquitectura de red y post-procesamiento) no se considera una solución definitiva al problema de la predicción de primeros arribos en trazas sísmológicas, sino un resultado que se basa en la experiencia de nosotros como diseñadores.

7. Recomendaciones

- Para el conjunto de datos del Servicio Geológico Colombiano se recomienda realizar un proceso de clasificación de trazas sismológicas diferente a la selección mediante la amplitud en la escala de Richter, con el fin de aprovechar el gran número de eventos de baja magnitud que son grabados y reportados por el Servicio Geológico Colombiano en el valle medio del Magdalena, con la finalidad de incrementar la cantidad de datos sísmicos a usar para los procesos de entramiento, validación y evaluación en el desarrollo de sistemas de *Machine Learning*.
- Para procesos posteriores también es posible tomar el modelo entrenado en este proyecto y someterlo a diferentes procesos de reentrenamiento, como algoritmos Multi Armed Bandit (MAB), presentando nuevas grabaciones recopiladas con el fin de generar un modelo a partir del modelo pre entrenado que mejore los resultados en la predicción.
- En la realización de este trabajo se implementó un escalón unitario, el cuál presentaba un cambio repentino de cero a uno en el número de muestra del valor original del primer arribo, como etiqueta para el entrenamiento y validación de la red neuronal. Teniendo en cuenta lo anterior, el uso de un tipo diferente de etiquetado como puede ser un impulso unitario o un tipo de trayectoria con un cambio suavizado, como puede ser una campana de Gauss con

variación en el ancho, puede ser benéfico para la predicción del sistema.

- El método de post-procesamiento es una parte fundamental de la red neuronal, pues este convierte la salida de la red en un valor más sencillo de comparar para el usuario, por lo expresado, la selección de un método puede implicar un aumento en el error del sistema. Teniendo en cuenta lo anterior, el uso de otro método de post-procesamiento diferente al cálculo de la primera diferencia utilizado en este trabajo, podría implicar una mejoría en los resultados de predicción de la red sin la necesidad de realizar otro proceso de entrenamiento.

Referencias Bibliográficas

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., & Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Al-Mashhor, A. A., Al-Shuhail, A. A., Hanafy, S. M., & Mousa, W. A. (2019). First arrival picking of seismic data based on trace envelope. *IEEE Access*, 7:128806–128815.
- Anaconda (2022). Anaconda software distribution.
- Brownlee, J. (2021). How to update neural network models with more data. =<https://machinelearningmastery.com/update-neural-network-models-with-more-data/>.
- Burrus, C. S., Gopinath, R. A., & Guo, H. (1998). *Introduction to wavelets and wavelet transforms : a primer*. Prentice Hall.
- Education, A. (2021). MI model optimization with ensemble learning, retraining.

=<https://www.techtarget.com/searchenterpriseai/post/ML-model-optimization-with-ensemble-learning-retraining>.

Geron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc., 1st edición.

Klabjan, D. & Zhu, X. (2020). Neural network retraining for model serving.

Krohn, C. E. (1984). Geophone ground coupling. *Geophysics*, 49(6):722–731.

Lee, G. R., Gommers, R., Waselewski, F., Wohlfahrt, K., & O8217;Leary, A. (2019). Pywavelets: A python package for wavelet analysis. *Journal of Open Source Software*, 4(36):1237.

Mexicano, S. G. (s.f.). *¿Qué es la Geofísica?* pp. 1–2. <https://www.sgm.gob.mx>.

Mousavi, S. M., Sheng, Y., Zhu, W., & Beroza, G. C. (2019). Stanford earthquake dataset (stead): A global data set of seismic signals for ai. *IEEE Access*, 7:179464–179476.

Musson, R. (2012). Intensity and intensity scales. *New manual of seismological observatory practice 2*, pp. 1–41.

Romero, J. E., Titos, M., Ángel Bueno, Álvarez, I., García, L., Ángel de la Torre, & Benítez, M. C. (2016). Apasvo: A free software tool for automatic p-phase picking and event detection in seismic traces. *Computers and Geosciences*, 90:213–220.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation.

Servicio Geológico Colombiano, S. (2022). Catálogo Sísmico Colombiano. <http://bdrsnc.sgc.gov.co/paginas1/catalogo/index.php>. [Online].

Slivkins, A. (2019). Introduction to multi-armed bandits. *Foundations and Trends in Machine Learning*, 12:1–286.

The HDF Group (2000-2010). Hierarchical data format version 5.

Tsai, K. C., Hu, W., Wu, X., Chen, J., & Han, Z. (2020). Automatic first arrival picking via deep learning with human interactive learning. *IEEE Transactions on Geoscience and Remote Sensing*, 58:1380–1391.

Zambrano, S. J. R. & Cachopo, O. F. R. (2022). Detección automática de primeros arribos en trazas sintéticas terrestres del valle medio del Magdalena usando redes neuronales artificiales.

Zhu, W. & Beroza, G. C. (2019). Phasenet: A deep-neural-network-based seismic arrival-time picking method. *Geophysical Journal International*, 216:261–273.

Apéndices

A. Códigos de programación implementados

La implementación de lo expuesto durante este trabajo se encuentra alojado en el repositorio de GitLab del Grupo de Investigación en Conectividad y Procesamiento de Señales (CPS), en donde se pueden encontrar los archivos desarrollados e implementados para la generación del conjunto de datos del Servicio Geológico Colombiano, junto con una librería desarrollada por los autores junto con la asesoría y orientación del director y codirector del presente proyecto.

El desarrollo de este proyecto fue propuesto usando la versión de Python 3.8, bajo un ambiente de Anaconda, el cual fue previamente puesto a punto para hacer uso de las tarjetas gráficas disponibles en el clúster del grupo CPS. Los diferentes resultados y modelos generados se encuentran alojados en el usuario otorgado a los autores para ingresar al clúster del grupo CPS.

Para acceder al repositorio de este proyecto es necesario la previa autorización por parte del grupo CPS y puede ser clonado posterior a la autorización mediante el siguiente enlace https://gitlab.com/CPS_Repo/Seismic/semillerogan2020/fb-seismology.git junto con la herramienta git.