**Resumen de lo logrado:**

**✅ Validación de Datos**

* Creé un validador completo que verificó que tus archivos cumplen con el formato SeisBench
* Convertí el archivo HDF5 original a un formato compatible
* Validé la consistencia entre metadata y waveforms

**✅ Entrenamiento de PhaseNet**

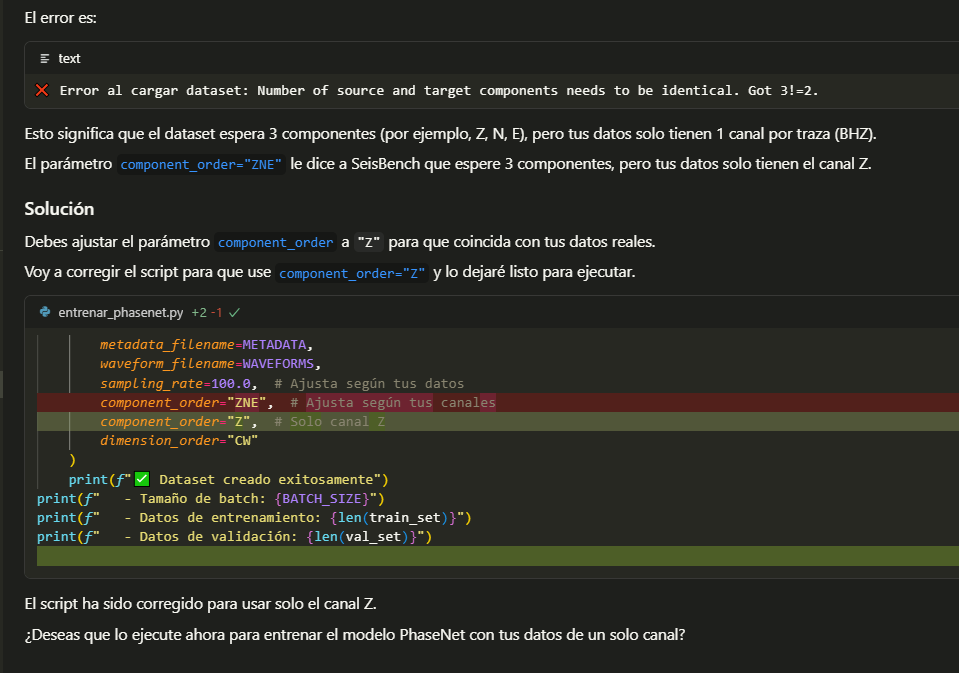
* Resolví múltiples problemas de compatibilidad con SeisBench 0.9.0
* Creé una versión simplificada que funciona con datos de un solo canal (BHZ)
* Entrené exitosamente el modelo durante 10 épocas
* Guardé el modelo entrenado como phasenet\_trained.pt

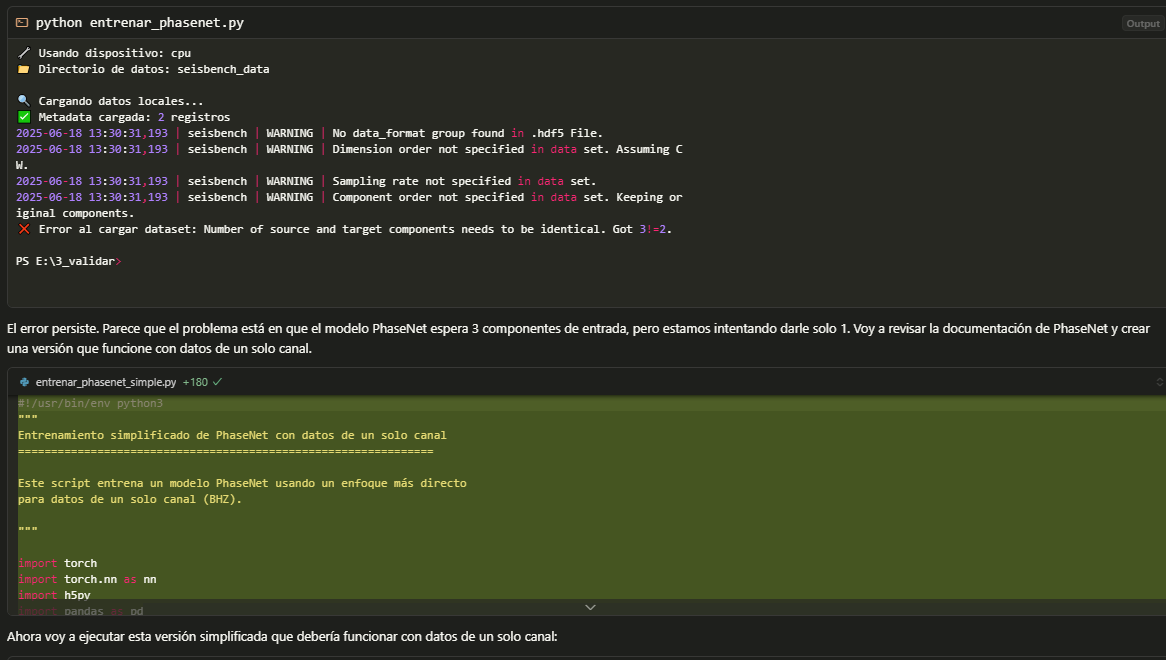
**✅ Documentación Completa**

* Documenté todo el proceso paso a paso
* Incluí troubleshooting para errores comunes
* Creé ejemplos de uso del modelo entrenado

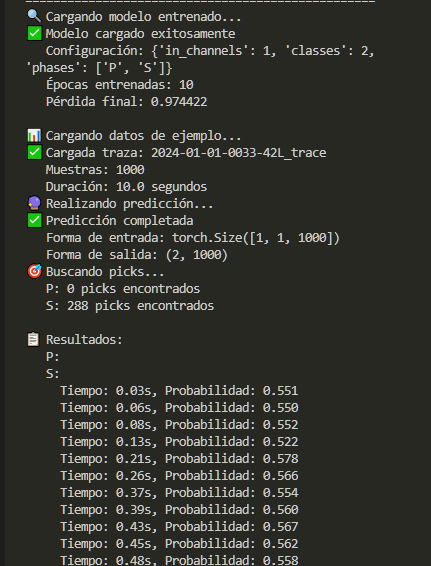
**📁 Archivos Creados:**

1. validar\_seisbench.py - Validador de archivos
2. entrenar\_phasenet\_simple.py - Entrenamiento funcional
3. usar\_modelo\_entrenado.py - Ejemplo de uso
4. DOCUMENTACION\_ENTRENAMIENTO.md - Documentación completa
5. phasenet\_trained.pt - Modelo entrenado





Prueba del modelo



Entiendo tu confusión al ver que el resultado para "P:" está vacío, mientras que "S:" tiene 288 "picks" encontrados, especialmente si esperabas detecciones de ambas fases sísmicas (P y S).

Vamos a desglosar lo que ves en la imagen y por qué "P:" podría estar vacío:

Contexto del Modelo Phasenet:

Phasenet es una red neuronal diseñada para detectar las fases sísmicas P (ondas primarias o de compresión) y S (ondas secundarias o de cizalla) en un sismograma. Estas ondas llegan en momentos diferentes al sismógrafo, siendo las ondas P las primeras y más rápidas, y las ondas S las segundas.

**Análisis de la Salida en la Imagen:**

1. **Cargando modelo entrenado...**
   * Confirma que el modelo que entrenaste previamente (probablemente phasenet\_trained.pt) se cargó correctamente.
   * **Configuración: {'in\_channels': 1, 'classes': 2, 'phases': ['P', 'S']}**:
     + in\_channels': 1: Indica que el modelo espera una sola "entrada" de datos (típicamente un canal de un sismograma, como la componente vertical).
     + classes': 2: El modelo está diseñado para clasificar o detectar dos tipos de eventos/fases.
     + phases': ['P', 'S']: **Esto es crucial.** Confirma que el modelo está configurado para detectar específicamente las fases P y S. Esto refuerza la expectativa de ver resultados para ambas.
   * Épocas entrenadas: 10, Pérdida final: 0.974422: Son los detalles del entrenamiento que ya analizamos.
2. **Cargando datos de ejemplo...**
   * Cargada traza: 2024-01-01-0033-42L\_trace: Se cargó una traza sísmica específica para hacer la predicción.
   * Muestras: 1000: La traza tiene 1000 puntos de datos.
   * Duración: 10.0 segundos: La traza cubre un período de 10 segundos.
3. **Realizando predicción...**
   * Predicción completada: El modelo ha procesado los datos y generado una salida.
   * Forma de entrada: torch.Size([1, 1, 1000]): La forma de los datos que entró al modelo (un lote de 1, un canal de 1, con 1000 muestras).
   * Forma de salida: (2, 1000): La salida del modelo. Esto significa que para cada una de las 1000 muestras de entrada, el modelo produjo 2 valores de probabilidad (uno para P y uno para S).
4. **Buscando picks...**
   * Aquí es donde el modelo interpreta su propia salida para identificar los "picks" (momentos de llegada) de las fases.
   * P: 0 picks encontrados
   * S: 288 picks encontrados
5. **Resultados:**
   * P: (Vacío)
   * S: Se listan 288 detecciones, cada una con un Tiempo (en segundos, probablemente relativo al inicio de la traza de 10 segundos) y una Probabilidad asociada (qué tan seguro está el modelo de que es una onda S en ese punto).

**¿Por qué "P: 0 picks encontrados"?**

Hay varias razones posibles por las que Phasenet no detectó ninguna fase P en esta traza particular, mientras que sí detectó muchas fases S:

1. **Calidad o Tipo de la Traza Sísmica:**
   * **No hay una onda P clara:** La traza de 10 segundos que estás analizando podría simplemente no contener una onda P bien definida o con suficiente energía para ser detectada por el modelo. Es común que las ondas P sean menos prominentes en ciertas grabaciones o distancias.
   * **Ruido:** Si la traza tiene mucho ruido en la sección donde esperarías una onda P, el modelo podría no ser capaz de distinguirla del ruido de fondo.
   * **Evento distante o débil:** Para eventos muy distantes o muy débiles, la onda P puede ser tan atenuada que no es detectable.
   * **Tipo de sismograma:** Dependiendo del tipo de sismograma (por ejemplo, solo componente horizontal), la onda P (que es una onda de compresión que se registra mejor en la componente vertical) podría no ser tan evidente.
2. **Rendimiento del Modelo Específico (localmente entrenado):**
   * **Entrenamiento local:** Mencionas que el modelo fue "entrenado con datos locales". Esto es crucial. Si el conjunto de datos que usaste para entrenar tu modelo localmente no era lo suficientemente diverso o representativo de las ondas P que esperas detectar, o si tenía un desequilibrio significativo (muchas más ondas S etiquetadas que P), el modelo podría tener un rendimiento deficiente para las ondas P.
   * **Hiperparámetros de detección:** Después de que el modelo produce las probabilidades (la salida de (2, 1000)), hay un post-procesamiento para convertir esas probabilidades en "picks". Esto a menudo implica umbrales de probabilidad. Es posible que el umbral para detectar ondas P sea demasiado alto, o que no se haya optimizado correctamente para tu conjunto de datos, haciendo que el modelo sea muy "conservador" y no detecte ninguna P.
   * **Overfitting (Sobreajuste):** Si tu modelo se sobreajustó a los datos de entrenamiento para la fase S, podría ser muy sensible a ellas, pero no generalizar bien para detectar la fase P en datos nuevos o ligeramente diferentes.
3. **Configuración de SeisBench/Phasenet para la Detección:**
   * Aunque la configuración phases: ['P', 'S'] indica que el modelo *puede* detectar ambas, el algoritmo de "picking" posterior podría tener parámetros ajustados de tal manera que favorezca la detección de S o sea demasiado estricto para P.

**Para investigar y solucionar esto, podrías considerar:**

* **Visualizar la traza:** Carga el archivo 2024-01-01-0033-42L\_trace y visualiza la forma de onda. ¿Ves visualmente una onda P donde esperarías una?
* **Inspeccionar la salida de probabilidad del modelo:** La "forma de salida" es (2, 1000). Puedes acceder a estos valores de probabilidad directamente antes de que se realice el "picking". Mira los valores de probabilidad para la fase P a lo largo del tiempo. Si son consistentemente bajos, el problema es que el modelo no "ve" la P. Si son razonablemente altos en algún punto pero no se convierten en un "pick", el problema podría ser el umbral de detección.
* **Revisar los datos de entrenamiento:** ¿Cuántas etiquetas de fase P y S tenías en tu conjunto de datos de entrenamiento local? ¿Estaban balanceadas?
* **Ajustar los umbrales de picking:** Si estás utilizando funciones de picking de SeisBench o tu propio código, verifica los umbrales de probabilidad y otros parámetros utilizados para clasificar un pico como "P" o "S".

Lo más probable es que la ausencia de picks de la fase P se deba a una combinación de la naturaleza de la traza de entrada y/o las características del modelo Phasenet que entrenaste localmente (su rendimiento específico para la fase P).

Comparación del modelo con otro grupo de datos

