

hyperparam-search

March 18, 2024

```
[ ]: # This Python 3 environment comes with many helpful analytics libraries
      ↳ installed
      # It is defined by the kaggle/python Docker image: https://github.com/kaggle/
      ↳ docker-python
      # For example, here's several helpful packages to load

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the read-only "../input/" directory
# For example, running this (by clicking run or pressing Shift+Enter) will list
↳ all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# You can write up to 20GB to the current directory (/kaggle/working/) that
↳ gets preserved as output when you create a version using "Save & Run All"
# You can also write temporary files to /kaggle/temp/, but they won't be saved
↳ outside of the current session
```

```
[1]: !pip install torch-geometric optuna torchmetrics
```

Collecting torch-geometric

Downloading torch_geometric-2.5.0-py3-none-any.whl.metadata (64 kB)

64.2/64.2 kB

1.5 MB/s eta 0:00:00a 0:00:01

Requirement already satisfied: optuna in /opt/conda/lib/python3.10/site-packages (3.5.0)

Requirement already satisfied: torchmetrics in /opt/conda/lib/python3.10/site-packages (1.3.1)

Requirement already satisfied: tqdm in /opt/conda/lib/python3.10/site-packages (from torch-geometric) (4.66.1)

Requirement already satisfied: numpy in /opt/conda/lib/python3.10/site-packages (from torch-geometric) (1.26.4)

Requirement already satisfied: scipy in /opt/conda/lib/python3.10/site-packages

(from torch-geometric) (1.11.4)
 Requirement already satisfied: fsspec in /opt/conda/lib/python3.10/site-packages
 (from torch-geometric) (2024.2.0)
 Requirement already satisfied: jinja2 in /opt/conda/lib/python3.10/site-packages
 (from torch-geometric) (3.1.2)
 Requirement already satisfied: aiohttp in /opt/conda/lib/python3.10/site-
 packages (from torch-geometric) (3.9.1)
 Requirement already satisfied: requests in /opt/conda/lib/python3.10/site-
 packages (from torch-geometric) (2.31.0)
 Requirement already satisfied: pyparsing in /opt/conda/lib/python3.10/site-
 packages (from torch-geometric) (3.1.1)
 Requirement already satisfied: scikit-learn in /opt/conda/lib/python3.10/site-
 packages (from torch-geometric) (1.2.2)
 Requirement already satisfied: psutil>=5.8.0 in /opt/conda/lib/python3.10/site-
 packages (from torch-geometric) (5.9.3)
 Requirement already satisfied: alembic>=1.5.0 in /opt/conda/lib/python3.10/site-
 packages (from optuna) (1.13.1)
 Requirement already satisfied: colorlog in /opt/conda/lib/python3.10/site-
 packages (from optuna) (6.8.2)
 Requirement already satisfied: packaging>=20.0 in
 /opt/conda/lib/python3.10/site-packages (from optuna) (21.3)
 Requirement already satisfied: sqlalchemy>=1.3.0 in
 /opt/conda/lib/python3.10/site-packages (from optuna) (2.0.25)
 Requirement already satisfied: PyYAML in /opt/conda/lib/python3.10/site-packages
 (from optuna) (6.0.1)
 Requirement already satisfied: torch>=1.10.0 in /opt/conda/lib/python3.10/site-
 packages (from torchmetrics) (2.1.2)
 Requirement already satisfied: lightning-utilities>=0.8.0 in
 /opt/conda/lib/python3.10/site-packages (from torchmetrics) (0.10.1)
 Requirement already satisfied: Mako in /opt/conda/lib/python3.10/site-packages
 (from alembic>=1.5.0->optuna) (1.3.2)
 Requirement already satisfied: typing-extensions>=4 in
 /opt/conda/lib/python3.10/site-packages (from alembic>=1.5.0->optuna) (4.9.0)
 Requirement already satisfied: setuptools in /opt/conda/lib/python3.10/site-
 packages (from lightning-utilities>=0.8.0->torchmetrics) (69.0.3)
 Requirement already satisfied: greenlet!=0.4.17 in
 /opt/conda/lib/python3.10/site-packages (from sqlalchemy>=1.3.0->optuna) (3.0.3)
 Requirement already satisfied: filelock in /opt/conda/lib/python3.10/site-
 packages (from torch>=1.10.0->torchmetrics) (3.13.1)
 Requirement already satisfied: sympy in /opt/conda/lib/python3.10/site-packages
 (from torch>=1.10.0->torchmetrics) (1.12)
 Requirement already satisfied: networkx in /opt/conda/lib/python3.10/site-
 packages (from torch>=1.10.0->torchmetrics) (3.2.1)
 Requirement already satisfied: attrs>=17.3.0 in /opt/conda/lib/python3.10/site-
 packages (from aiohttp->torch-geometric) (23.2.0)
 Requirement already satisfied: multidict<7.0,>=4.5 in
 /opt/conda/lib/python3.10/site-packages (from aiohttp->torch-geometric) (6.0.4)
 Requirement already satisfied: yarll<2.0,>=1.0 in /opt/conda/lib/python3.10/site-

```

packages (from aiohttp->torch-geometric) (1.9.3)
Requirement already satisfied: frozenlist>=1.1.1 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->torch-geometric) (1.4.1)
Requirement already satisfied: aiosignal>=1.1.2 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->torch-geometric) (1.3.1)
Requirement already satisfied: async-timeout<5.0,>=4.0 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->torch-geometric) (4.0.3)
Requirement already satisfied: MarkupSafe>=2.0 in
/opt/conda/lib/python3.10/site-packages (from jinja2->torch-geometric) (2.1.3)
Requirement already satisfied: charset-normalizer<4,>=2 in
/opt/conda/lib/python3.10/site-packages (from requests->torch-geometric) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-
packages (from requests->torch-geometric) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/opt/conda/lib/python3.10/site-packages (from requests->torch-geometric)
(1.26.18)
Requirement already satisfied: certifi>=2017.4.17 in
/opt/conda/lib/python3.10/site-packages (from requests->torch-geometric)
(2024.2.2)
Requirement already satisfied: joblib>=1.1.1 in /opt/conda/lib/python3.10/site-
packages (from scikit-learn->torch-geometric) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/opt/conda/lib/python3.10/site-packages (from scikit-learn->torch-geometric)
(3.2.0)
Requirement already satisfied: mpmath>=0.19 in /opt/conda/lib/python3.10/site-
packages (from sympy->torch>=1.10.0->torchmetrics) (1.3.0)
Downloading torch_geometric-2.5.0-py3-none-any.whl (1.1 MB)
1.1/1.1 MB
15.0 MB/s eta 0:00:0000:010:01
Installing collected packages: torch-geometric
Successfully installed torch-geometric-2.5.0

```

```

[2]: import torch
chunk1 = torch.load('/kaggle/input/sci-data-graph/first.pt')
chunk2 = torch.load('/kaggle/input/sci-data-graph/second (1).pt')
chunk1+=chunk2
del chunk2

```

```

[3]: from sklearn.model_selection import train_test_split
rand_seed = 42
X_train, X_test = train_test_split(chunk1, test_size=0.1, random_state =
    ↪rand_seed)
X_train, X_val = train_test_split(X_train, test_size=0.1, random_state =
    ↪rand_seed)
print(len(X_train), len(X_val), len(X_val))

```

14580 1620 1620

```
[4]: from torch_geometric.loader import DataLoader

train_loader = DataLoader(X_train, batch_size=32, shuffle=True)
val_loader = DataLoader(X_val, batch_size=32, shuffle=False)
test_loader = DataLoader(X_test, batch_size=32, shuffle=False)

[ ]: import torch
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torch_geometric.nn import SAGEConv, global_mean_pool
from torch_geometric.data import DataLoader
from torchmetrics.classification import BinaryAUROC
from torchmetrics import AUROC
import optuna

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')

auroc = BinaryAUROC()

class Network(nn.Module):
    def __init__(self, c_in, c_hidden, c_out, p=0.3):
        super(Network, self).__init__()
        torch.manual_seed(123)
        self.conv1 = SAGEConv(c_in, c_hidden, aggr='mean')
        self.conv2 = SAGEConv(c_hidden, 3*c_hidden, aggr='mean')
        self.conv3 = SAGEConv(3*c_hidden, 2*c_hidden, aggr='mean')
        self.conv4 = SAGEConv(2*c_hidden, c_hidden, aggr='mean')
        self.lin1 = nn.Linear(c_hidden, 4*c_out)
        self.lin2 = nn.Linear(4*c_out, c_out)
        self.p = p

    def forward(self, x, edge_index, batch, is_train):
        x = self.conv1(x, edge_index)
        x = x.relu()
        x = self.conv2(x, edge_index)
        x = x.relu()
        x = self.conv3(x, edge_index)
        x = x.relu()
        x = self.conv4(x, edge_index)
        x = global_mean_pool(x, batch)

        x = F.dropout(x, p=self.p, training=is_train)
        x = self.lin1(x)
        x = x.relu()
```

```

        x = F.dropout(x, p=self.p, training=is_train)
        x = self.lin2(x)

        return x

def objective(trial):

    c_hidden = 32
    p = trial.suggest_uniform('p', 0.1, 0.5)
    learning_rate = trial.suggest_loguniform('learning_rate', 1e-5, 1e-2)

    model = Network(c_in=5, c_hidden=c_hidden, c_out=2, p=p).to(device)
    optimizer = optim.Adam(model.parameters(), lr=learning_rate)
    criterion = nn.BCEWithLogitsLoss()

    num_epochs = 10

    for epoch in range(num_epochs):
        model.train()
        epoch_loss = 0
        #         print(epoch)

        for idx, batch in enumerate(train_loader):
            batch = batch.to(device)

            pred = model(batch.x.float(), batch.edge_index, batch.batch, True)
            target = F.one_hot(batch.y, 2).float()
            loss = criterion(pred, target)
            epoch_loss += loss.item()

            optimizer.zero_grad()
            loss.backward()
            optimizer.step()

            #         break
        # break

    # Evaluate on the validation set
    _, _, val_auroc = evaluate(val_loader, model, criterion)

    return val_auroc

def evaluate(loader, model, criterion):
    model.eval()
    total_loss = 0.0
    correct = 0
    total_samples = 0

```

```

all_preds = []
all_labels = []

with torch.no_grad():
    for batch in loader:
        batch.to(device)
        pred = model(batch.x.float(), batch.edge_index, batch.batch, False)
        target = F.one_hot(batch.y, 2).float()
        loss = criterion(pred, target)
        total_loss += loss.item()

        # Calculate accuracy
        pred_labels = torch.softmax(pred, -1).argmax(dim=-1)
        correct += (pred_labels == batch.y).sum().item()
        total_samples += len(batch.y)
        all_labels.append(batch.y)
        all_preds.append(pred_labels)

pred = all_preds[0]
label = all_labels[0]

for p, l in zip(all_preds[1:], all_labels[1:]):
    pred = torch.cat([pred, p])
    label = torch.cat([label, l])
# print(pred, label)
return total_loss / len(loader), correct / total_samples, auroc(pred.cpu(),
↪label.cpu())

# Optimizing hyperparameters with Optuna
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=80)

print('Best trial:')
trial = study.best_trial
print('Value: {}'.format(trial.value))
print('Params: ')
for key, value in trial.params.items():
    print('    {}: {}'.format(key, value))

best_p = trial.params['p']
best_learning_rate = trial.params['learning_rate']

final_model = Network(c_in=5, c_hidden=64, c_out=2, p=best_p).to(device)
final_optimizer = optim.Adam(final_model.parameters(), lr=best_learning_rate)
final_criterion = nn.BCEWithLogitsLoss()

```

```

num_epochs = 100

for epoch in range(num_epochs):
    final_model.train()
    epoch_loss = 0

    for idx, batch in enumerate(train_loader):
        batch = batch.to(device)

        pred = final_model(batch.x.float(), batch.edge_index, batch.batch, True)
        target = F.one_hot(batch.y, 2).float()
        loss = final_criterion(pred, target)
        epoch_loss += loss.item()

        final_optimizer.zero_grad()
        loss.backward()
        final_optimizer.step()

# Evaluate on the test set
test_loss, test_accuracy, test_auroc = evaluate(test_loader, final_model,
        final_criterion)
print(f'Test Loss: {test_loss:.4f}, Test Accuracy: {test_accuracy:.4f}, Test
        AUROC: {test_auroc:.4f}')

```

[I 2024-03-09 20:36:02,213] A new study created in memory with name: no-name-76ab11f1-ae36-4f1a-a547-6a739eb89974
/tmp/ipykernel_34/1793251812.py:50: FutureWarning: suggest_uniform has been deprecated in v3.0.0. This feature will be removed in v6.0.0. See <https://github.com/optuna/optuna/releases/tag/v3.0.0>. Use suggest_float instead.
p = trial.suggest_uniform('p', 0.1, 0.5)
/tmp/ipykernel_34/1793251812.py:51: FutureWarning: suggest_loguniform has been deprecated in v3.0.0. This feature will be removed in v6.0.0. See <https://github.com/optuna/optuna/releases/tag/v3.0.0>. Use suggest_float(..., log=True) instead.

learning_rate = trial.suggest_loguniform('learning_rate', 1e-5, 1e-2)
[I 2024-03-09 20:37:33,189] Trial 0 finished with value: 0.6842309832572937 and parameters: {'p': 0.1982082705778666, 'learning_rate': 4.102608508754551e-05}. Best is trial 0 with value: 0.6842309832572937.
[I 2024-03-09 20:39:04,119] Trial 1 finished with value: 0.7005489468574524 and parameters: {'p': 0.33457499616056496, 'learning_rate': 9.583443909648773e-05}. Best is trial 1 with value: 0.7005489468574524.
[I 2024-03-09 20:40:36,316] Trial 2 finished with value: 0.7230759859085083 and parameters: {'p': 0.1281157538412595, 'learning_rate': 0.0003808361298915004}. Best is trial 2 with value: 0.7230759859085083.
[I 2024-03-09 20:42:07,756] Trial 3 finished with value: 0.6540194153785706 and parameters: {'p': 0.12884084067762905, 'learning_rate': 1.0919962051894905e-05}.

Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:43:38,671] Trial 4 finished with value: 0.7027444243431091 and parameters: {'p': 0.24511586913932654, 'learning_rate': 0.00011518744469786833}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:45:09,602] Trial 5 finished with value: 0.6477644443511963 and parameters: {'p': 0.48140940146265765, 'learning_rate': 1.4111971335697399e-05}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:46:40,200] Trial 6 finished with value: 0.6757804155349731 and parameters: {'p': 0.2545407321199509, 'learning_rate': 5.504880098387557e-05}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:48:10,776] Trial 7 finished with value: 0.6997599005699158 and parameters: {'p': 0.2716761292827566, 'learning_rate': 0.00012698614801271146}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:49:41,356] Trial 8 finished with value: 0.5 and parameters: {'p': 0.20268604130687265, 'learning_rate': 0.0026307140968659204}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:51:10,859] Trial 9 finished with value: 0.690485954284668 and parameters: {'p': 0.11866885150586644, 'learning_rate': 4.4060368454011586e-05}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:52:40,439] Trial 10 finished with value: 0.5 and parameters: {'p': 0.3527426853833819, 'learning_rate': 0.0013319982849342856}. Best is trial 2 with value: 0.7230759859085083.

[I 2024-03-09 20:54:10,701] Trial 11 finished with value: 0.7249971032142639 and parameters: {'p': 0.2019302366779353, 'learning_rate': 0.000529227651575555}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 20:55:39,970] Trial 12 finished with value: 0.7242996096611023 and parameters: {'p': 0.16406771470163942, 'learning_rate': 0.0005610330986994265}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 20:57:09,570] Trial 13 finished with value: 0.5 and parameters: {'p': 0.18770201715009055, 'learning_rate': 0.00998524698776423}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 20:58:39,736] Trial 14 finished with value: 0.720571756362915 and parameters: {'p': 0.383477402804532, 'learning_rate': 0.0006018687638830771}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:00:09,254] Trial 15 finished with value: 0.7191766500473022 and parameters: {'p': 0.17115511612622608, 'learning_rate': 0.0010799746052849442}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:01:39,036] Trial 16 finished with value: 0.5 and parameters: {'p': 0.2268153582212193, 'learning_rate': 0.003466691955888177}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:03:09,593] Trial 17 finished with value: 0.7113550305366516 and parameters: {'p': 0.295656002706911, 'learning_rate': 0.00021535206561826427}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:04:39,866] Trial 18 finished with value: 0.722595751285553 and parameters: {'p': 0.1573404839933911, 'learning_rate': 0.0004957752810611625}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:06:08,289] Trial 19 finished with value: 0.7203201651573181 and parameters: {'p': 0.10434756244321997, 'learning_rate': 0.0009942386720890037}.

Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:07:37,473] Trial 20 finished with value: 0.7219897508621216 and parameters: {'p': 0.46590344644113624, 'learning_rate': 0.002676332541243299}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:09:06,473] Trial 21 finished with value: 0.7130817770957947 and parameters: {'p': 0.1428318844375741, 'learning_rate': 0.000308976843604219}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:10:34,424] Trial 22 finished with value: 0.7217152118682861 and parameters: {'p': 0.2150402265647451, 'learning_rate': 0.0005085306898652704}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:12:05,218] Trial 23 finished with value: 0.7164779901504517 and parameters: {'p': 0.16188753426290303, 'learning_rate': 0.00024262227447588504}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:13:35,186] Trial 24 finished with value: 0.7237392663955688 and parameters: {'p': 0.13966920619110937, 'learning_rate': 0.0006247126108681989}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:15:06,271] Trial 25 finished with value: 0.5 and parameters: {'p': 0.1761171449698685, 'learning_rate': 0.00143353696543582}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:16:35,624] Trial 26 finished with value: 0.7186164259910583 and parameters: {'p': 0.2898797554093683, 'learning_rate': 0.000723542714126483}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:18:07,184] Trial 27 finished with value: 0.71931391954422 and parameters: {'p': 0.22716380976033324, 'learning_rate': 0.000198221848451625}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:19:37,779] Trial 28 finished with value: 0.5 and parameters: {'p': 0.423737935398716, 'learning_rate': 0.00577203316718783}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:21:08,253] Trial 29 finished with value: 0.5 and parameters: {'p': 0.10007765131908031, 'learning_rate': 0.001855985032806444}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:22:38,813] Trial 30 finished with value: 0.7221726775169373 and parameters: {'p': 0.19328131152550662, 'learning_rate': 0.0008585101067286634}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:24:08,767] Trial 31 finished with value: 0.7217152118682861 and parameters: {'p': 0.1414807183178992, 'learning_rate': 0.00047645514200838116}. Best is trial 11 with value: 0.7249971032142639.

[I 2024-03-09 21:25:39,695] Trial 32 finished with value: 0.7260262966156006 and parameters: {'p': 0.13413313347984926, 'learning_rate': 0.00038013014923441644}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:27:11,402] Trial 33 finished with value: 0.7144539952278137 and parameters: {'p': 0.14287302315660716, 'learning_rate': 0.00030678872883631764}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:28:42,290] Trial 34 finished with value: 0.711057722568512 and parameters: {'p': 0.1688459819622187, 'learning_rate': 0.00015317208520328985}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:30:14,427] Trial 35 finished with value: 0.6970269083976746 and parameters: {'p': 0.12370555975997462, 'learning_rate': 7.401575174115971e-05}.

Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:31:47,077] Trial 36 finished with value: 0.6738250851631165 and parameters: {'p': 0.3263034774823485, 'learning_rate': 2.193605467138049e-05}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:33:18,638] Trial 37 finished with value: 0.7173584699630737 and parameters: {'p': 0.20536196320065417, 'learning_rate': 0.0003875752018989258}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:34:50,540] Trial 38 finished with value: 0.6859691739082336 and parameters: {'p': 0.25040179010904673, 'learning_rate': 8.685489635863698e-05}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:36:22,132] Trial 39 finished with value: 0.7233847379684448 and parameters: {'p': 0.14865517624238866, 'learning_rate': 0.0007339808119778373}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:37:52,157] Trial 40 finished with value: 0.5 and parameters: {'p': 0.12098013247344702, 'learning_rate': 0.0019128913277545012}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:39:23,715] Trial 41 finished with value: 0.7214637398719788 and parameters: {'p': 0.14787278799432046, 'learning_rate': 0.000595779290331924}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:40:55,559] Trial 42 finished with value: 0.5 and parameters: {'p': 0.18269904558704098, 'learning_rate': 0.0007551072874378217}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:42:26,753] Trial 43 finished with value: 0.7183419466018677 and parameters: {'p': 0.12724965261741766, 'learning_rate': 0.00036187654703443567}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:43:59,231] Trial 44 finished with value: 0.7244368195533752 and parameters: {'p': 0.23079055158619005, 'learning_rate': 0.001226056890085937}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:45:31,697] Trial 45 finished with value: 0.5 and parameters: {'p': 0.23248656094229608, 'learning_rate': 0.001200034498855268}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:47:04,495] Trial 46 finished with value: 0.5 and parameters: {'p': 0.1994632125442621, 'learning_rate': 0.00180296930642743}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:48:36,171] Trial 47 finished with value: 0.7056260108947754 and parameters: {'p': 0.2735318198859898, 'learning_rate': 0.00014638276889708383}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:50:08,569] Trial 48 finished with value: 0.7170383334159851 and parameters: {'p': 0.2632135570462474, 'learning_rate': 0.0002527172260741916}. Best is trial 32 with value: 0.7260262966156006.

[I 2024-03-09 21:51:41,287] Trial 49 finished with value: 0.726346492767334 and parameters: {'p': 0.21422317642721472, 'learning_rate': 0.00039829575172081093}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 21:53:13,074] Trial 50 finished with value: 0.716935396194458 and parameters: {'p': 0.24295882322715306, 'learning_rate': 0.00040654239859585286}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 21:54:45,632] Trial 51 finished with value: 0.7224699258804321 and parameters: {'p': 0.21543367511445152, 'learning_rate': 0.000626998623151666}.

Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 21:56:17,974] Trial 52 finished with value: 0.7232475876808167 and parameters: {'p': 0.18230076227610106, 'learning_rate': 0.0009822331201833475}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 21:57:49,981] Trial 53 finished with value: 0.5 and parameters: {'p': 0.21042222452688353, 'learning_rate': 0.0013966668744401706}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 21:59:22,589] Trial 54 finished with value: 0.7079360485076904 and parameters: {'p': 0.16203800746897845, 'learning_rate': 0.00016842555700275513}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:00:54,551] Trial 55 finished with value: 0.71733558177948 and parameters: {'p': 0.31906131577299496, 'learning_rate': 0.00028964486323548085}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:02:27,063] Trial 56 finished with value: 0.7247112393379211 and parameters: {'p': 0.11084828307322794, 'learning_rate': 0.0004944585765377107}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:03:59,261] Trial 57 finished with value: 0.7230188250541687 and parameters: {'p': 0.1915828157202879, 'learning_rate': 0.0004669617672950447}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:05:31,334] Trial 58 finished with value: 0.5 and parameters: {'p': 0.11597632514690247, 'learning_rate': 0.0031398699902611283}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:07:03,223] Trial 59 finished with value: 0.7177358269691467 and parameters: {'p': 0.10981440266749518, 'learning_rate': 0.0002191289362763716}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:08:34,687] Trial 60 finished with value: 0.7057976126670837 and parameters: {'p': 0.23777926505284647, 'learning_rate': 0.00011354381287830016}. Best is trial 49 with value: 0.726346492767334.

[I 2024-03-09 22:10:05,282] Trial 61 finished with value: 0.7274671196937561 and parameters: {'p': 0.13413341106717633, 'learning_rate': 0.0005827059588796402}. Best is trial 61 with value: 0.7274671196937561.

[I 2024-03-09 22:11:36,884] Trial 62 finished with value: 0.5 and parameters: {'p': 0.1328700638809459, 'learning_rate': 0.0008811797587264916}. Best is trial 61 with value: 0.7274671196937561.

[I 2024-03-09 22:13:07,871] Trial 63 finished with value: 0.7210177779197693 and parameters: {'p': 0.155348652846431, 'learning_rate': 0.0003530465294215336}. Best is trial 61 with value: 0.7274671196937561.

[I 2024-03-09 22:14:39,160] Trial 64 finished with value: 0.7185477018356323 and parameters: {'p': 0.1765458019121911, 'learning_rate': 0.0005410348914320352}. Best is trial 61 with value: 0.7274671196937561.

[]: