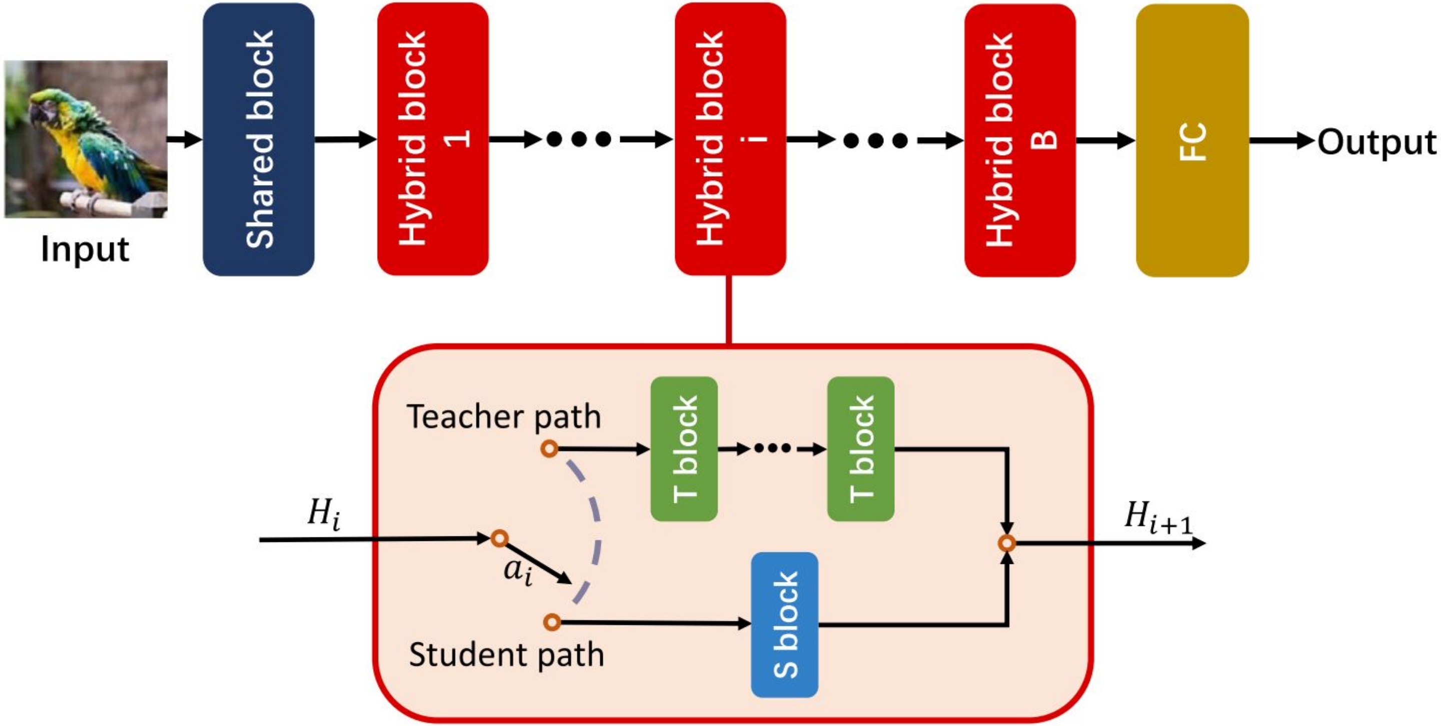
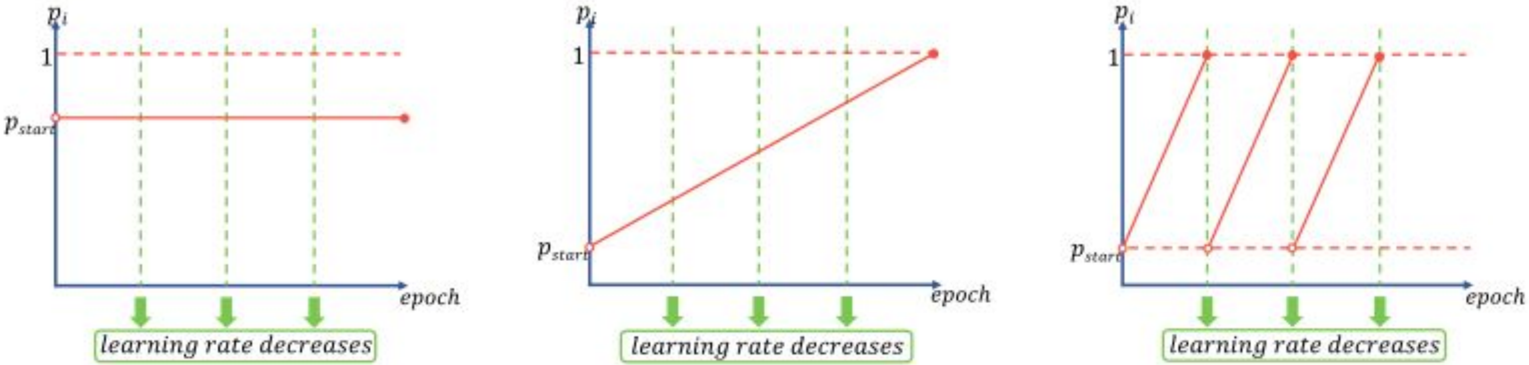
Interactive Knowledge Distillation for image classification

Maciej Chylak, Dawid Janus, Arkadiusz Kniaź

Architektura



Eksperymenty

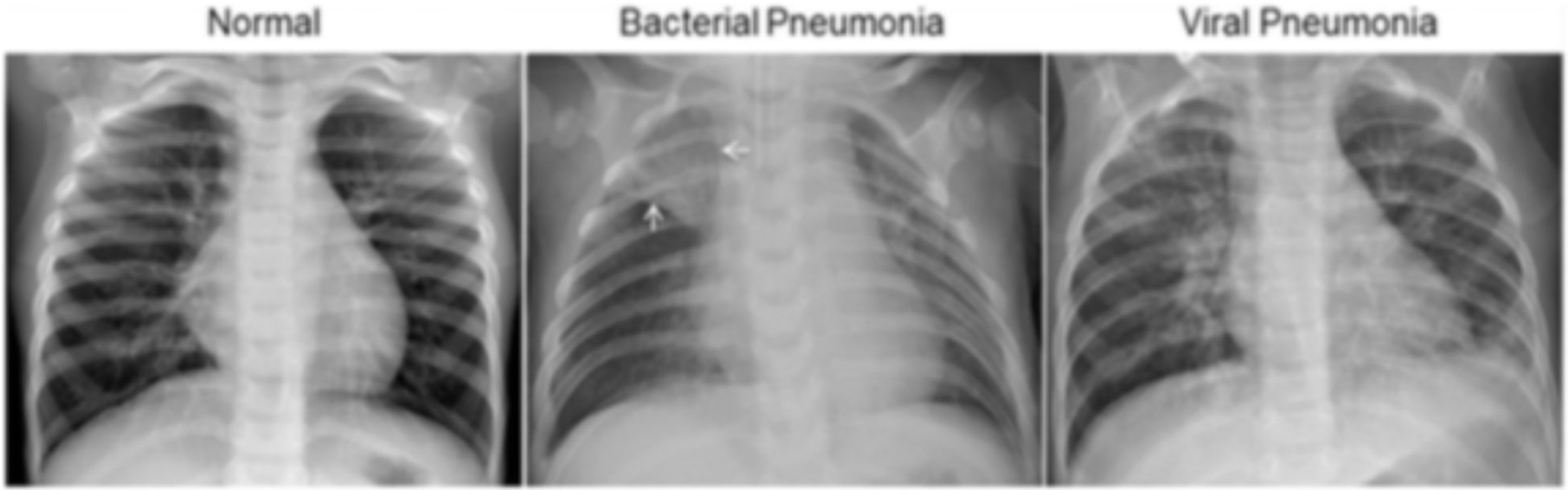


(a) Uniform schedule

(b) Linear growth schedule

(c) Review schedule

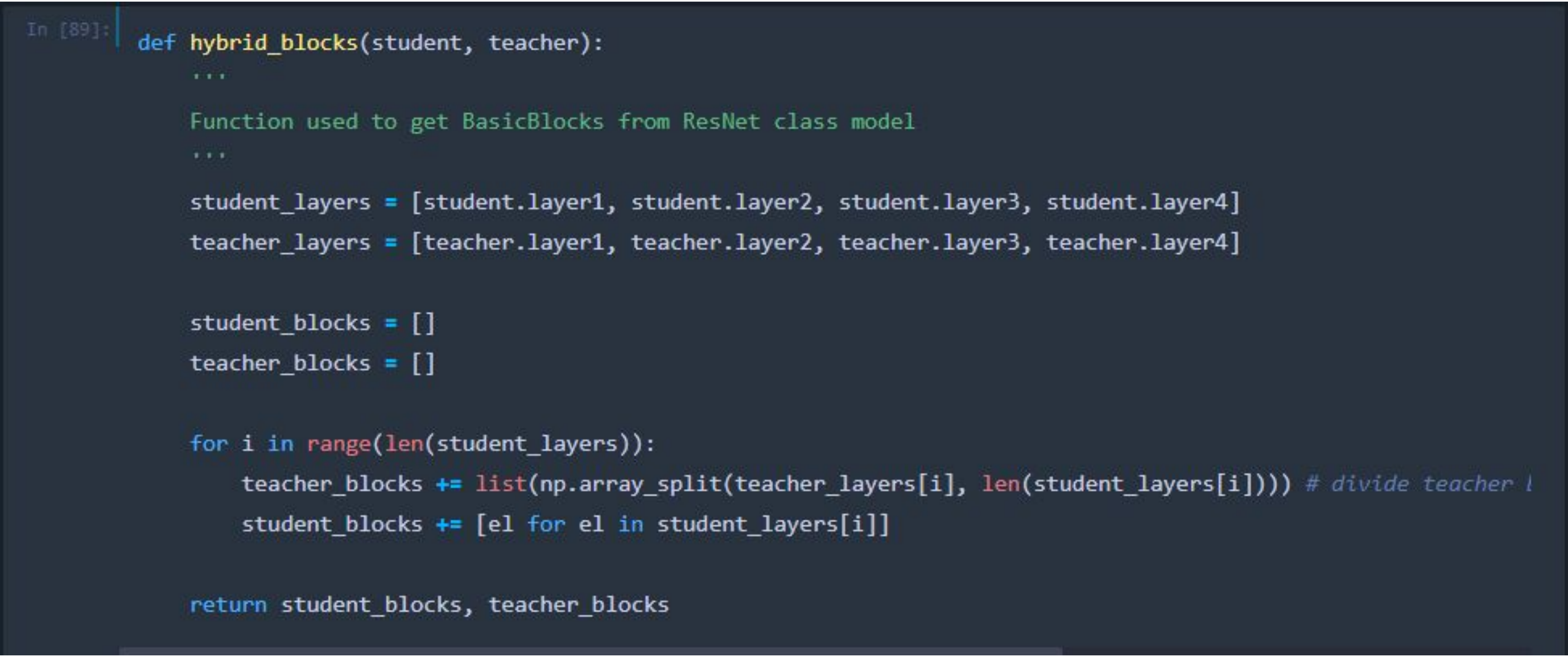
Dane: Chest X-ray Images (Pneumonia)



Normal

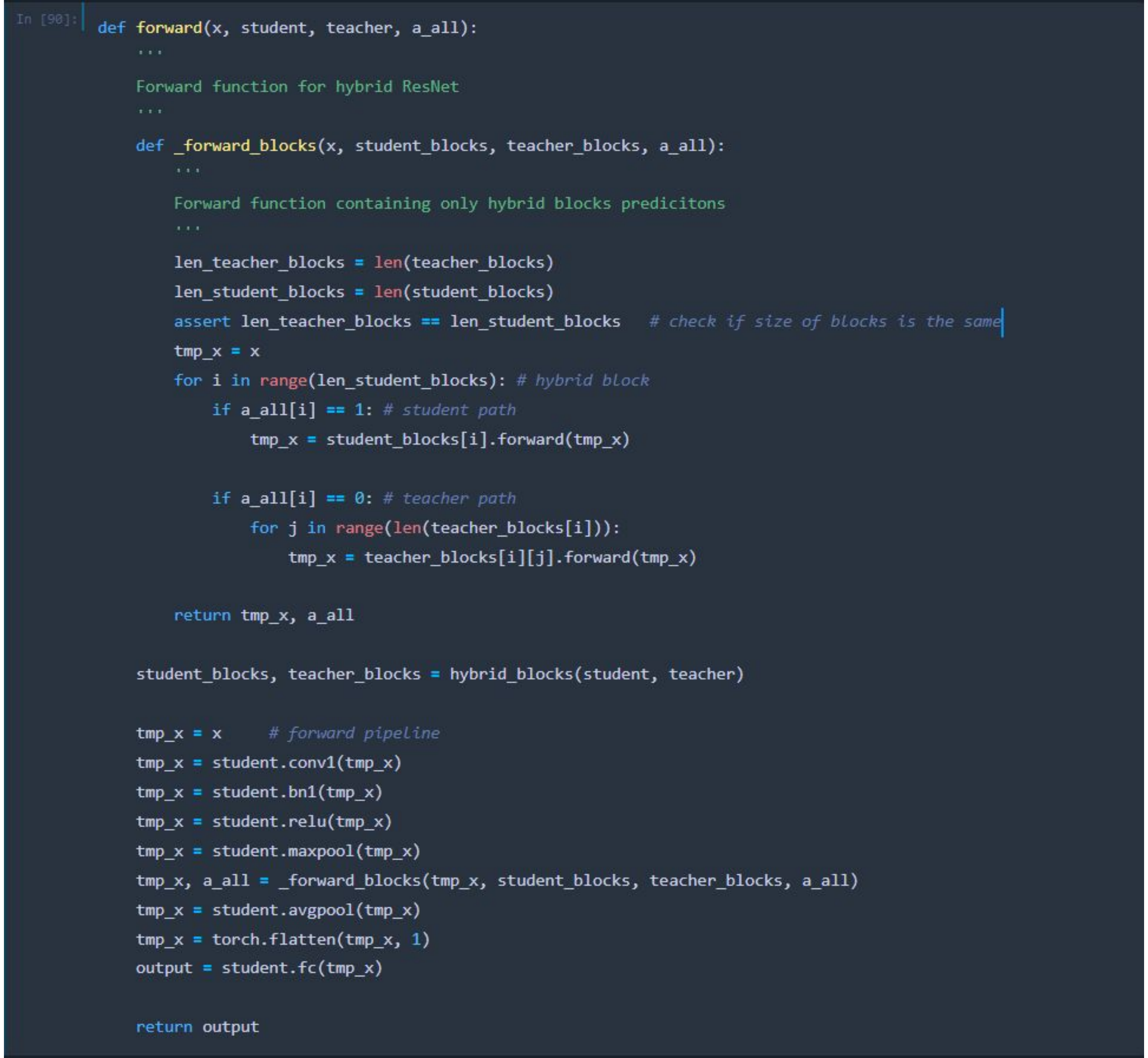
Bacterial Pneumonia

Viral Pneumonia

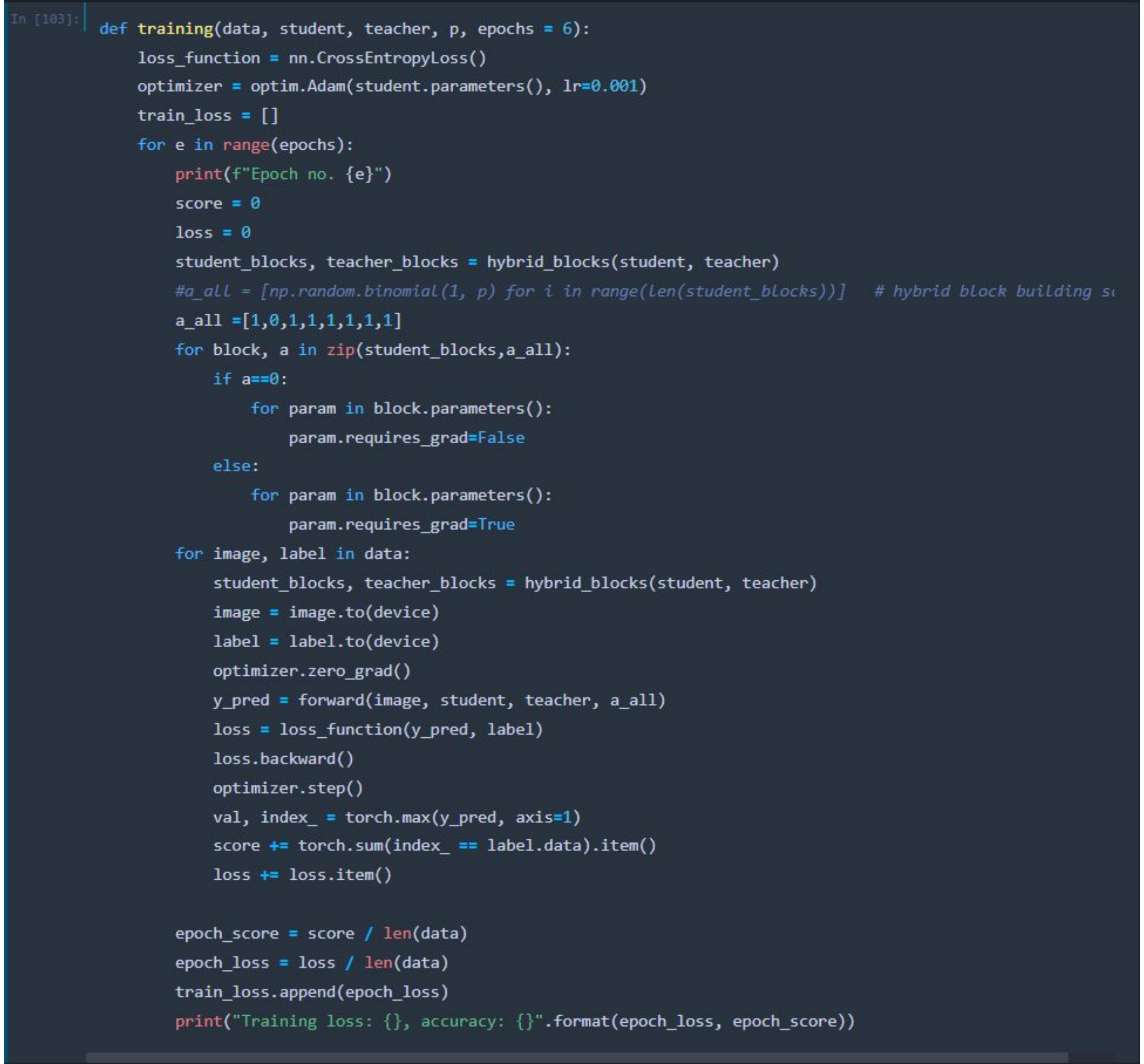


In [89]:

```python
def hybrid_blocks(student, teacher):
    '''
    Function used to get BasicBlocks from ResNet class model
    '''
    student_layers = [student.layer1, student.layer2, student.layer3, student.layer4]
    teacher_layers = [teacher.layer1, teacher.layer2, teacher.layer3, teacher.layer4]

    student_blocks = []
    teacher_blocks = []

    for i in range(len(student_layers)):
        teacher_blocks += list(np.array_split(teacher_layers[i], len(student_layers[i]))) # divide teacher l
        student_blocks += [el for el in student_layers[i]]

    return student_blocks, teacher_blocks
```



```python
def forward(x, student, teacher, a_all):
    '''
    Forward function for hybrid ResNet
    '''
    def _forward_blocks(x, student_blocks, teacher_blocks, a_all):
        '''
        Forward function containing only hybrid blocks predicitons
        '''
        len_teacher_blocks = len(teacher_blocks)
        len_student_blocks = len(student_blocks)
        assert len_teacher_blocks == len_student_blocks   # check if size of blocks is the same
        tmp_x = x
        for i in range(len_student_blocks): # hybrid block
            if a_all[i] == 1: # student path
                tmp_x = student_blocks[i].forward(tmp_x)

            if a_all[i] == 0: # teacher path
                for j in range(len(teacher_blocks[i])):
                    tmp_x = teacher_blocks[i][j].forward(tmp_x)

        return tmp_x, a_all

    student_blocks, teacher_blocks = hybrid_blocks(student, teacher)

    tmp_x = x     # forward pipeline
    tmp_x = student.conv1(tmp_x)
    tmp_x = student.bn1(tmp_x)
    tmp_x = student.relu(tmp_x)
    tmp_x = student.maxpool(tmp_x)
    tmp_x, a_all = _forward_blocks(tmp_x, student_blocks, teacher_blocks, a_all)
    tmp_x = student.avgpool(tmp_x)
    tmp_x = torch.flatten(tmp_x, 1)
    output = student.fc(tmp_x)

    return output
```



In [103]:

```
def training(data, student, teacher, p, epochs = 6):
    loss_function = nn.CrossEntropyLoss()
    optimizer = optim.Adam(student.parameters(), lr=0.001)
    train_loss = []
    for e in range(epochs):
        print(f"Epoch no. {e}")
        score = 0
        loss = 0
        student_blocks, teacher_blocks = hybrid_blocks(student, teacher)
        #a_all = [np.random.binomial(1, p) for i in range(len(student_blocks))]   # hybrid block building s
        a_all =[1,0,1,1,1,1,1,1]
        for block, a in zip(student_blocks,a_all):
            if a==0:
                for param in block.parameters():
                    param.requires_grad=False
            else:
                for param in block.parameters():
                    param.requires_grad=True
        for image, label in data:
            student_blocks, teacher_blocks = hybrid_blocks(student, teacher)
            image = image.to(device)
            label = label.to(device)
            optimizer.zero_grad()
            y_pred = forward(image, student, teacher, a_all)
            loss = loss_function(y_pred, label)
            loss.backward()
            optimizer.step()
            val, index_ = torch.max(y_pred, axis=1)
            score += torch.sum(index_ == label.data).item()
            loss += loss.item()

        epoch_score = score / len(data)
        epoch_loss = loss / len(data)
        train_loss.append(epoch_loss)
        print("Training loss: {}, accuracy: {}".format(epoch_loss, epoch_score))
```

Zrealizowane zadania

- wczytanie danych oraz modeli studenta i nauczyciela
- zaimplementowanie interaktywnego uczenia modelu studenta
- obsługa stałego wpływu nauczyciela w procesie uczenia

Planowane prace

- stworzenie wizualizacji treningu
- zaimplementowanie różnych wersji wyboru prawdopodobieństwa przejścia przez blok nauczyciela
- napisanie artykułu