

דו"ח תרגיל 5

ארכיטקטורה:

יצרנו 4 פעמים את הצירוף (שכבת קונבולוציה, batchNorm וMaxPooling). הפרמטרים הוגדרו באופן הבא:

- שכבת קונבולוציה
 - פרמטרים בצירוף הראשון:
 - 1 – Input channels
 - כמות קרנלים – 32
 - גודל קרנל – 5
 - 1 – Stride
 - פרמטרים בצירוף השני:
 - 32 – Input channels
 - כמות קרנלים – 32
 - גודל קרנל – 3
 - 1 – Stride
 - פרמטרים בצירוף השלישי:
 - 32 – Input channels
 - כמות קרנלים – 64
 - גודל קרנל – 3
 - 1 – Stride
 - פרמטרים בצירוף הרביעי:
 - 64 – Input channels
 - כמות קרנלים – 32
 - גודל קרנל – 3
 - 1 – Stride
- שכבת batchNorm – לכל אחד מהצירופים, הגדרנו את גודל השכבה ככמות הקרנלים שהוגדרו בשכבת הקונבולוציה שמאותו צירוף.
- שכבת maxPooling – בשלושת הצירופים הראשונים הקטנו פי 2 את גודל הקלט, ובצירוף הרביעי הקטנו פי 4.

בסוף הגדרנו שכבת fully-connected שגודל הקלט הוא 64 (פי 2 מכמות הקרנלים שיצאו מהצירוף האחרון), והפלט הוא כמות הקלאסים שהרשת אמורה לסווג.

```
self.conv1 = nn.Conv2d(n_input, n_channel, kernel_size=5, stride=stride)
self.bn1 = nn.BatchNorm2d(n_channel)
self.pool1 = nn.MaxPool2d(2)
self.conv2 = nn.Conv2d(n_channel, n_channel, kernel_size=3)
self.bn2 = nn.BatchNorm2d(n_channel)
self.pool2 = nn.MaxPool2d(2)
self.conv3 = nn.Conv2d(n_channel, 2 * n_channel, kernel_size=3)
self.bn3 = nn.BatchNorm2d(2 * n_channel)
self.pool3 = nn.MaxPool2d(2)
self.conv4 = nn.Conv2d(2 * n_channel, n_channel, kernel_size=3)
self.bn4 = nn.BatchNorm2d(n_channel)
self.pool4 = nn.MaxPool2d(4)
self.fc1 = nn.Linear(2 * n_channel, n_output)
```

על כל צירוף הפעלנו relu בין שכבת batchNorm לבין שכבת MaxPooling לפני שכבת fully-connected, הפעלנו avgPooling על התוצאה של הצירוף האחרון. בנוסף, שיטחנו את המטריצה שקיבלנו כדי שהיא תוכל להיכנס ל-fully-connected. הפעלנו softmax על השכבה של fully-connected. פונקציית הloss שהשתמשנו בה היא negative log likelihood.

היפר פרמטרים:

- Learning rate – 0.01
- Optimizer – Adam
- Learning Rate Scheduler (with step size=10, and gamma=0.1)
כלומר כל 10 אפוקים, הוזל יורד פי 10.
- Batch size – 100
- Num epochs – 50 (but we can see in the graphs that 15 epochs were enough)

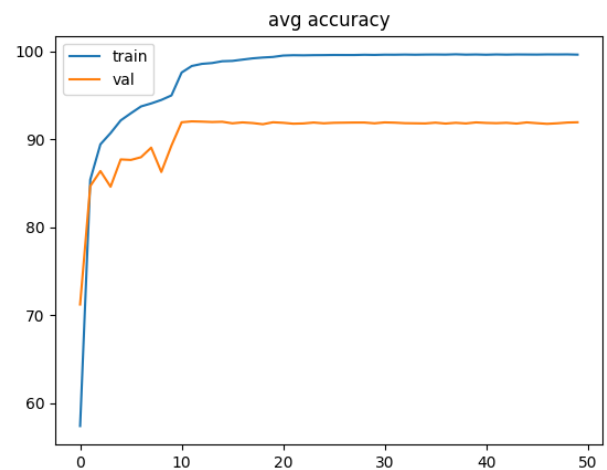
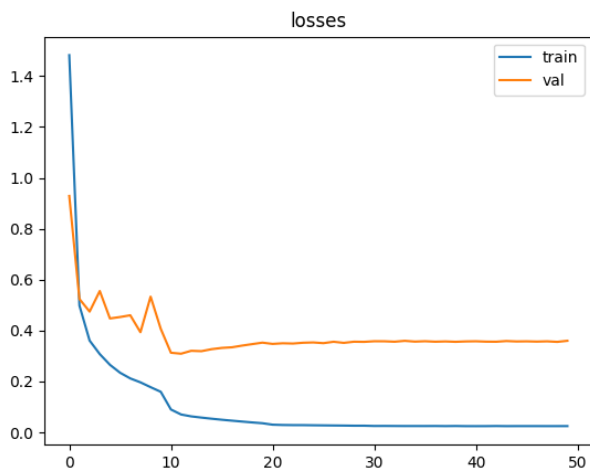
תהליך בחירת הארכיטקטורה וההיפר פרמטרים:

ראינו באינטרנט דוגמה לארכיטקטורה שמטרתה לפתור זיהוי פקודה. לקחנו השראה מהארכיטקטורה הזו, שגם הייתה בנויה מצירופים של Convolution, batchNorm and MaxPooling עדכנו את הארכיטקטורה הזו בהתאם לצרכים שלנו - שינינו את גדלי הpooling, שיטחנו את התוצאה שהתקבלה מרשת הקונבולוציה (הרשת שמצאנו עבדה על קלט חד מימד ואילו אנחנו היינו צריכים לעבוד על דו מימד). בנוסף בדקנו ערכי learning rate, מספר step_size וכן מספר epochs שונים ובחרנו את הערכים שעבורם קיבלנו את התוצאות הטובות ביותר.

תוצאות:

הגענו לvalidation accuracy של 92.04%

גרפים של ה-loss וה-accuracy:



אופן הרצת הקוד:

התוכנית מקבלת כארגומנט את ה data path בה נמצאות התיקיות train, valid, test. נדגיש שתיקיית ה test מכילה את קבצי השמע עצמם, ולא תיקייה נוספת שמכילה את קבצי השמע.

שורת ההרצה תיראה כך:

```
python ex5.py <data_path>
```