

דו"ח למידת מכונה – תרגיל 5

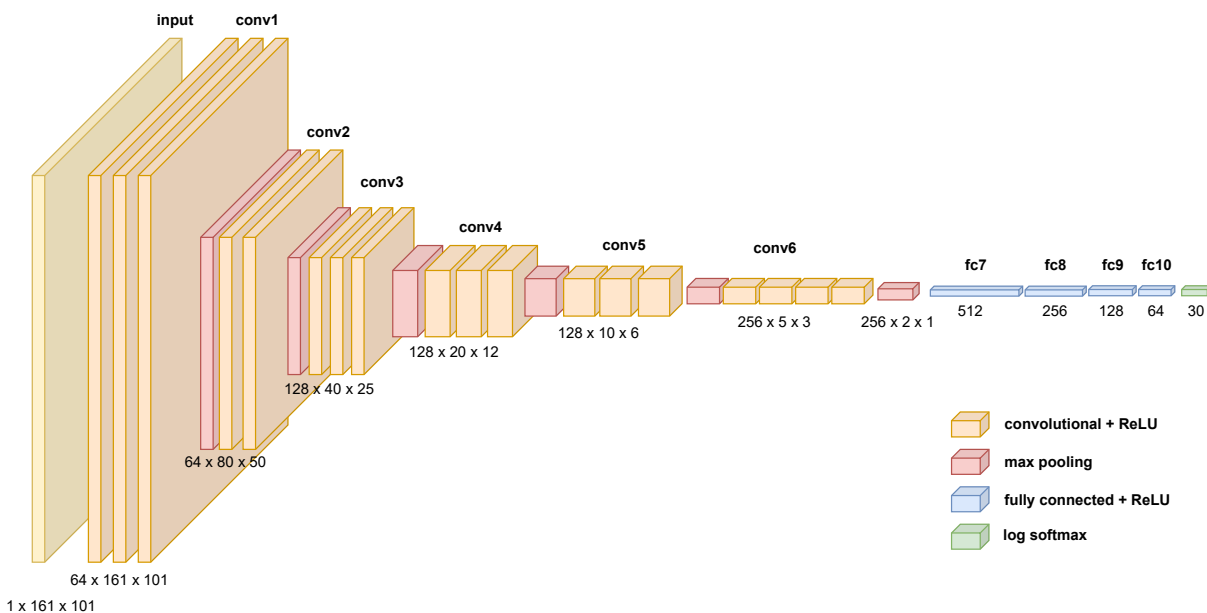
תומר שי 323082701, רועי גידה 32225897

8 בינואר 2022

1 Our BEST MODEL

המודל הטוב ביותר שלנו, אשר הגיע לכ-94.7 אחוזי דיוק על ה-validation set וכ-98 אחוזי דיוק על ה-training set. ניתן לראות את אחוזי הדיוק וערכי ה-loss של המודל באיור 1.2.

1.1 Architecture



איור 1.1: Best Model Architecture

שכבה 1. שכבת קונבולוציה עם $\text{kernel size} = 7$ ו- $\text{zero padding} = 3$. פלט השכבה הוא בגודל 161×101 עם $\text{depth} = 64$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 2. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 80×50 עם $\text{depth} = 64$ (ללא שינוי).

שכבה 3. שכבת קונבולוציה עם $\text{kernel size} = 3$ ו- $\text{zero padding} = 1$. פלט השכבה הוא בגודל 80×50 עם $\text{depth} = 64$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 4. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 40×25 עם $\text{depth} = 64$ (ללא שינוי).

שכבה 5. שכבת קונבולוציה עם $\text{kernel size} = 3$ ו- $\text{zero padding} = 1$. פלט השכבה הוא בגודל 40×25 עם $\text{depth} = 128$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 6. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 20×12 עם $\text{depth} = 128$ (ללא שינוי).

שכבה 7. שכבת קונבולוציה עם $\text{kernel size} = 3$ ו- $\text{zero padding} = 1$. פלט השכבה הוא בגודל 20×12 עם $\text{depth} = 128$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 8. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 10×6 עם $\text{depth} = 128$ (ללא שינוי).

שכבה 9. שכבת קונבולוציה עם $\text{kernel size} = 3$ ו- $\text{zero padding} = 1$. פלט השכבה הוא בגודל 10×6 עם $\text{depth} = 128$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 10. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 5×3 עם $\text{depth} = 128$ (ללא שינוי).

שכבה 11. שכבת קונבולוציה עם $\text{kernel size} = 3$ ו- $\text{zero padding} = 1$. פלט השכבה הוא בגודל 5×3 עם $\text{depth} = 256$. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 12. שכבת Max Pool עם $\text{kernel size} = 2$. פלט השכבה הוא בגודל 2×1 עם $\text{depth} = 256$ (ללא שינוי).

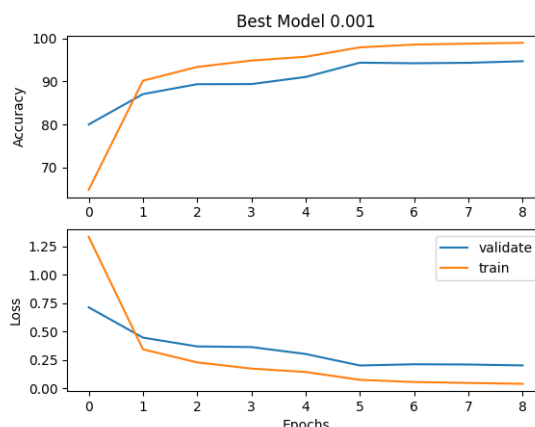
שכבה 13. שכבת Fully Connected המקבלת קלט בגודל 512 ומוציא פלט בגודל 256. לאחר השכבה הזו ביצענו Batch Normalization, הפעלנו את פונקציית ה-ReLU וביצענו $\text{Dropout} = 0.1$.

שכבה 14. שכבת Fully Connected המקבלת קלט בגודל 256 ומוציא פלט בגודל 128. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 15. שכבת Fully Connected המקבלת קלט בגודל 128 ומוציא פלט בגודל 64. לאחר השכבה הזו ביצענו Batch Normalization והפעלנו את פונקציית ה-ReLU.

שכבה 16. שכבת Fully Connected המקבלת קלט בגודל 64 ומוציא פלט בגודל 30. לאחר השכבה הזו הפעלנו את פונקציית ה-Log Softmax. זוהי שכבת הפלט.

Accuracy And Loss 1.2



איור 1.2: Best Model Accuracy and Loss Per Epoch

Hyper Parameters 1.3

כפי שצויין קודם, השתמשנו ב-Dropout = 0.1 לאחר שכבת ה-Fully Connected הראשונה. הרצנו את המודל במשך כ-12 epochs, עם learning rate = 0.0001, batch size = 64 ו-optimizer = Adam.

How We Choose It 1.4

בתור התחלה, בין היתר כדי לבדוק שהמודל מצליח ללמוד ועובד כראוי, הגדרנו מודל בסיסי ביותר, שהכיל 2 שכבות קונבולוציה ($1 \rightarrow 8 \rightarrow 16$), עם Max Pool ביניהם, Batch Normalization אחרי כל אחד ועם פונקציית ה-ReLU. לאחר מכן 4 שכבות Fully Connected: $3840 \rightarrow 1024 \rightarrow 512 \rightarrow 30$ עם Batch Normalization לאחר כל שכבה. הרשת הצליחה ללמוד, עם אחוזי דיוק ואחוזי Loss הבאים:

Epoch	Train Accuracy	Train Loss	Validate Accuracy	Validate Loss
1	51.28%	2.03	71.93%	1.34
2	77.07%	1.14	78.74%	0.99
3	85.32%	0.80	82.39%	0.82
4	89.39%	0.62	82.02%	0.79

טבלה 1.1: First Network Accuracy And Loss Per Epoch

לאחר שראינו כי הרשת מצליחה ללמוד, ניסינו לשנות אותה קצת בכל פעם על-מנת להגיע לאחוזי דיוק מירביים.

לאחר ששינינו אותה המון, ולאחר קריאה באינטרנט על מודלים טובים יותר וטובים פחות להתמודדות עם קבצי שמע, מימשנו את מודל ה-VGG11 שהוביל לתוצאות מעולות. לאחר מכן שינינו בקצת את המודל, הוספנו והורדנו שכבות ולבסוף הגענו למודל הסופי שלנו, שלא הצלחנו לשפר את אחוזי הדיוק שלו.

Running Instruction 2

על-מנת להריץ את המודל, יש לוודא תחילה כי כל קבצי ה-data מסודרים תחת תיקייה הנקראת gcommands, ושתחתיה יהיו תיקיות עבור ה-train, validate ועבור ה-test. בכל אחת מהתיקיות האלו, צריכים להימצא מספר תתי-תיקיות עבור ה-labels השונים (כאשר ב-test אין לשם התיקיות שום משמעות), כאשר בתוכם להיות קבצי ה-wav.

כעת, יש לשים באותה תיקייה את קובץ ההרצה ex5.py ואת קובץ טעינת הקבצים gcommand_dataset.py.

יש להתקין את הסיפריות הבאות:

- PrettyTable
- Numpy
- Soundfile
- PyTorch
- Librosa
- Tqdm
- Matplotlib

ולתמוך ב-python 3.9 (ייתכן כי גרסאות מאוחרות יותר או מוקדמות יותר לא יעבדו).

הקובץ ex5.py יכול לקבל ארגומנט אחד, והוא האם לבצע שימוש ב-cuda או לא. ברירת המחדל היא הרצה ב-cpu, ואם ברצונכם להריץ את הקוד באמצעות cuda יש להוסיף "cuda" כארגומנט באופן הבא:

```
python3 ex5.py cuda
```

לאחר מכן, אם תתאפשר הרצה על ה-GPU, יודפס כי ה-Device שנבחר הוא cuda ואחרת יודפס CPU.

במהלך הריצה, יודפסו אחוזי ה-accuracy וה-loss על ה-training set וה-validation set לאחר כל epoch, וכן יוצג progress bar שבו ניתן לראות מתי ה-epoch יסתיים.

```
D:\Users\tomer\PycharmProjects\ML\ex5>python ex5.py cuda
Hyper Parameters
-----
| Device | Batch Size | Number Of Epochs | Learning Rate |
-----
| cuda:0 | 64         | 15                | 0.001         |
-----
EPOCH #1
train iterations: 33% | 157/469 [03:54<07:41, 1.48s/it]
```

איור 2.1: Running example