

# למידת מכונה – תרגיל 5

ראגב גאזי – 314892506

מוחמד יוסף – 211668975

## ארכיטקטורת המודל:

הרשת שלנו בנויה משלושה שכבות קונבולוציה (CNN) ושלושה שכבות fully connected.

ואז ביצענו dropout כדי להימנע מ over-fitting .

## פירוט שכבות CNN :

- שכבה ראשונה : מספר הערוצים הנכנסים כקלט הינו 1, הגדרנו את מספר הערוצים אשר מיוצר ע"י הקונבולוציה להיות 16 , גודל הפילטר הינו  $5 \times 5$  כאשר מספר הצעדים (strides) מוגדר להיות 2 וה padding-מוגדר להיות גם 2.
- שכבה שניה : מספר הערוצים הנכנסים כקלט הינו התוצאה מהשכבה הקודמת (כלומר 16), הגדרנו את מספר הערוצים אשר מיוצר ע"י הקונבולוציה להיות 32, גודל הפילטר הינו  $3 \times 3$ , מספר הצעדים (stride) מוגדר להיות 1, וגודל ה padding-מוגדר להיות גם 1.
- שכבה שלישית : מספר הערוצים הנכנסים כקלט הינו התוצאה מהשכבה הקודמת (כלומר 32), הגדרנו את מספר הערוצים אשר מיוצר ע"י הקונבולוציה להיות 32, גודל הפילטר הינו  $3 \times 3$ , מספר הצעדים (stride) מוגדר להיות 1, וגודל ה padding-מוגדר להיות גם 1 ותוצאת שכבה זאת מוגדרת להיות 64.

## פירוט שכבות fully connected :

- שכבה ראשונה : כתוצאה מהשכבות הקודמות , הקלט לשכבה זו הינו בגודל של 3840 , הגדרנו את הפלט להיות 1024 עבור השכבה הבאה.
- שכבה שניה : כתוצאה מהשכבה הקודמת , הקלט לשכבה זו הינו בגודל של 1024 , הגדרנו את הפלט להיות 128 עבור השכבה הבאה.
- שכבה שלישית : כתוצאה מהשכבה הקודמת , הקלט לשכבה זו הינו בגודל של 128 , הגדרנו את הפלט להיות 30 .

עבור כל אחת מהשכבות השתמשנו בפונקציית האקטיבציה LeakyReLU, וביצענו MaxPool2d עם גודל פילטר של  $2 \times 2$  ו stride של 2.

עבור סט האימון והוולידציה הגדרנו את batch\_size להיות 100 ועבור סט ה-test להיות 1.

עשינו את פונקציית train לאימון של ה-data ואז עשינו את פונקציית test שמכניסה את ה predictions למערך ואז ממיינת אותם לפי המספר ואז מחזירה. בסוף עשינו את

פונקציית write to file שמייצרת קובץ בשם test\_y וכותבת את ה predictions לתוכו.

וכמובן עשינו את פונקציית forward כרגיל כמו בתרגיל קודם או רוב התרגילים.

השתמשנו ב - Loss Entropy Cross ו- Optimizer Adam כאשר הגדרנו learning rate = 0.001.

### בחירת ההיפר פרמטרים :

- **Optimizer:** בחרנו את אופטימיזר של ADAM כיוון שראינו בתרגיל הקודם שהוא הניב לתוצאות הכי טובות וגם בתרגיל זה אחרי שבדקנו כמה אופטימיזרים קיבלנו אחוז הצלחה הכי גבוה עבור אופטימיזר ADAM.
- **Learning rate:** ראיתי שערך ברירת המחדל הוא 0.1 או 0.01, בהתחלה בחרתי את 0.1 ואז עשיתי כמה בדיקות עבור מספר ערכים לlearning rate כמו : 0.01, 0.015, 0.1, 0.001, 0.005, 0.0015 ובסוף אחרי בדיקות של accuracy ו-loss קיבלנו שהערך 0.001 נותן לaccuracy הכי גבוה לloss הכי נמוך.
- **Epochs:** הסתכלתי על ה-validation ו- training losses ועקבתי אחר הערכים שלהם. אם ה-validation loss הולך לגדול זה אומר התאמת יתר. ואז הבנתי שכדאי להגדיר את מספר ה epochs גבוה כל האפשר ולהימנע מהתאמת יתר אבל אם אני מגדיל את זה ככל האפשר אז מקבל התאמת יתר וגם אחרי 10 epochs ראיתי שאין שיפור משמעותי ובסוף הסתפקתי במספר epochs = 10 שזה בינוני.

### כיצד ניתן להריץ את המודל :

בתיקיה ראשית (אצלנו בשם Ex\_5) יש את קבצי ה-python :

ex5.py, data\_loader.py, gcommand\_dataset.py

בנוסף את תיקיה בשם data שיש בתוכה שלושה תיקיות:

1. אימון : בשם train שמכילה את קבצי ה-wav.
2. וולידציה : בשם valid שמכילה את קבצי ה-wav.
3. מבחן : בשם test שיש בתוכה תיקיה בשם TEST שמכילה את קבצי ה-wav.

