

דו"ח הגשה – תרגיל 5, למידת מכונה

במסמך זה נפרט אודות החלקים הבאים:

- א. מבנה המודל שבחרנו.
- ב. ההיפר-פרמטרים השונים בקוד שלנו.
- ג. אופן הרצת הקוד.

מבנה המודל שבחרנו:

בחרנו לממש רשת CNN המורכבת מ-9 שכבות לא-ליניאריות ומ-2 שכבות ליניאריות. כמות הצ'אנלים המקסימלית שהגענו אליה היא 128.

כמו כן, לסירוגין על שכבה אחת מכל 2-3 שכבות לא-ליניאריות במודל ביצענו maxpool. בהתחלה, ניסינו לבצע maxpool על כל אחת מהשכבות הלא ליניאריות במודל, אך גילינו שאם אנחנו מפעילים maxpool לסירוגין, אנו מקבלות אחוזי דיוק גבוהים יותר על סט הולידציה. בנוסף, לא השתמשנו ב-dropout, שכן לאחר מספר בדיקות גילינו שהוא לא גורם לעלייה באחוזי הדיוק על סט הולידציה.

בשכבות הלא ליניאריות השתמשנו בפונקציית האקטיבציה Relu ובאופטימיזר Adam, שכן ראינו בבדיקותינו שאלו הביאו לתוצאות טובות ביותר. בנוסף, יש לציין שבשכבות השונות במודל ביצענו batch normalization מסיבה דומה. בסוף מתודת ה-forward ביצענו softmax לפי הלוגיקה שראינו בהרצאה.

ההיפר-פרמטרים השונים:

1. **Learning rate**: בדקנו את הביצועים שהתקבלו (כלומר, את אחוזי הדיוק המתקבלים על הולידציה) על מגוון רחב של ערכי lr – 0.1, 0.01, 0.0001, 0.00015, 0.00009, 0.00001 ועוד. בסופו של דבר החלטנו להגיש את המודל עם $\text{learning_rate} = 0.001$ כי ראינו שהוא הביא לתוצאה הטובה ביותר באחוזי הדיוק.
2. **Batch**: החלטנו, בדומה לתרגיל 4, להישאר עם $\text{batch} = 64$.
3. **Epochs**: החלטנו להגיש את הקוד עם 10 איפוקים, שכן ראינו שמעל 10 איפוקים המודל מתחיל לרדת באחוזי הדיוק (אולי כתוצאה מ-overfitting) על סט הולידציה.

אופן הרצת הקוד:

התכנית לא מקבלת פרמטרים. הקוד שלנו עושה שימוש בקובץ הזיפ המחולץ המכיל את סט האימון, הולידציה והטסט, ובקובץ ה-gcommand_dataset.py.

כלומר, כדי להריץ, צריך ליצור תיקייה שמכילה את הקבצים הבאים:

ex_5.py , gcommand_dataset.py

ובמו כן **3 תיקיות:**

תיקיית train (ללא שינוי, כמו שהופיעה בזיפ שסופק לתרגיל)

תיקיית validation (ללא שינוי, כמו שהופיעה בזיפ שסופק לתרגיל)

תיקיית test **עם שינוי: כלומר, תיקיית ה-test תכיל בתוכה את תיקיית ה-test שסופקה בזיפ של התרגיל.**

