

למידה עמוקה ליישומי ראייה ממוחשבת

עבודה 1 – Chest xray

מס' קורס – 51283 (יום ו')

שמות המגישים:

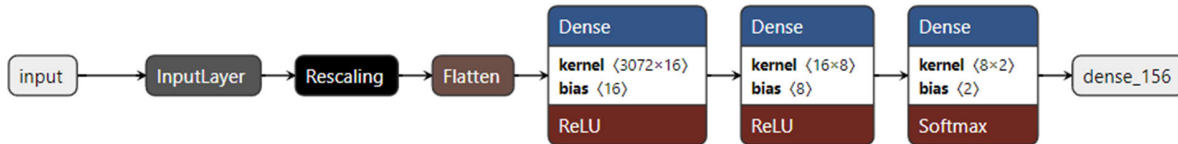
יקיר שלזינגר ת.ז 206345068

רמי דיסלן ת.ז 323837039

סמסטר א תשפ"ב

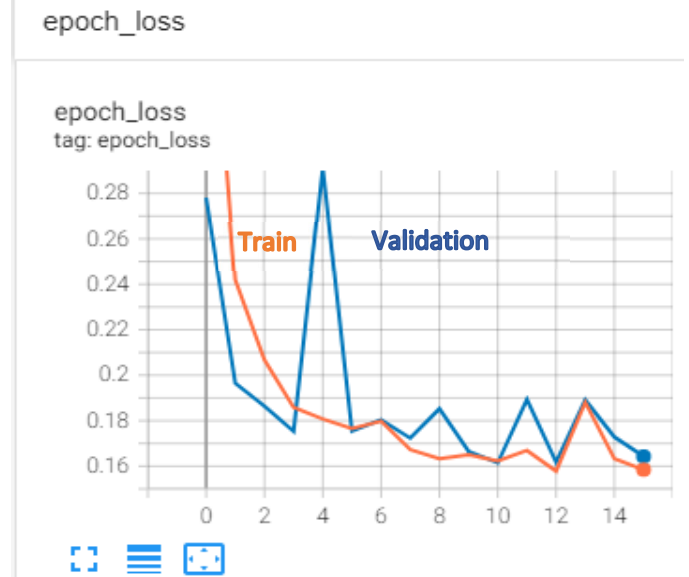
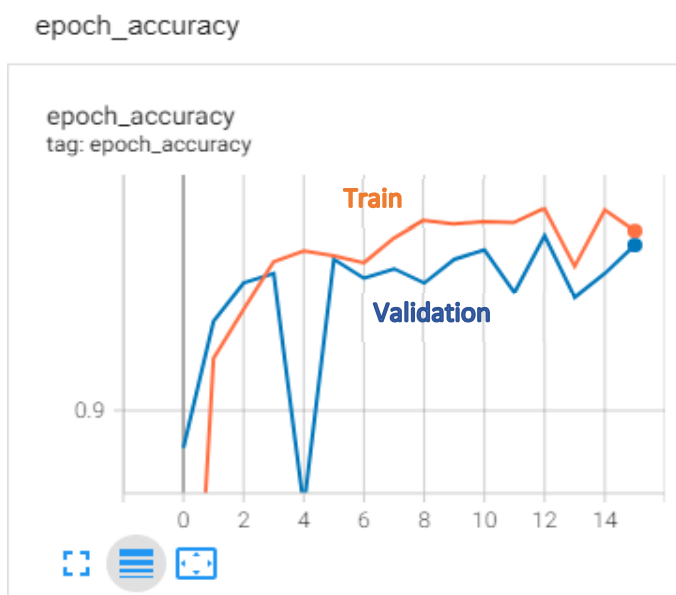
הסבר על הרשת

בפרויקט זה יצרנו רשת ללמידה עמוקה שמזהה פציינטים עם דלקת ריאות בעזרת צילומי הרנטגן שלהם. בכדי ליצור רשת זו השתמשנו במאגר בעל 5,863 תמונות. את המאגר שלנו חילקנו 70% אימון, 20% ולידציה ו 10% בחינה. תחילה יצרנו רשת מסוג DNN Fully Connected. להלן סכמת מלבנים המתארת את מודל הרשת:

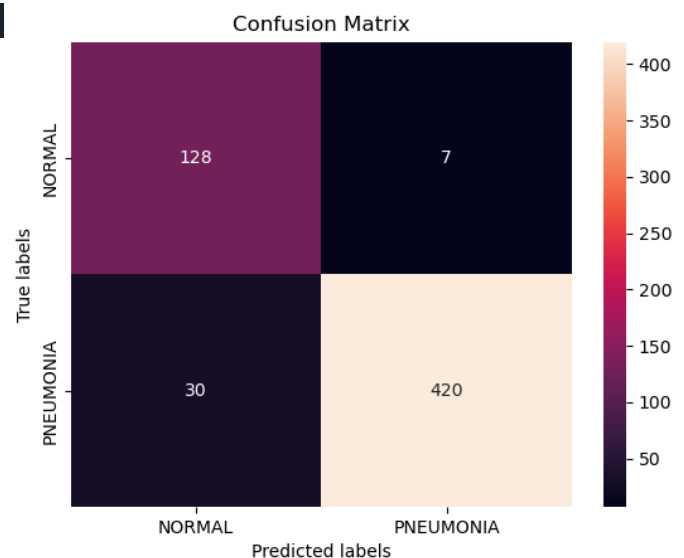


כפי שניתן לראות השתמשנו ברשת ששתי שכבות מוסתרות (הראשונה עם 16 ניוונים והשנייה עם 8 ניוונים), שכבת נרמול, שכבת Flatten ושכבת מוצא.

תוצאות

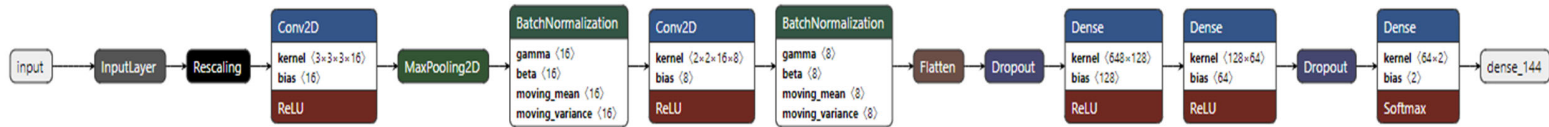


```
test loss, test acc: [0.1746823489665985, 0.9367521405220032]  
Precision: 0.9481481481481482 Recall: 0.810126582278481
```



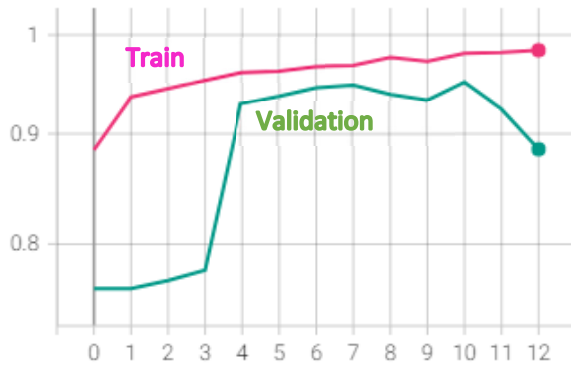
הסבר על הרשת

כעת אנו מוסיפים לרשת שכבות מסוג CNN בשביל לשפר את ביצועי הרשת. הוספנו שתי שכבות קונבולוציה, שתי שכבות Batchnormalization ו שכבת Maxpolling. להלן סכמת מלבנים המתארת את מודל הרשת:



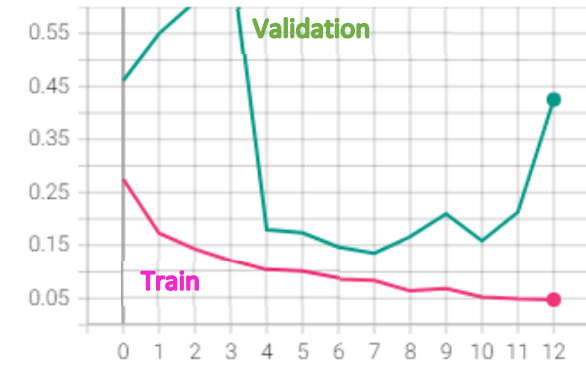
epoch_accuracy

epoch_accuracy
tag: epoch_accuracy

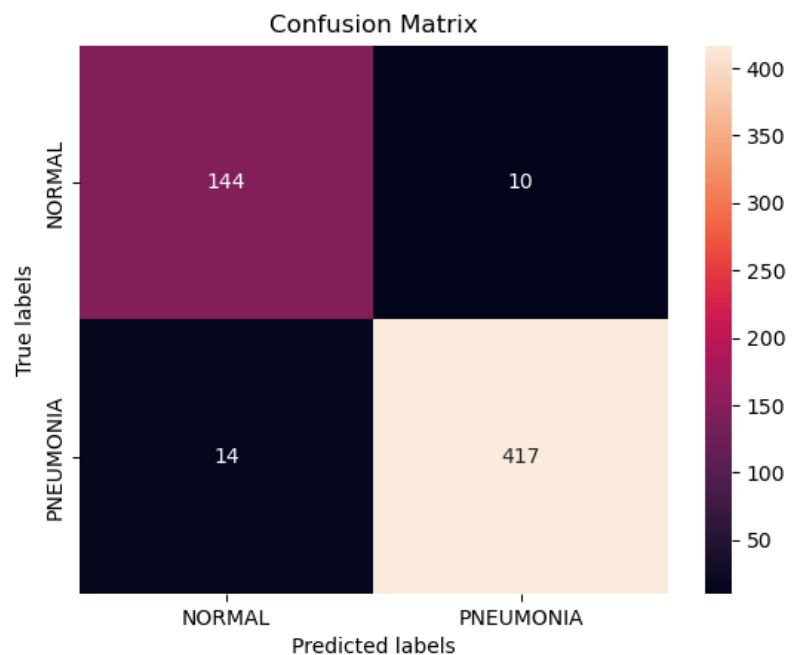


epoch_loss

epoch_loss
tag: epoch_loss



```
test loss, test acc: [0.1272963583469391, 0.9589743614196777]
Precision: 0.935064935064935 Recall: 0.9113924050632911
```



באימון מודל זה השתמשנו במנגנון early stopping למניעת overfitting עם הפרמטרים הבאים:

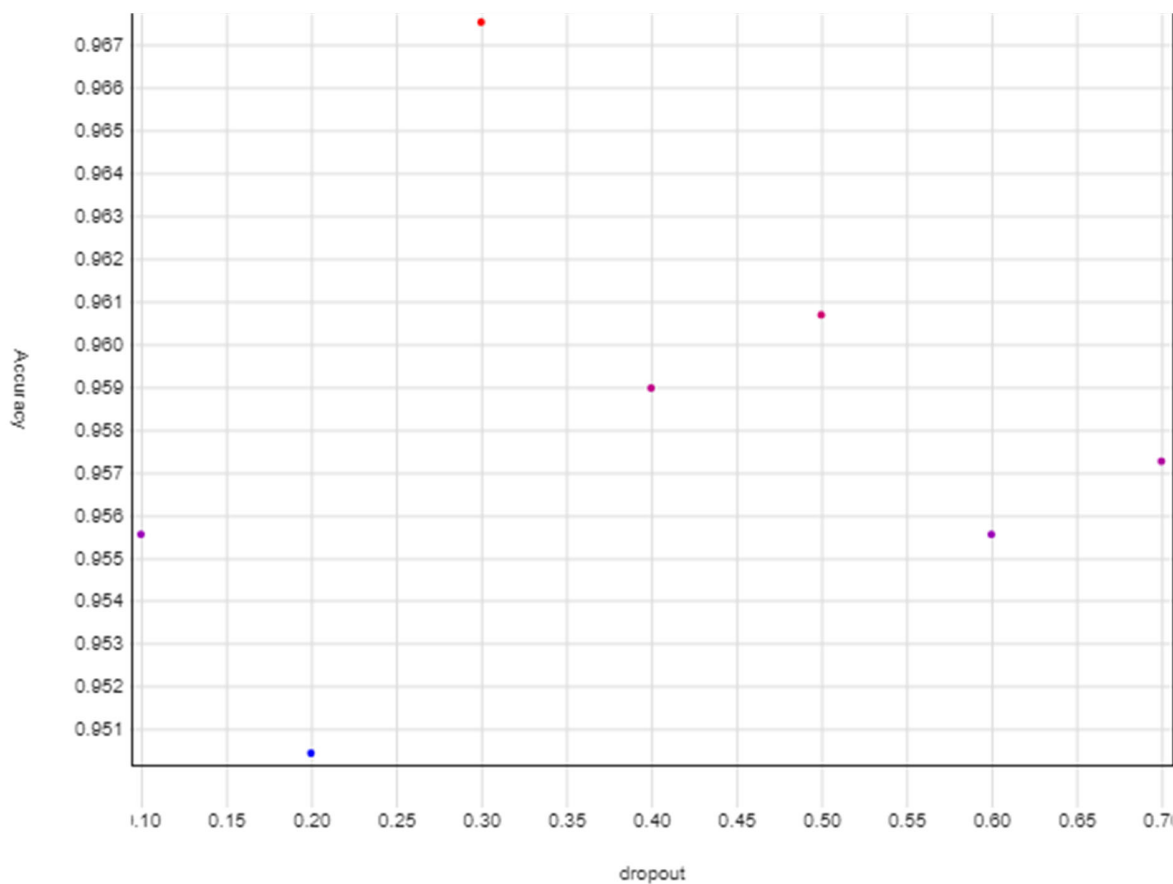
```
monitor='val_loss',  
mode = 'auto',  
patience=3,  
restore_best_weights = True
```

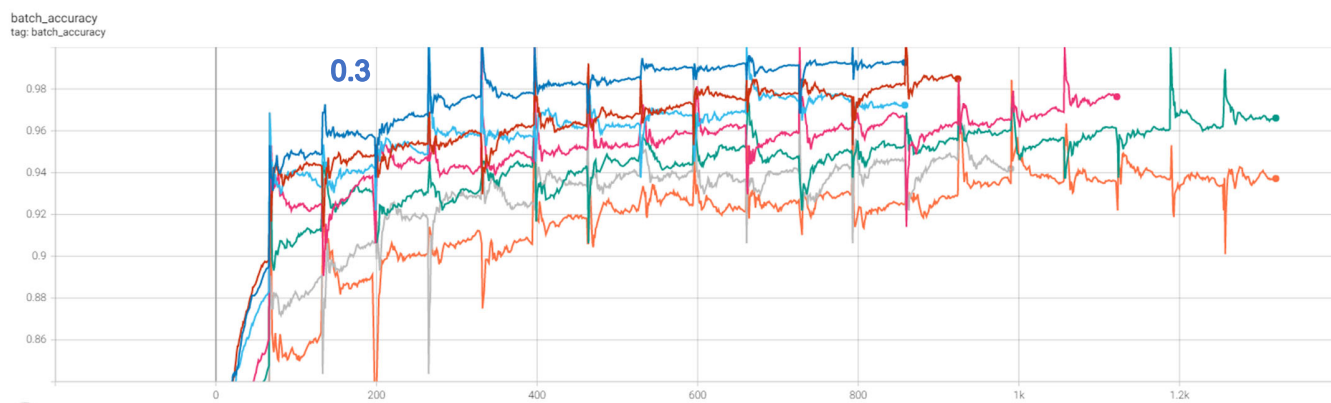
ניתן לראות שב epoch מספר 12 המערכת עצרה מעצמה בכדי למנוע overfitting ולא המשיכה ל 13 ה epochs שנשארו. בנוסף, מנגנון זה מחזיר את המשקולים למצב הטוב ביותר (epoch 10).

השפעת dropout: hyperparameter

התבקשנו להראות כיצד שינוי של hyperparameter יכול להשפיע על הרשת שבנינו. בחרנו להראות כיצד שינוי של ה Dropout משפיע על אחוזי הדיוק של הרשת. להלן התוצאות:

Trial ID	Show Metrics	dropout	Accuracy
273349e7fdf41...	<input type="checkbox"/>	0.20000	0.95043
63d857ebf5b1d...	<input type="checkbox"/>	0.30000	0.96752
802c669fae024...	<input type="checkbox"/>	0.50000	0.96068
c5a7869709b73...	<input type="checkbox"/>	0.10000	0.95556
cbd74acc330e9...	<input type="checkbox"/>	0.40000	0.95897
d12ca4a371e73...	<input type="checkbox"/>	0.60000	0.95556
d7dd341f58909...	<input type="checkbox"/>	0.70000	0.95726





מן התוצאות ניתן לראות שעבור ערך dropout של 0.3 נקבל ערך דיוק הכי גבוה מבין כל הערכים שניסנו ולכן נבחר להשתמש בערך זה לרשת שאנו מתכננים.

בדיקת השפעת augmentations על ביצועי המודל

לאחר עדכון פרמטר ה dropout ל 0.3 אימנו את המודל מחדש ל 20 epochs וקיבלנו את התוצאות הללו:

```
test loss, test acc: [0.1473180204629898, 0.964102566242218]
```

לאחר מכן הרצנו אימון נוסף (כהמשך לאימון הקודם) על סטים עם אוגמנטציות אך התוצאות מאכזבות ולא ראינו שיפור אלא רק ירידה משמעותית בדיוק המערכת. להלן התוצאות:

```
test loss, test acc: [3.127434015274048, 0.7299145460128784]
```

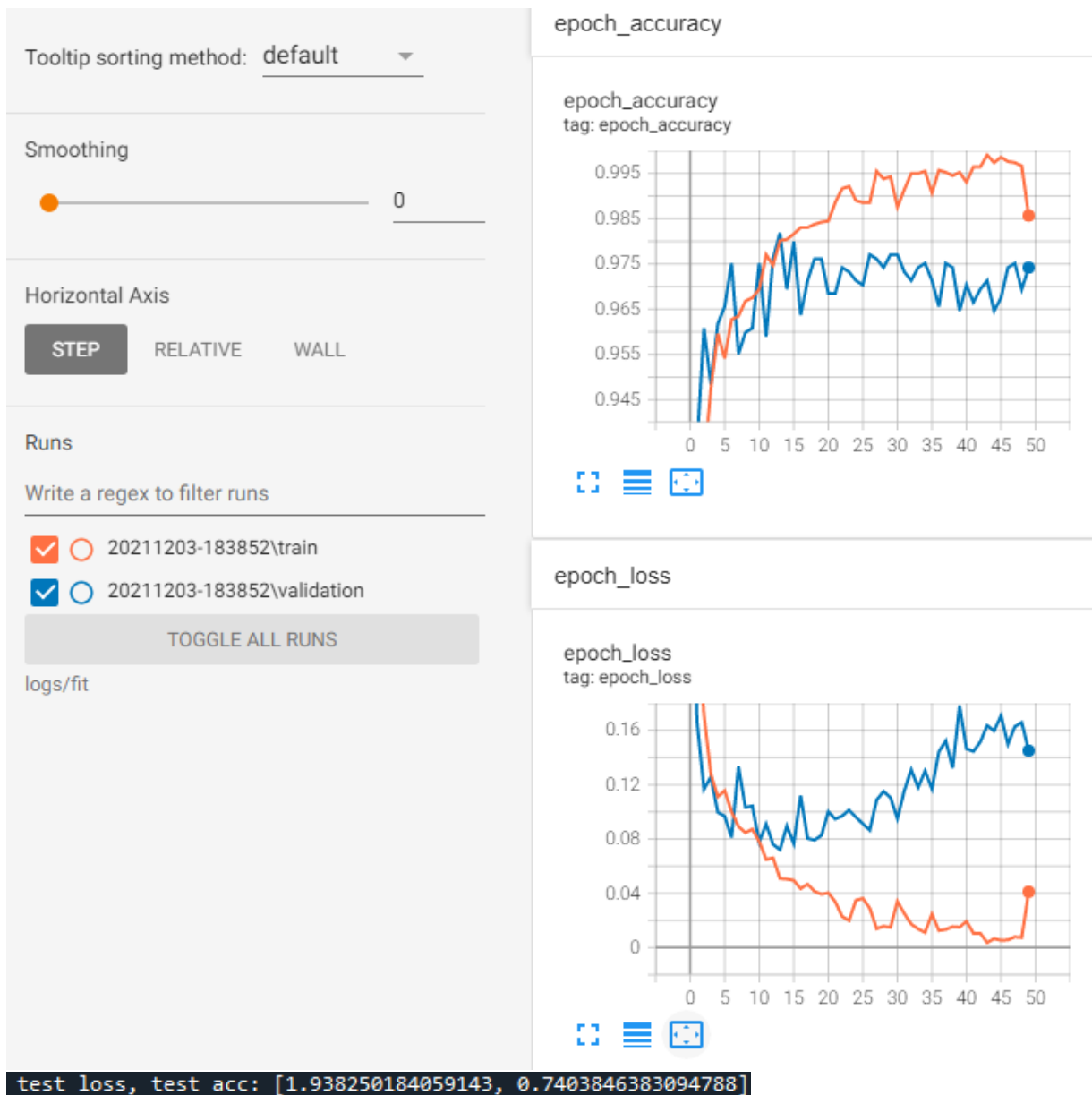
האוגמנטציות שביצענו על מאגר התמונות הינן: רוטציה (סיבוב) ושינוי ניגודיות (contrast). בגלל שמאגר התמונות שלנו מכיל תמונות X-ray אשר מצולמות תמיד באותה הזווית ובאותו color palette שינוי של זווית התמונה והניגודיות רק ישפיעו על המערכת לרעה ויהרסו את תהליך הלמידה שלה.

שיפור של מאגר ה data

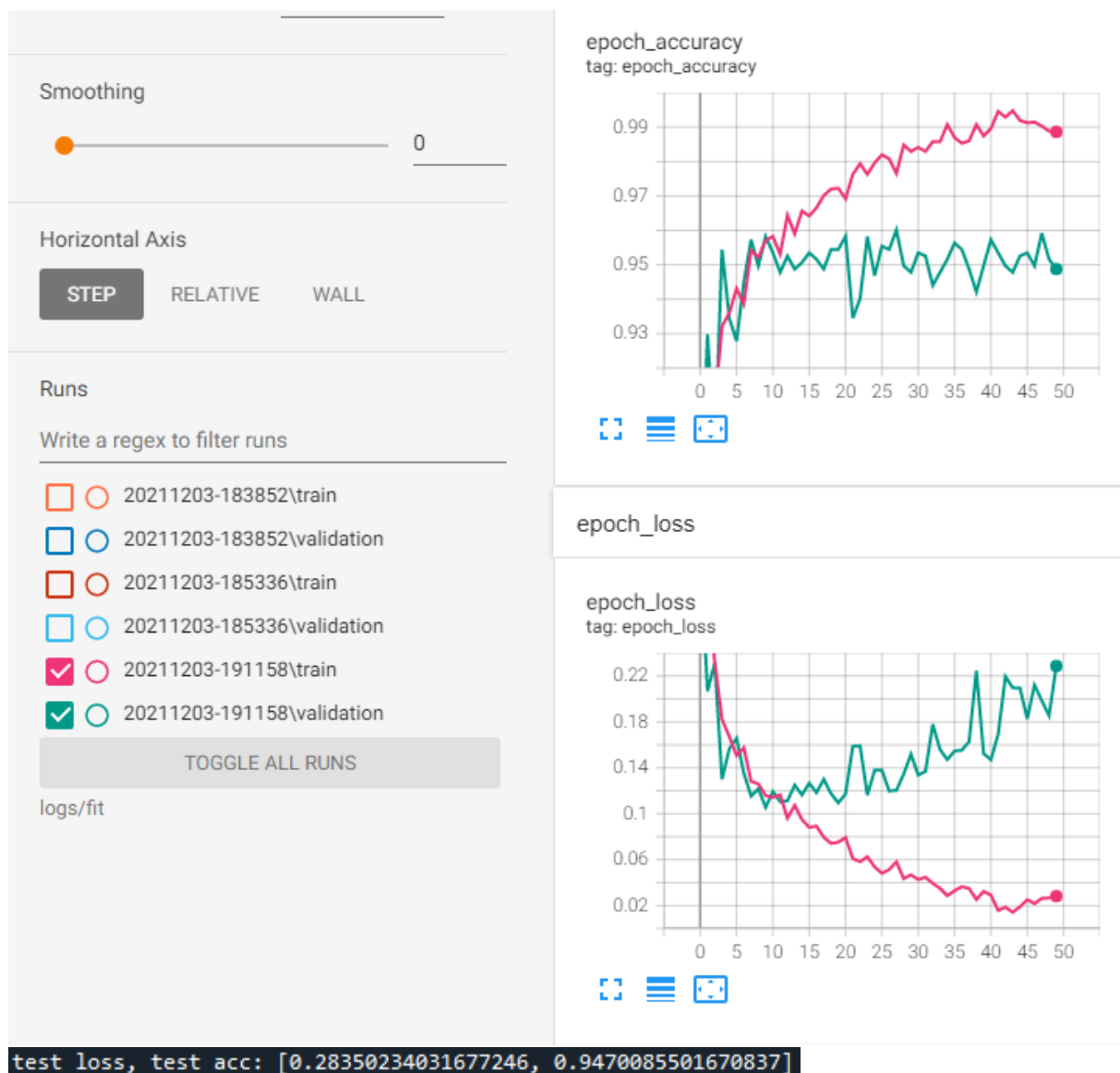
כאשר עברנו על סט המידע שמנו מיד לב שיש לנו בחלק של הוולידציה רק 16 תמונות. ביצוע של וולידציה על סט של 5,863 בעזרת רק 16 תמונות לא יעיל לצורת הלמידה של הרשת ולכן שינו את מאגר ה data לחלוקה של 70% אימון, 20% ולידציה ו 10% בחינה.

בנוסף לכך שמנו לב גם שלמרות שאנו מקבלים אחוזי דיוק גבוהים באימון והוולידציה של הרשת אנו מקבלים אחוזי דיוק נמוכים ב Test ולכן הסקנו שמאגר ה Test מכיל תמונות שונות מאוד לעומת שאר המאגר ולכן ארגנו מחדש את מאגר התמונות לפי החלוקה 70% אימון, 20% ולידציה ו 10% בחינה בצורה רנדומלית. להלן תמונות מריצות של המערכת עם המאגר המקורי ואם המאגר לאחר חילוק התמונות מחדש:

סטים מקוריים:



סידור מחדש:



GitHub

הזמנה ל GitHub תישלח במייל