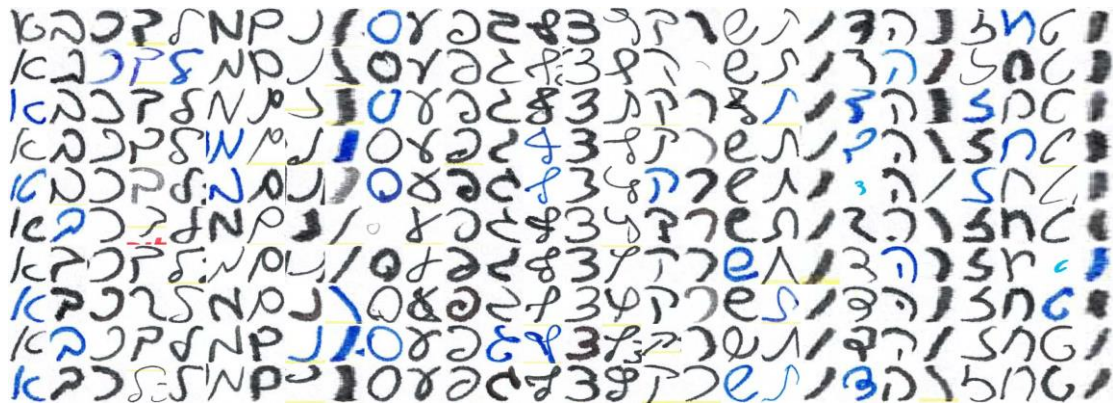


# Classifier for Handwritten Hebrew Letters with Neural Networks

תאריך ההגשה: 27.04.2025, שעה 23:59

בתרגיל זה, תאמנו Feedforward NN כדי לסווג תמונות של אותיות ממאגר HHD, שמורכב מאותיות בכתב יד בעברית. המאגר HHD מכיל בסביבות 5000 תמונות של אותיות בודדות. תמונות אלו מחולקות ל-27 תתי-קבוצות (תתי-תיקיות). כל תיקייה מכילה תמונות של אות מסוימת מתוך האלפבית העברי. פרטים אודות המאגר HHD ניתן למצוא ב-[1].



איור 1: דגימה ממאגר HHD של אותיות בכתב יד

מטרת התרגיל היא לאמן רשת נירונים לסווג אותיות.

העבודה תחולק למספר צעדים:

1. עיבוד מקדים (pre-processing)

a. המירו את התמונה לגווני אפור (greyscale)

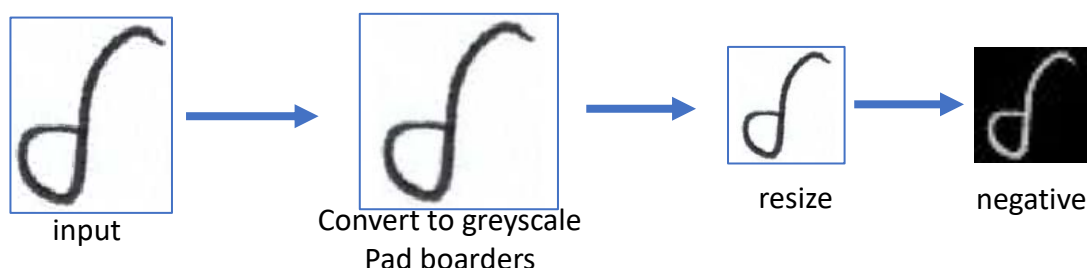
b. הוסיפו לתמונה ריפוד לבן (padding) כדי שגודלה יהיה מרובע

- אם רוחב התמונה קטן מגובה, יש להוסיף Padding מימין ומשמאל  
- אחרת, אם רוחב גדול מגובה, יש להוסיף Padding מלמעלה ולמטה

אפשר להיעזר בפונקציית [cv2.copyMakeBorder](#) של OpenCV.

c. העבירו את התמונה לגודל אחיד (32, 32) בעזרת פונקציית `cv2.resize`

d. הפכו את התמונה ל-negatives. לרוב, עבודה עם negative בתמונות טקסט משפרת את הדיוק. הפיכת תמונה לנגטיב ניתן לבצע באמצעות הפקודה  
$$\text{dst} = 255 - \text{dst}$$



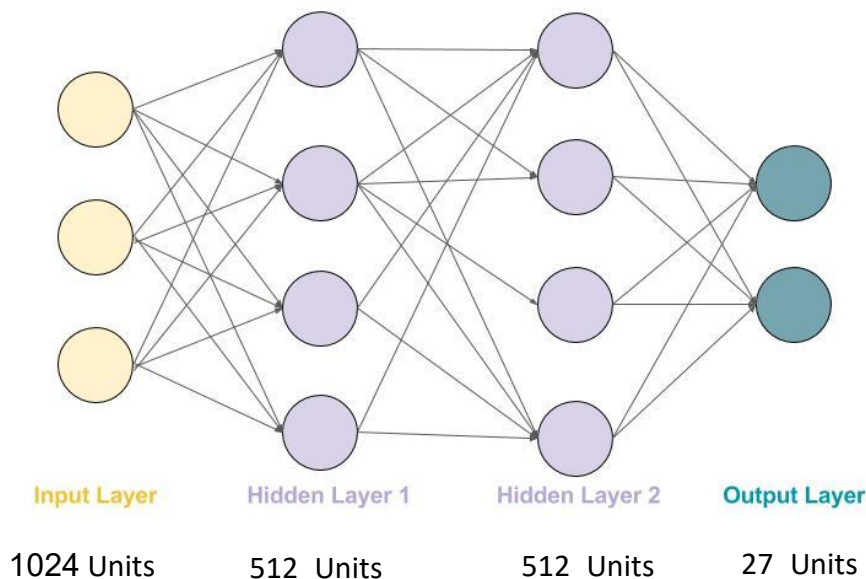
1. חילקו את המאמגר באופן אקראי לשלוש קבוצות training, validation, and testing sets. החלוקה תהיה ביחס 80% ל-training, 10% ל-validation, ו-10% ל-testing. חידוד: תמונות של כל אות צריכות להופיע בכל אחת מהקבוצות ביחס 80%:10%:10%.

בשלב 2, אתם תשתמשו ב-training set כדי לאמן רשת נוירונים, וב-validation כדי לבדוק את הביצועים של המודל בקונפיגורציות שונות (למשל, הוספת רגולריזציה). את המודל הסופי (זה שנתן תוצאות הכי טובות על validation set) תריצו על ה-testing set כדי לחשב את הדיוק על קבוצת הנתונים אותה המודל לא ראה במהלך האימון.

2. אימון (training).

בשלב זה יש לבנות ולאמן רשת נוירונים בקונפיגורציות שונות. המודל יהיה מורכב מ:

- שכבת קלט עם 1024 units ו-ReLU activation function (תמונה בגודל  $32 \times 32$  מעבירים לוקטור עם 1024 אלמנטים)
- שתי שכבות מסותרות (hidden layers) עם 512 units כל אחת
- שכבת פלט עם 27 units ו-softmax activation function



- יש לאמן את רשת במספר קונפיגורציות, 50 epochs בכל קונפיגורציה
1. ללא regularization
  2. עם הוספת L1 regularization עם בכל השכבות פרט לשכבת הפלט
    - עם ערכים של למבדה  $\lambda = 0.01$ ,  $\lambda = 0.001$
  3. עם הוספת L2 regularization בכל השכבות פרט לשכבת הפלט
    - עם ערכים של למבדה  $\lambda = 0.01$ ,  $\lambda = 0.001$

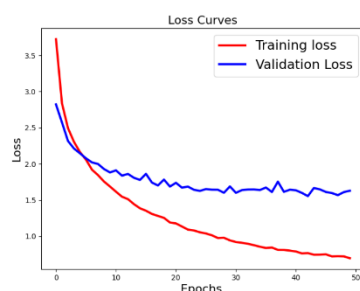
4. עם הוספת dropout עם  $p = 0.5$  בכל השכבות פרט לשכבת הפלט
5. עם הוספת שתיהם L2 ו- dropout בכל השכבות פרט לשכבת הפלט
- עם ערכים של למבדה  $\lambda = 0.001$ ,  $\lambda = 0.01$  ו-  $p = 0.5$

לאחר הניסויים, יש לבחור את המודל שנתן ביצועיים הכי טובים על validation set

3. הערכת ה-NN על testing set.  
ברגע שמצאתם את הקונפיגורציה הטובה ביותר, יש להעריך את התוצאות של NN על testing set ולדווח את התוצאות.

### פלט התוכנית יכלול

1. קובץ טקסט בשם "results.txt" שיכיל:
  - a. קונפיגורציה של המודל הסופי
  - b. תמונה עם עקומות Loss על training & validation sets עבור המודל הסופי. ראו דוגמה:



- c. דיוק אליו הגיע המסווג על testing set עבור כל אחת מהאותיות (27 אותיות שונות) ודיוק ממוצע בפורמט

Letter	Accuracy
0	...
1	...
...	
26	...
<hr/>	
Avg	...

2. [Confusion matrix](#) עבור התוצאות בקובץ excel/scv בשם "confusion\_matrix.csv". בלינק המצורף של ויקיפדיה נמצא הסבר מהי [Confusion matrix](#)

הגשה:

יש להגיש קובץ zip שמכיל את הקבצים הבאים:  
1. קובץ קוד עם התוכנית.

הקוד שהוגש צריך לכלול עיבוד מקדים, ניסויים עם אימון ובחירת מודל, והרצת המודל הסופי על `testing set`.

2. קובץ [readme.txt](#)

The readme.txt should include the following information:

**The authors' contact information**

**Description**

A brief description of your program.

**Environment**

Describe the OS and compilation requirements needed to compile and run the program

**How to Run Your Program**

Provide instructions and examples so users how to run the program.

Describe if there are any assumptions on the input.

You can also include screenshots to show examples.

3. קבצים "results.txt" ו- "confusion\_matrix.csv" בפורמט שמתואר למעלה

אופן הבדיקה:

הבדיקה תתבצע בצורה פרונטלית. מועדי הבדיקה ייקבעו בהמשך.

בכל שימוש המאגר HHD, יש לתת הפנייה ל-[1]

## עבודה נעימה!

### References

[1] I. Rabaev, B. Kurar Barakat, A. Churkin and J. El-Sana. The HHD Dataset. *The 17<sup>th</sup> International Conference on Frontiers in Handwriting Recognition*, pp. 228–233, 2020.