

תוכן עניינים

3.....	מבוא
4.....	מדריך למשתמש
5.....	מדריך למפתח
7.....	מסקנות הרצת המודל
8.....	רפלקציה
9.....	ביבליוגרפיה
10.....	נספחים

מבוא:

התוכנה שפיתחתי נקראת Clothing Classifier - מקטרג בגדים

מטרת התוכנה: מטרת התוכנה היא לזהות את סוג פריט הלבוש אשר המשתמש מכניס לתוכה מתוך רשימה של פרטי לבוש הכוללים: תיק גב, תיק צד, כפכפים, משקפי שמש, נעליים, חולצות-T ושעוני יד

אופן פעולת התוכנה: התוכנה לומדת ממאגר תמונות המכיל מעל 7000 תמונות של פרטי הלבוש מהרשימה, אותם היא לומדת לזהות בעזרת ספריות Keras ו TensorFlow ובכך יודעת לשייך את הפריט אותו מכניס המשתמש לאחת מהקטגוריות ברשימה.

התחלת הרצת הפרויקט: בכניסה לתוכנה יש למשתמש כמה אפשרויות, על מנת להזין לתמונה פריט לבוש יש תחילה לטעון מודל או ללמוד מודל חדש.

סיום ההרצה: בסיום ההרצה תראה התוכנה את שמו של החפץ אותו היא משערת שהיא רואה.

דרישות להרצה: Python3.6.x מכיוון שגרסאות מעל לא תומכות ב TensorFlow, הספריות בהן משתמשים הן, TensorFlow, Keras, os, Pil, Numpy, Sys

מדריך למשתמש:

ביצוע התקנות - תחילה יש להתקין למחשב את גרסת הפיטון העדכנית (רצוי) ולאשר את הוספת ה PATH, יש לפתוח Command Prompt ולהוריד את הספריות בדרכים הבאות.

pip install tensorflow

pip install keras

pip install Pillow

מסך ראשי - במסך הראשי של התוכנה ישאל המשתמש מה ירצה לעשות

```
Welcome To Bar's Project, Choose What You Desire
1. Load The Model
2. Test An Image
3. Train The Model
4. Exit
```

לטעון מודל, לבדוק תמונה, ללמוד מודל או לצאת אלו האופציות הנתונות

על המשתמש לטעון את המודל או לאמן מודל חדש

במקרה וינסה המשתמש לטעון תמונה לפני שטען או למד מודל, תדפיס התוכנה שיש לטעון מודל

קודם

```
Welcome To Bar's Project, Choose What You Desire
1. Load The Model
2. Test An Image
3. Train The Model
4. Exit
2

Please Load The Model First
Welcome To Bar's Project, Choose What You Desire
1. Load The Model
2. Test An Image
3. Train The Model
4. Exit
```

לאחר שנטען המודל או נלמד מודל חדש ניתן ללחוץ 2 ולגרור תמונה ל Command Prompt אותה ינסה המודל לקטרג.

מדריך למפתח

התוכנה בנוייה משתי מחלקות:

מחלקה Model - מטרתה לרכז את מבנה המודל ושכבותיו לשינוי קל ויעיל, נמצאת בקובץ נפרד

מזומנת על ידי המחלקה הראשית Project

מחלקה Project - מחלקה זאת כוללת את כלל פעולות הפרויקט ומרכזת אותן בצורה נוחה ונגישה פעולות אלה הן:

פעולת folderToTest - פעולה זו מקבלת מסלול לתיקייה מלאה בתמונות אותה היא מצמצמת לכ 20% מנפחה המקורי והופכת אותה לתקיית בדיקה, פעולה זו לא מחזירה דבר.

פעולת predictImg - פעולה זאת מקבל מודל ותמונה את התמונה מריצים על פעולת ה predict

של המודל ומחזירה את מספר החפץ מתוך רשימת החפצים, נקראת בפעולת predict

פעולת showMenu - פעולה זו מציגה את תפריט המשתמש, קולטת קלט בתור בחירת המשתמש,

בודקת את תקינותו ומחזירה את בחירת המשתמש לmain, נקראת בפעולת main

פעולת train_model - פעולה זו מאמנת את המודל שבמחלקה Model ומחזירה את המודל

המאומן, נקראת בפעולת main.

פעולת load_model - פעולה זו טוענת את המודל השמור ומכינה אותו לפעולה, נקראת בפעולה ה

main

פעולת load_photo - פעולה זו מקבל ממדי תמונה ומסלול לתמונה אותה היא קולטת, מקטינה או

מגדילה למימדים הנתונים ומשנה לה את הצבעים בהתאם לצורכי התוכנה ומחזירה אותה מוכנה,

נקראת בפעולת ה predict

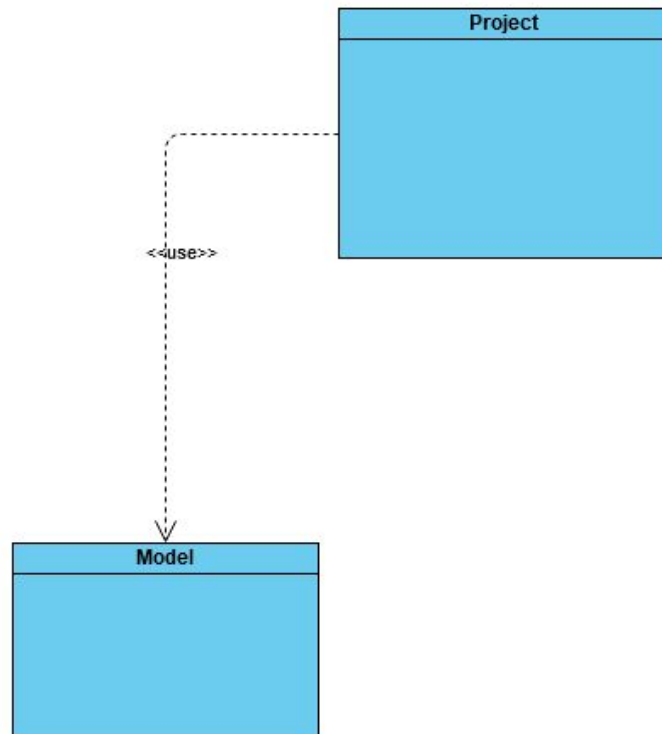
פעולת predict - פעולה זו מקבלת מודל, קולטת תמונה ומשייכת את התמונה לאחת מהקטגוריות

אותן למד המודל, משתמשת בפעולה predictImg ומדפיסה את שמו של החפץ בהתאם למקומו

ברשימה, נקראת בפעולת ה main

פעולת main - פעולה זו בודקת את ההחזרה מ showMenu ופועלת בהתאם לדרישות

המשתמש



מסקנות הרצת המודל

```
7629/7629 [=====] - 388s 51ms/step - loss: 0.6461 - accuracy: 0.8287 - val_loss: 0.2811 - val_accuracy: 0.9442
Epoch 2/3
7629/7629 [=====] - 352s 46ms/step - loss: 0.2068 - accuracy: 0.9446 - val_loss: 3.3379e-06 - val_accuracy: 0.9869
Epoch 3/3
7629/7629 [=====] - 344s 45ms/step - loss: 0.1861 - accuracy: 0.9571 - val_loss: 3.5763e-07 - val_accuracy: 0.9875
7629/7629 [=====] - 157s 21ms/step
Test accuracy: 0.9851881265640259
```

אחוזי דיוק בעת בדיקה accuracy 98.51%

loss 0.1861

הloss הוא שגיאתו של המודל - נמוך מאוד

בעוד ה accuracy היא רמת הדיוק שלו - גבוהה מאוד

רפלקציה

העבודה הייתה מעשירה, למדתי דברים חדשים שלא חשבתי שאלמד לבד.
כמו כן אני מאמין שלמדתי דרכי למידה חדשות הכוללות לימוד עצמי מעמיק תוך ניסוי וטעייה וקריאת
מאמרים באינטרנט.

ביבליוגרפיה

למדתי מאתרים כמו

<https://www.analyticsvidhya.com/blog/2019/01/build-image-classification-model-10-minutes/>

<https://www.tensorflow.org/>

<https://keras.io/>

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 158, 118, 32)	896
conv2d_2 (Conv2D)	(None, 156, 116, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 78, 58, 64)	0
dropout_1 (Dropout)	(None, 78, 58, 64)	0
flatten_1 (Flatten)	(None, 289536)	0
dense_1 (Dense)	(None, 128)	37060736
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 7)	903
Total params: 37,081,031		
Trainable params: 37,081,031		
Non-trainable params: 0		

```

7629/7629 [=====] - 388s 51ms/step - loss: 0.6461 - accuracy: 0.8287 - val_loss: 0.2811 - val_accuracy: 0.9442
Epoch 2/3
7629/7629 [=====] - 352s 46ms/step - loss: 0.2068 - accuracy: 0.9446 - val_loss: 3.3379e-06 - val_accuracy: 0.9869
Epoch 3/3
7629/7629 [=====] - 344s 45ms/step - loss: 0.1861 - accuracy: 0.9571 - val_loss: 3.5763e-07 - val_accuracy: 0.9875
7629/7629 [=====] - 157s 21ms/step

Test accuracy: 0.9851881265640259

```

