

דף הבית



שם בית הספר: ישיבת תמר

שם העבודה: פרויקט למידת מכונה מכון שמי'ר

שם התלמיד: גלעד וינברגר

ת.ז. התלמיד: 216921403

שם המנחה: ליאל סנדלר, מירב צברי

שם החלופה: למידת מכונה - *deep learning*

תאריך תחילת עבודה: 13/11/2024

תאריך הגשה: 1/6/2025

תוכן עניינים

דף הבית	1
תוכן עניינים	2
מבוא	3
רקע לפרויקט	3
קהל יעד	3
אופן פעולה של הפרויקט	3
תהליכי המחבר	3
אתגרים מרכזים	3
מבנה הפרויקט	4
איסוף, הכנה וניתוח הנתונים	4
בנייה ואמון המודל	4
יישום הפרויקט (שימוש משותם)	6

מבוא

רקע לפרויקט

הפרויקט זהה נוצר בשביל לעזרה לחוקר מכון שחם, ד"ר ליאור גור. הוא חוקר עלי גפן וצריך לסייע אוטם לפיה מחלות. הפרויקט שלו עוזר לו לחשוף כמה שטח לבן יש על כל עלה, וזה מקדם את המחקר שלו יותר מהר ויעיל.

קהל יעד

הפרויקט מיועד לחוקרים שחוקרים את אותו הדבר ורוצים לחסוך זמן בעבודה שלהם.

אפקט פעולה של הפרויקט

המשתמש מעלה תמונה של עלה דרך ממתק משותם, והמערכת מחזירה לו את כמות השטח הלבן על העלה.

תהליכי המחקר

- המצב היום בשוק הוא שהרבה חוקרים לא יכולים להשתמש בכלים ניתוח ממוחשבים מתקדמים, אז הם עושים הרבה עבודה ידנית.
- החידוש בפרויקט הוא שהוא יכול לזהות כמה אחז לבן יש על עלי גפן, וזה עוזר לזהות מחלת שנקראית כשותית הגפן.
- השימוש בטכנולוגיות שלא למדנו בבית הספר:
 - Django - זה פיתוחardiatri Full Stack בפייטון, והוא עוזר ליצור אתרים בקלות. יש לו גם ממשק מנהל מבנה עם גישה למסד הנתונים.
 - יצירת מסד הנתונים נעשית בפייטון, ומסד הנתונים הוא תיקייה של תמונות, ופרטן כל תמונה נתונים בשם שלה.

אתגרים מרכזיים

במהלך הפרויקט היו לי כמה אתגרים:

- למדתי על למדידת מכונה.
- הייתה צריכה להבין את המתמטיקה שמאחוריה האלגוריתמים של למדידת מכונה.

הפרויקט נותן פתרון לצורך בכלי סיווג ומיפוי שלא צריך הרבה עבודה, יוכל לעבוד מהר, יעיל ולחסוך זמן מחקר.

מבנה הפרויקט

איסוף, הכנה וניתוח הנתונים

מבנה הנתונים הוא תיוקה של כל התמונות השמורות בצורה שהאחז לבן של העלה נמצא בשם של הקובץ.

לדוגמה: בקובץ "50_leaf.jpg" יהיו 50% לבן בתמונה. התמונות והערכים לדאטה נלקחו ממכוון שמיר במחקר ליאור גור. אחרי יצירת התיקייה הראשונית של התמונות (כ-40 תמונות), כל תמונה משוכפלת 7 פעמים על ידי קוד פיתון - כל פעם עם הזזה ושיקוף בצורה שונה על מנת ליצור את כמות הדאטה המksamלית האפשרית לאימון המודל. וכל התמונות משוכפלות לתיקייה נפרדת..

בניה ואימון המודל

ארQUITוטורת רשת הנוירונים (CNN - Convolutional Neural Network)

בפרויקט זה נבחרה ארQUITוטורה של רשת נוירונים מסווג CNN לצורך חיזוי אחז השטח הלבן בעלי גפן. רשתות CNN הן סוג של רשתות נוירונים המציגות בעיבוד תמונות, והן בנויות מ שכבות שונות שתפקידן לחץ מאפיינים רלוונטיים מהתמונה ולבצע את המשימה הסופית (במקרה שלנו, רגרסיה - חיזוי ערך מסווג רציף).

המודל הכללי הספציפי שבנית מרכיב משנה חלקים עיקריים:

1. שכבות לחילוץ מאפיינים (Features): חלק זה כולל סדרה של שכבות קונבולוציה (CNN) ושכבות איגום (Max Pooling Layers), בינהן ממוקמות פונקציות הפעלה מסווג ReLU. שכבות הקונבולוציה לומדות פילטרים קטנים שמשהרים דפוסים מקומיים בתמונה (כגון קצוט, פינות, מרקמים). שכבות ה-ReLU מוסיפות אי-ליניאריות לרשת, מה שמאפשר לה ללמידה פונקצייתית נוספת. שכבות האיגום מפחיתות את המימד המרחבית של מफות המאפיינים, ובכך מגבירות את העמידות של המודל לשינויים קטנים בתמונה (כגון תזוזה או שינוי קטן בקנה מידה).
2. שכבות רגרסיה (Regressor): חלק זה כולל סדרה של שכבות ליניאריות (Fully Connected Layers). לאחר שכבות חילוץ המאפיינים מצוא את המידע הרלוונטי מהתמונה, הן משוטחות לווקטור חד-ממדי, אשר מוען לשכבות הליניאריות. שכבות אלו מביצעות את החיזוי הסופי של אחז השטח הלבן.

פירוט השכבות במודל:

1. שכבה קונבולוציה ראשונה: מקבלת תמונה צבעונית (3 ערוצים - RGB) ומפיקה 32 מפות מאפיינים. גודל המסנן הוא 3×3 , הצעד הוא 1, והrifod הוא 1 (שומר על גודל התמונה).
2. פונקציית הפעלה ReLU: שולטת על הפלט של שכבת הקונבולוציה הראשונה.
3. שכבת איגום ראשונה: מפחיתה את גודל מפות המאפיינים בחצי (גודל מסנן 2×2).
4. שכבת קונבולוציה שנייה: מקבלת 32 מפות מאפיינים ומפיקה 64 מפות מאפיינים (גודל מסנן 3×3 , הצעד 1, Rifod 1).
5. פונקציית הפעלה ReLU: שולטת על הפלט של שכבת הקונבולוציה השנייה.

6. שכבה איגום שנייה: מפחיתה את גודל מפות המאפיינים בחצי.
7. שכבה קונבולוציה שלישית: מקבלת 64 מפות מאפיינים ומפיקה 128 מפות מאפיינים (גודל מסנן 3x3, צעד 1, ריפוד 1).
8. פונקציית הפעלה ReLU: שולטת על הפלט של שכבה הקונבולוציה השלישית.
9. שכבה איגום שלישית: מפחיתה את גודל מפות המאפיינים בחצי.
10. שכבה LINEARITAT ראשונה: מקבלת את הפלט המשוטח של שכבות חילוץ המאפיינים (128 ערכאים בגודל 28x28, סך הכל $128 * 28 * 28 = 28,192$ נוירונים) ומפיה אותו ל-256 נוירונים.
11. פונקציית הפעלה ReLU: שולטת על הפלט של השכבה הLINEARITAT הראשונה.
12. שכבה LINEARITAT שנייה (שכבת הפלט): מקבלת 256 נוירונים וממפה אותם לננוון יחיד, שמייצג את החיזוי של אחוז השטח הלבן (ערך בין 0 ל-1 שמייצג ערך בין 0 ל-100%).

הסבר על שכבות ה-CNN:

- שכבה קונבולוציה (Conv2d): שכבה זו מבצעת פעולה קונבולוציה בין המסלנים שלה (גרעינים) לבין התמונה (או מפת המאפיינים משכבה קודמת). כל מסנן לומד לzechות מאפיין ספציפי בתמונה, כגון קו אנכי, פינה, או מרקם מסוים. הפרמטרים של שכבה זו כוללים את מספר ערוצי הקלט, מספר ערוצי הפלט (מספר המסלנים), גודל המסלן (בדרך כלל 3x3 או 5x5), הצעד (כמה המסלן ZZ בכל פעם), והrifod (הוספה שלולים סביב התמונה כדי לשנות על גודל הפלט).
- פונקציית הפעלה ReLU: פונקציה זו שולטת על הפלט של שכבה הקונבולוציה (או שכבה LINEARITAT) ומחליפה ערכים שליליים באפס. ערכים חיוביים נשארים כמו שהם.
- שכבת איגום (MaxPool2d): שכבה זו מפחיתה את גודל מפות המאפיינים על ידי בחירת הערך המקסימלי מתוך חילון קטן (בדרך כלל 2×2) שז' על פני המפה. פעולה זו מפחיתה את מספר הפרמטרים בראשת, מקלת על האימון, ומשפרת את העמידות של המודל לשינויים קטנים בתמונה.
- שכבה LINEARITAT (Linear): שכבה זו מבצעת טרנספורמציה LINEARITAT על הקלט שלה (כפל מטריצות וחיבור וקטור). בשכבות האחרונות של רשת CNN, מפות המאפיינים המשוטחות מוגזנות לשכבות LINEARITAT כדי לבצע את החיזוי הסופי.

בחירה ספריות תוכנה:

המודל נבנה בשפת Python תוך שימוש בספריית PyTorch. PyTorch מספקת כלים גמישים ויעילים לבנייה, אימון והערכתה של מודלים של רשתות נוירוניים. בנוסף, נעשו שימוש בספריות הבאות:

- NumPy: לביצוע פעולות מתמטיות בסיסיות ויעילות על מערכים.
- Matplotlib: להציג גרפים של התקדמות האימון ותוצאות החיזוי.
- PIL (Pillow): לטיפול בתמונות (טעינה ו שינוי גודל).
- torchvision.transforms: לביצוע טרנספורמציות שונות על התמונות לפני הכנסתן למודל (כגון שינוי גודל ומרה לטנסור).
- DataLoader-torch.utils.Dataset: לניהול וטעינה יעליה של הנתונים במהלך האימון.

תהליכי האימון

לאחר בניית ארכיטקטורת המודל, הוא מתאים על מאגר הנתונים של תמונות על הغان. תהליכי האימון הטענו במשך מספר סבבים (epochs), כאשר בכל סיבוב המודל עובר על כל הנתונים. בכל איטרציה של האימון, המודל מקבל תמונה קלה, מבצע חיזוי של אחוז השטח הלבן, ומשווה את החיזוי לערך האמתי (הלייבל) שמופיע בשם הקובץ. ההפרש בין החיזוי לערך האמתי מחושב באמצעות פונקציית Loss (במקרה זה - MSE - Mean Squared Error), ומטרת האימון היא למזער את ערך loss על ידי עדכון משקל המודל באמצעות אלגוריתם אופטימיזציה (במקרה זה, Adam עם קבוע למידה של 0.001). במהלך האימון, רמת הדיווק של המודל בחיזי הולכת וגדלה, והשגיאה (Loss) הולכת וקטנה. תהליכי האימון נמשך עד שמספר הסבבים המוגדר מראש (30 במקרה זה) מסתיים. במהלך האימון, נעשה שימוש בסרגל התקדמות (tqdm) כדי לעקוב אחר התקדמות האימון בכל epoch.

ישום הפרויקט (ממשק משתמש)

ממשק המשתמש בנוי בשפת פיתון עם Django Framework, כלי פשוט ונוח ליצור אתרי Full-stack כולל חיבור מובנה מראש לקובץ SQLite המשמש כ-Database, וממשק ניהול מובנה (Create, Read, Update, Delete) בסיסיות (CRUD).

בכונסה לכתובת האתר נכנסים לדף הבית עם אופציה להוספה של תמונה, והחיזוי לתמונה מתבצע באמצעות המודל שנמצא בצד השירות של האתר. לאחר מכן התמונה עוברת את המודל, ותוצאת המודל מוכפלת ב-100 (מספר בין 0 ל-100) ומוצגת על המסך.

מדריך למפתח

סיכון זום: 12/12/2024

המעבדה עוסקת במחלות צמחיים (מחלות נוף, קרקע וכו'), הנגרמות ע"י פטריות וירוסים.
אנחנו עוסקים עם מחלות נוף (מחלות שנראות לעין)

בפרויקט מתמקדמים במחלת כשותית הגפן, הנגרמת על ידי דמו-פטרייה כמו עובש-מים.

נקודות:

- לתקן את המבואה (لتאר את המצב הנוכחי) ואת מטרת הפרויקט והיתרונות בכך, לפני תיאור של למי הפרויקט
-

איפוא:

 MachinLearning-Project (1) (1) (1).pdf

מחוון:

 machinLearningProject_Finalt-5 (1).pdf