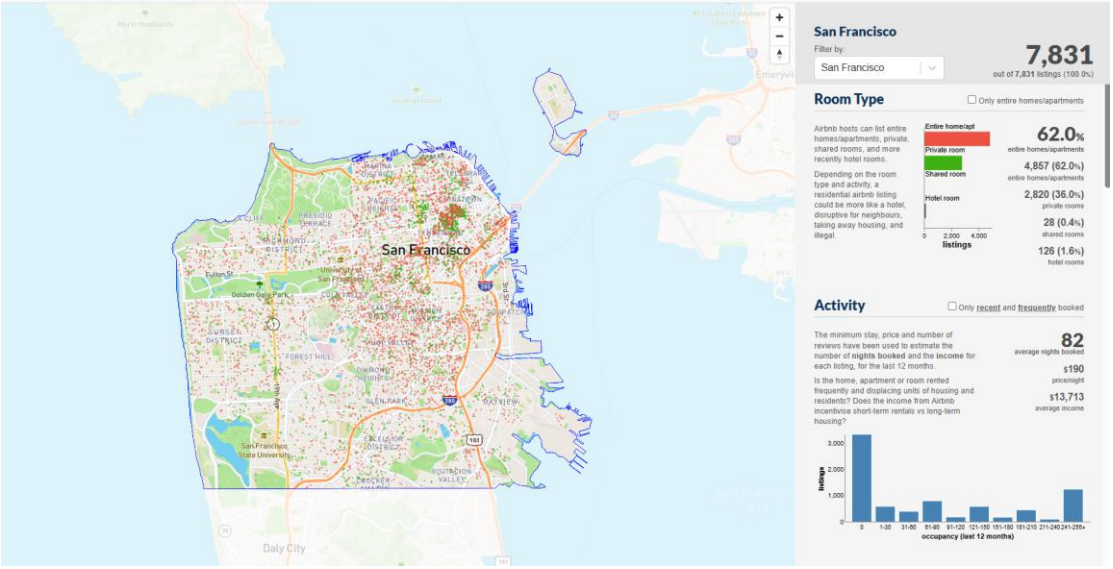
🏠 מיני פרויקט: יועץ נדל"ן חכם בסן פרנסיסקו

שילוב של Machine Learning + Streamlit + Telegram Bot + LLM :



🎯 מטרה:

בנו מערכת שלמה שמקבלת פרטים על דירה להשכרה בסן פרנסיסקו ומבצעת:

- חיזוי מחיר השכרה ללילה.
- הסבר בשפה פשוטה למה המחיר הזה.
- הצגת המידע בדשבורד אינטראקטיבי (Streamlit)
- יצירת בוט טלגרם (או פלטפורמה אחרת) שמבצע את כל הפעולות דרך הודעות.

📁 שלב 1: הגדרת מבנה הפרויקט

יצירת תיקיות מסודרות:

- data/ שם יהיה קובץ ה-CSV
- scripts/ כל שלבי העבודה בפייתון.
- models/ המודל המאומן.
- utils/ פונקציות עזר (לא חייב).
- README.md ו־ requirements.txt לשימוש עתידי.

את המילון שמפענח את המשתנים ניתן למצוא כאן:

https://docs.google.com/spreadsheets/d/1iWCNJcSutYqpULSQHlNyGInUvHg2BoUGoNRIGa6Szc4/edit?gid=1322284596#gid=1322284596

🧹 שלב 2: ניקוי נתונים (Data Cleaning)

מטרות:

- הסרת עמודות לא רלוונטיות.
- טיפול בערכים חסרים (NaN)
- המרה של עמודת מחיר מפורמט טקסט למספר.
- סינון קיצונים לא הגיוניים במחיר.

משימות לביצוע:

1. לטעון את קובץ הנתונים המקורי.
2. לבחור רק את העמודות החשובות למודל: מיקום, סוג חדר, מספר חדרים, אמבטיות, מחיר וכו.'
3. לבדוק האם בעמודת המחיר יש סימני $ ולהסיר אותם.
4. להמיר את המחיר לערך מספרי.
5. להסיר שורות עם ערכים חסרים בעמודות קריטיות.
6. לסנן שורות עם מחירים חריגים (למשל מעל $1000 או מתחת ל־$40).
7. למלא ערכים חסרים בעמודות משניות עם ממוצע או חציון.
8. לשמור את הקובץ כ־cleaned_data.csv

🧠 שלב 3: הנדסת מאפיינים (Feature Engineering)

מטרות:

- להפוך עמודות טקסט לייצוג מספרי.
- להכין את הנתונים לקראת מודל ML

משימות לביצוע:

1. לבדוק אילו עמודות הן קטגוריאליות (כמו סוג החדר).
2. לבצע קידוד One-Hot לאותן עמודות.
3. לוודא שאין בעיות של ערכים חסרים לאחר ההמרה.
4. לשמור את הקובץ כ־ featured_data.csv

📊 שלב 4: חלוקה לאימון ובדיקה

מטרות:

- לחלק את הנתונים ל־ Train ו־ Test
- להכין את סט הנתונים למודל.

משימות לביצוע:

1. להפריד בין משתני הקלט (features) למשתנה המטרה (price)
2. להשתמש בפונקציית train_test_split ולבחור למשל 80% לאימון, 20% לבדיקה.
3. לשמור את הסטים אם צריך להערכה מאוחרת.

🤖 שלב 5: בניית מודל ML

מטרות:

- לאמן מודל לחיזוי מחיר.
- לבחור אלגוריתם ולכוון פרמטרים בסיסיים.

משימות לביצוע:

1. לבחור אחד מהמודלים הבאים: CatBoost / XGBoost / LightGBM.
2. לאמן את המודל על סט האימון.
3. לשמור את המודל המאומן כ־ pkl. (לא חובה, מומלץ לקרוא על זה)
4. למדוד ביצועים על סט הבדיקה (לא שלכוח לעשות Cross Validation):
   - MAE
   - R^2
5. לרשום הסבר על תוצאות ההערכה: איפה הוא טועה, האם המחיר מנופח מדי?

🌐 שלב 6: בניית ממשק Streamlit

מטרות:

- לאפשר למשתמש להזין פרטי נכס.
- להציג את המחיר הצפוי.

משימות לביצוע:

1. ליצור טופס קלט:
   - בהתאם למאפיינים שבחרתם במודל.
2. להמיר את הקלט לפורמט שהמודל מבין.
3. להשתמש במודל ולחזות את המחיר.

4. להציג את המחיר בצורה ברורה.
5. להוסיף גרפים או מפה להצגת התפלגות מחירים.
6. לבדוק מה קורה אם מכניסים ערכים לא תקינים.

💬 שלב 7: בוט טלגרם

מטרות:

- לאפשר שליחה של פרטי נכס כטקסט.
- להחזיר חיזוי מחיר + הסבר.

משימות לביצוע:

1. ליצור בוט טלגרם דרך BotFather
2. לכתוב קוד שמאזין להודעות.
3. לפענח את פרטי ההודעה: כמות חדרים, סוג החדר וכו.'
4. לבצע חיזוי מחיר עם המודל.
5. לשלוח תשובה עם המחיר.
6. לשלב מודל LLM כמו OpenAI כדי להסביר למה המחיר כזה.

🧠 שלב 8: שילוב LLM להסברים

מטרות:

- להוסיף רובד של ניתוח בשפה טבעית.

משימות לביצוע:

1. לשלוח בקשה ל־ LLM עם המחיר והקלט המקורי.
2. לבקש הסבר פשוט על המחיר ("למה דווקא ככה?").
3. להחזיר את הטקסט הזה בטלגרם או בדשבורד.
4. בנו דאטהבייס שמשמר את ה- prompt והתשובה ל-prompt המתקבל מהמודל.

🧪 שלב 9: בדיקות קצה

מה לבדוק:

- מה קורה אם יש קלט שגוי?
- איך נראית תשובה לדירה יקרה מאוד או זולה מאוד?
- האם אפשר להשתמש באותו מודל גם לדירות בלי אמבטיה?

📜 **שלב 10: תיעוד והעלאה ל-Github**

משימות:

1. לכתוב README שמסביר:
   - מה הפרויקט עושה.
   - איך להריץ כל סקריפט.
   - דוגמאות לקלט ופלט.
2. לכתוב מסמך קצר: אילו בעיות נתקלתם בהן? מה למדתם?