



חוברת פרויקט גמר

קבוצה 46

ניתוח ומידול הקצאת כוננים אופטימלית בעמותת
'ידידים'

ידידים

סיוע בדרכים

מנחה אקדמאי : ד"ר ניר שוולב

מנחה תעשייתי : משה שלו

צוות מבצע :

סלין מירילשוילי 207277526

ליאור בירנדורף 208582114

שרון סייג 313194656



תוכן עניינים:

1.	תקציר מנהלים	3
2.	תיאור הארגון	3
2.1	המבנה הארגוני	4
2.2	תיאור סביבת הארגון	5
3.	הגדרת הבעיה	5
4.	מטרות הפרויקט	5
4.1	מטרת העל	5
4.2	מטרה משנית	6
4.3	חשיבות הפרויקט	6
4.4	מדדי הצלחה	6
5.	תרשים גאנט	7
6.	סקר ספרות	7
6.1	מבוא	7
6.2	מאמר 1	7
6.3	מאמר 2	9
6.4	מאמר 3	11
6.5	מאמר 4	13
6.6	מאמר 5	15
6.7	מאמר 6	17
6.8	סיכום סקר ספרות	18
7.	תיאור הנדסי של המצב הקיים	23
7.1	תיאור המצב הקיים	23
7.2	חקר מדדים הקדמה	24
8.	מתודולוגיה	34
8.1	מבוא למתודולוגיה	34
8.2	השגת הנתונים	34
8.3	עיבוד הנתונים	34
8.4	הכנה לבניית האלגוריתם ההונגרי	43
8.5	בניית האלגוריתם ההונגרי	46
9.	הצגת חלופות	49
9.1	טבלה מסכמת	52
10.	מימוש פתרון	54
10.1	הצגת פתרון	54
10.2	הממשק והליך העבודה	54
10.3	אילוצים	56
11.	הערכת הפתרון	57
12.	דיון ומסקנות	59
12.1	ניתוח הממצאים	59
12.2	המלצות לארגון	59
12.3	תובנות ולקחים	59
12.4	השלכות ותחזיות לעתיד	60
12.5	תרומה של כל חבר צוות	60
13.	נספחים	63



1. תקציר מנהלים

הפרויקט הנוכחי נערך בשיתוף עם עמותת "ידידים", אשר מטרתה לספק עזרה ראשונה שאינה רפואית לאזרחים בכל רחבי הארץ. לאור הגידול המשמעותי בפעילות העמותה בשנים האחרונות, התגלה צורך בשיפור משמעותי בתהליך הקצאת המתנדבים לקריאות הסיוע, שמבוצע כיום באופן ידני וללא כלים אוטומטיים מתקדמים. תהליך זה מוביל לעיתים לזמני תגובה ארוכים ולהתאמה לא אופטימלית בין המתנדבים לקריאות, מה שמפחית את יעילות התפעול.

לשם כך, פיתחנו מערכת אוטומטית המבוססת על אלגוריתם ההונגרי, שתפקידה לבצע אופטימיזציה בתהליך השיבוץ של המתנדבים. המערכת החדשה שואפת לייעל את תהליך השיבוץ על ידי שקלול פרמטרים קריטיים כמו זמינות המתנדב, קרבה גיאוגרפית, ניסיון קודם ודירוגים היסטוריים, במטרה לקצר את זמני התגובה, לשפר את אחוזי ההצלחה של הטיפול בקריאות, ולהגביר את שביעות הרצון של המתנדבים ומבקשי הסיוע.

בתחילת הפרויקט, נאספו נתונים היסטוריים על פעולות המתנדבים והקריאות שהתקבלו בארגון, אשר עברו עיבוד ושמידה על פרטיות המשתמשים. האלגוריתם ההונגרי שולב במערכת והוטמע לצורך חישוב השיבוץ האופטימלי, תוך פיתוח ממשק משתמש אינטואיטיבי באמצעות Streamlit, המאפשר הזנת נתונים, ביצוע חישובים בזמן אמת, והצגת התוצאות בצורה ברורה ומסודרת.

לצורך הערכת השיפור, ביצענו השוואה בין ביצועי המערכת החדשה לתהליך השיבוץ הידני, באמצעות מדדים כמותיים ואיכותניים. המדדים הכמותיים כללו את זמן הטיפול הממוצע, אחוזי הצלחה באירועים, בעוד המדדים האיכותניים התמקדו בשביעות רצון הלקוחות והמתנדבים. לצורך הערכת איכות האלגוריתם, השתמשנו במבחנים סטטיסטיים כגון T-test ו-ANOVA מפני ש T-test מתאים להשוואת ממוצעים בין שתי קבוצות בלבד, מה שאפשר לנו לבדוק האם חל שינוי משמעותי בזמן הטיפול בעקבות השימוש באלגוריתם. בעוד ש-ANOVA מאפשרת להשוות בין מספר קבוצות או משתנים שונים ולהבין את ההבדלים המשמעותיים ביניהם. כך, ניתן להבטיח שהשינויים שנעשו הם מבוססי ממצאים אמיתיים ולא תוצאה של סיכוי מקרי. תוצאות המבחנים הצביעו על הבדל משמעותי ומובהק סטטיסטית בין זמן הטיפול לפני ואחרי השימוש באלגוריתם, מה שמעיד על השפעה חיובית של האלגוריתם על זמן הטיפול.

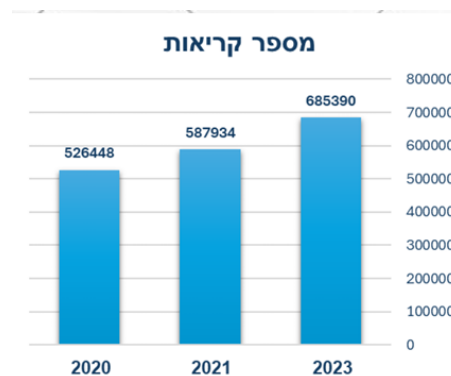
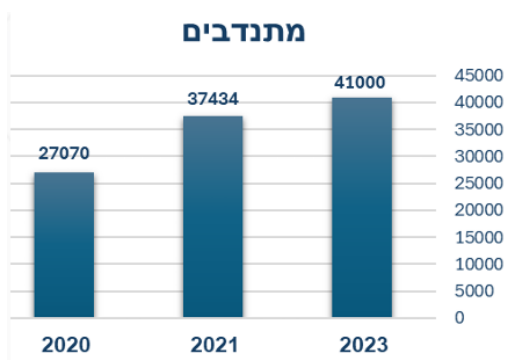
המערכת החדשה צפויה לאפשר לעמותת "ידידים" לנהל את המתנדבים בצורה מיטבית, להפחית את זמני התגובה ולשפר את חוויית השירות לכלל המעורבים בתהליך. הפרויקט מספק תשתית יציבה להמשך פיתוח ויישום כלים נוספים בעתיד, ומציג גישה חדשנית לשימוש במודלים אופטימיזציה בארגונים ללא מטרות רווח, תוך מתן מענה לאתגרים הייחודיים של ניהול מתנדבים ושיבוץ משימות.

2. תיאור הארגון-

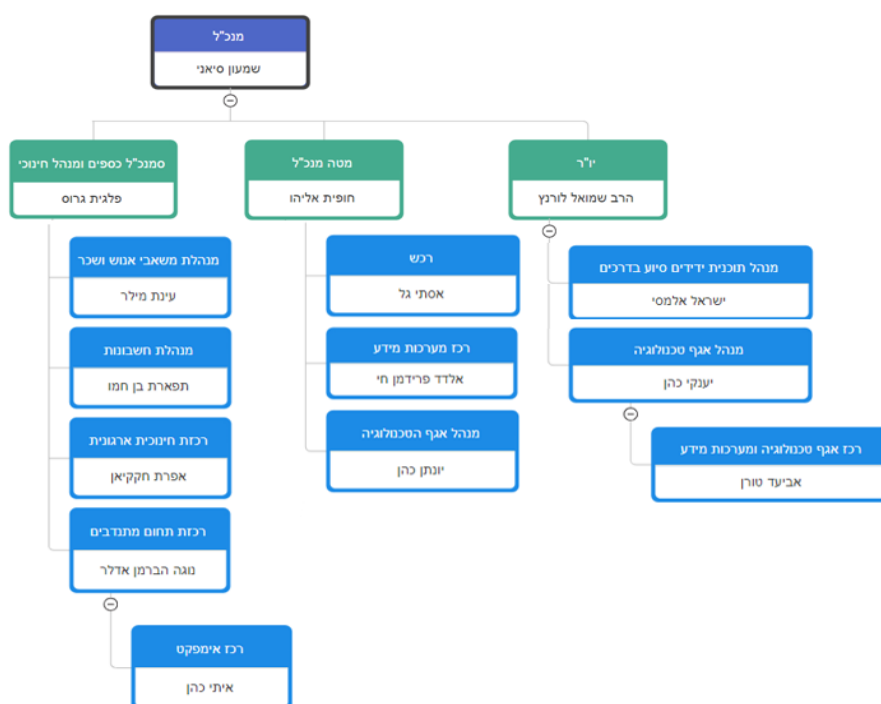
עמותת "ידידים - סיוע בדרכים" (ארגון ללא מטרות רווח) הוקמה בשנת 2006 ע"י מאיר ויינר. עמותה זו הינה עמותה ישראלית הפועלת למתן "עזרה ראשונה שאינה רפואית" בעבור נהגים ותושבים במצבי חירום בדרכים. העמותה נוסדה במטרה לסייע למי שנתקעו עם רכבם, או נזקקים לעזרה דחופה אחרת.

שירותי העמותה כוללים מגוון פעולות כמו סיוע בהתנעת רכבים, החלפת צמיגים, פתיחת רכבים נעולים ועוד.

ידידים פועלת כיום באמצעות רשת רחבה של למעלה מ-40,000 מתנדבים (נכון לשנת 2023) בפריסה ארצית, הזמינים לקריאות סיוע בכל שעות היממה (למעט שבתות וחגים). המתנדבים של העמותה מגיעים מכל חלקי החברה הישראלית ופועלים מתוך תחושת שליחות ורצון לעזור לזולת. העמותה מתבססת על תרומות בלבד ואינה גובה תשלום עבור השירותים שהיא מעניקה, כאשר כל פעילותה נשענת על רוח ההתנדבות של חבריה ועל התמיכה הציבורית שהיא מקבלת. "ידידים - סיוע בדרכים" חרתה בדגלה את ערכי הנתינה, התנדבות, סולידריות ואחריות חברתית ושמה לה למטרה להיות זמינה ויעילה ככל האפשר, על מנת לספק סיוע מהיר ואיכותי לכל מי שזקוק לכך, כל זאת על מנת לשפר את החברה הישראלית ככל הניתן במסגרת תפקידה ומטרתה.



2.1 המבנה הארגוני-





2.2 תיאור סביבת הארגון –

בכדי לאפשר לעמותה להנגיש את הסיוע שלה לציבור הרחב, הוקצה מס' טלפון מקוצר ע"י משרד התקשורת, בדומה לגופי החירום הנוספים. לעמותה היום יש מוקד טלפוני המאויש גם הוא ע"י מתנדבים שמטרתו לרכז את קריאות הבקשה לסיוע השונות שנכנסות בכל רגע נתון. הדרכים בהן הארגון מאפשר למבקשי העזרה לפתוח בקריאה חדשה, הן באמצעות שיחת טלפון למוקד או פתיחת בקשה דרך אפליקציה ייעודית. בעת קבלת קריאה חדשה למערכת, יוצאת קריאה באפליקציית הכוננים והראשון לאשר את הקריאה, הוא זה שיצא לסייע. מוקד החירום של העמותה עושה שימוש במערכת CiviCRM שהיא מערכת CRM חינוכית מקודדת פתוחה שמספקת פתרונות לניהול מתנדבים, תרומות וארגון אירועים. מאחר ומדובר בארגון ללא מטרות רווח, נעשים מאמצים רבים בשימוש בתוכנות חינוכיות כאלה ואחרות וזאת אחת הסיבות לשימוש בתוכנה זו בעמותה. למערכת זו מוזנים נתוני הקריאה (מיקום תקלה, פרטי פותח הקריאה, סוג תקלה וזמן פתיחה) ע"י פותח הקריאה בעזרת אחד התהליכים שהוצגו לעיל. בסיום טיפול התקלה, נתונים נוספים מצטרפים לשורת התקלה (שעת הגעה לתקלה, פרטי כונן וסטטוס גמר טיפול) ע"י הכונן שהגיע לסייע בפתרון התקלה דרך אפליקציית הכוננים. בפרויקט מחקר זה, עבדנו על נתונים אשר נשלפו מתוך ארכיון מערכת העמותה, והוצגו לנו בעזרת טבלת Excel ולא על המערכת באופן ישיר, מתוך חשש מפגיעת הפרטיות הן של הכוננים והן של מבקשי הסיוע. הנתונים עליהם נעבוד במסגרת פרויקט זה, עברו תהליך התממה ע"י העמותה הכולל הסתרת מספרי תעודות זהות, מספרי טלפונים, שינוי תצוגת מזהה כונן ומזהה פותח הקריאה ועוד.. בפרויקט זה, נעשה שימוש בסביבת עבודה Colab לצורך עיבוד, ניתוח הנתונים ובניית האלגוריתם ההונגרי בשפת Python.

3. הגדרת הבעיה

על אף הערכה הרבה מצד האוכלוסיה וגורמים מדיניים כאלה ואחרים, אין ביכולת העמותה לייעל את תהליך הקצאת הכוננים לקריאות השונות שמתקבלות. נכון להיום, כאשר מתקבלת קריאה חדשה לסיוע בדרכים המקובלות, לא מבוצע הליך תכליתי של סיווג התקלה והתאמת המתנדב האופטימלי בעל הכישורים הנדרשים לטיפול בה. התהליך הקיים מאפשר לכל מתנדב שמעוניין לסייע, לקחת חלק בטיפול בקריאה החדשה. התוצאה היא תהליך מסורבל שלעיתים מאריך את זמני התגובה אשר פוגע ביכולת העמותה לסייע במהירות וביעילות למבקשי העזרה. בנוסף, תהליך הקצאת הכונן לתקלה חדשה מבוסס על חוקים פשוטים ואינו מצליח לקחת בחשבון את כלל הפרמטרים הרלוונטיים, כגון מיקום המתנדבים, זמינותם, הצלחתם בטיפולים קודמים ופרמטרים נוספים המשפיעים על היכולת לתת מענה מיטבי לקריאה. הבעיה המרכזית היא הצורך בשיפור תהליך ההקצאה, כך שהמתנדב המתאים ביותר ייבחר באופן אוטומטי, על בסיס קריטריונים מוגדרים ובאופן שיבטיח מענה מהיר ויעיל לכל קריאה.

4. מטרות הפרויקט

4.1 מטרת העל-

המטרה הראשית בפרויקט זה היא פיתוח מערכת אופטימיזציה אוטומטית שתוביל לשיפור משמעותי במשימתה העיקרית של העמותה - "עזרה ראשונה שאינה רפואית". מערכת זו תשפר את הליך הקצאת הכונן לקריאה חדשה המתקבלת במערכת העמותה תוך מתן דגש על פתרון שיביא ליעול משמעותי בפרמטרים מהותיים שהוגדרו לנו מטעם מנהלי העמותה- קיצור זמני ההמתנה, זמן טיפול בתקלה וכן לשיפור בחווית הכוננים ומבקשי העזרה. (הרחבה אודות פרמטרים אלו - בסעיף מדדים). בהתחשב במיקוד העמותה בשירות מהיר ואיכותי, המערכת תאפשר שימוש אופטימלי בכוננים הזמינים ותייעל את ניצול המשאבים הקיימים, תוך הפחתת זמן ההמתנה וטיפול מהיר ויעיל יותר בקריאות. מערכת זו תתבסס על האלגוריתם ההונגרי ותציג באופן ויזואלי את הכונן האופטימלי לטיפול בכל תקלה תוך התחשבות בפרמטרים המהותיים לעמותה. הפרויקט יתמקד בפיתוח מערכת שמבוססת על נתונים היסטוריים ותוכל להפיק החלטות בזמן אמת, כך שתוכל להקצות את הכונן המתאים לכל קריאה חדשה.



4.2 מטרה משנית - ממטרת העל של פרויקט זה נגזרות מספר מטרות משניות אותן נרצה ונצטרך להשיג-

- עלייה במספר משתמשי אפליקציית מבקשי הסיוע.
- שיפור חווית הכוננים.

4.3 חשיבות הפרויקט-

לאור הצמיחה המרשימה של העמותה בשנים האחרונות, אנחנו סבורות כי הפרויקט הינו רלוונטי ויביא תועלת משמעותית לצרכי העמותה. באמצעות תהליך הקצאת כווננים נכון ויעיל, נקנה לעמותה את היכולת לתת שירות יעיל יותר לציבור ואף לגרום לשיעור רצון גבוהה יותר הן בחווית מבקשי העזרה והן בחווית מתנדבי העמותה.

הטמעת האלגוריתם אותו אנו בונות בפרויקט זה במערכות העמותה, תסייע לאחרונה להגדיל את היקף הסיוע שלה שכאמור זהו חזון העמותה: "החזון שלנו הוא להגיע למצב בו יהיה לפחות מתנדב אחד בכל בניין, כך נוכל להגיע למצב בו לא יתקע אף אדם למעלה ממספר דקות בכל מקום ברחבי הארץ. כך נהפוך את מדינת ישראל למקום טוב יותר." [1]. כל זאת בעזרת פישוט נכון וממוקד של התהליך הקיים היום בעמותה.

4.4 מדדי הצלחה-

במסגרת הפרויקט, הוגדרו מספר מדדים מרכזיים שמטרתם לאמוד את השיפור באפקטיביות וביעילות התפעולית של הארגון בעקבות ההטמעת המערכת. הצלחת הפרויקט תלויה במידה רבה במדדים כמותיים ואיכותניים, שנבחרו בקפידה לאחר דיון עם הגורמים הרלוונטיים בארגון. מדדים אלו משקפים את ההתקדמות הרצויה וההשפעה המיוחלת, תוך התייחסות ליעילות, שביעות רצון הן של מבקשי הסיוע והן של הכוננים, והיכולת של הארגון להגיב לאירועים אלו במהירות וביעילות. להלן היעדים:

- זמן טיפול ממוצע: נצפה לקיצור זמן הטיפול הממוצע ל-18 דקות או פחות, מה שמהווה שיפור של 25% מהמצב הנוכחי. צמצום זמן הטיפול יאפשר לארגון להתמודד עם יותר אירועים בזמן קצר יותר, ובכך לשפר את שביעות רצונם של מבקשי הסיוע.
- אחוז הצלחה באירועים: נשאף לעלייה של 10% באחוז האירועים שנסגרו בהצלחה, מה שיביא את אחוזי ההצלחה הכלליים ל-90%. שיפור זה יאפשר לארגון לספק שירות יעיל ואיכותי יותר, ולהשיג דירוגים גבוהים יותר מצד הלקוחות.
- ניצול משאבים וזמינות מתנדבים: נצפה לעלייה ניכרת בזמינות המתנדבים באמצעות אופטימיזציה של הקצאתם למשימות, כך שגם בתקופות של עומס קריאות, תוכל להבטיח מתן שירות ברמה גבוהה ובזמן תגובה קצר.
- שביעות רצון מבקשי הסיוע: שיפור שביעות הפונים הוא יעד מרכזי. נצפה לעלייה של 15% בדירוגי שביעות הרצון, עם יעד להגיע לציון ממוצע של 4 מתוך 5 או יותר בכל הקטגוריות. שיפור זה יתבסס על קיצור זמני הטיפול, העלאת איכות השירות, ושיפור התקשורת עם מבקשי הסיוע לאורך כל תהליך הסיוע.
- שביעות רצון המתנדבים: נדגיש את חשיבות שיפור שביעות רצון המתנדבים, אשר יתבטא בהפחתת העומס על המתנדבים, מתן תמיכה בזמן אמת, והיכולת להשתתף באופן פעיל בבחירת המשימות שהם מבצעים. נשאף לעלייה בשביעות הרצון של המתנדבים, מה שישפר את תחושת המוטיבציה והנכונות להמשיך ולהתנדב.



5. תרשים גאנט

משימות	ספטמבר	אוגוסט	יולי	יוני	מאי	ינואר-אפריל
המתנה לקבלת הנתונים						
סיעור מוחות וקבלת הנתונים						
מיפוי דרישות ויצירת KPIs						
ניקוי וסידור הדאטה						
עיבוד הנתונים ובניית המודל						
כתיבת מפרט טכני						
שיפורים אחרונים וסיום הפרויקט						

6. סקר ספרות

6.1 מבוא

בסיסי הידע המדעי והיישומי הם המנוע המרכזי המניע את הפרויקט הנוכחי לפיתוח ויישום מערכת אופטימיזציה להקצאת משאבים ולניהול תהליכים בארגון "ידידים". במסגרת סקירת הספרות, נבחן מחקרים ומאמרים רלוונטיים המעניקים תובנות קריטיות, עקרונות ושיטות עבודה המיושמים במודלים דומים, תוך התמקדות בבעיות הקצאה, ניהול מתנדבים, ושימוש במודלים מתקדמים כמו האלגוריתם ההונגרי.

הסקירה תסייע בהבנת הקשיים והאתגרים המרכזיים בניהול אופטימלי של משאבי המתנדבים, ותציג פתרונות מבוססי נתונים לשיפור תהליכי עבודה והגברת היעילות התפעולית בארגון. כל מאמר שנבחר לסקירה נבחן בזכות התרומה המעשית שלו לשיפור מערכות הקצאה וניהול משאבים. הדגש הוא על המתודולוגיות השונות שנעשה בהן שימוש, התוצאות שהתקבלו, והלקחים שניתן ללמוד מהן, במטרה להבטיח שהפתרונות המוצעים יהיו מבוססים על שיטות עבודה מוכחות ויעילות, שנבדקו ונמצאו מתאימות ביותר להקשר הספציפי של הפרויקט.

6.2 מאמר 1

A bipartite graph matching approach for relevance scoring and combinatorial optimization

המאמר עוסק בפיתוח מערכת לשיבוץ אופטימלי של מסמכים למבקרים בכנסים. המאמר עושה שימוש בגישה של גרף דו-צדדי (Bipartite Graph) ובאלגוריתם ההונגרי תוך התחשבות בציון רלוונטיות של המסמכים לכל מבקר, זאת בכדי להבטיח את ההתאמה האופטימלית בין המסמכים למבקרים תחת מגבלות שונות.

המטרה העיקרית של המחקר היא ליצור מערכת שמבצעת התאמה אופטימלית בין מסמכים למבקרים, תוך התחשבות במגבלות כמותיות כמו מספר מינימלי של ביקורות לכל מסמך ומקסימום של מסמכים שכל מבקר יכול לסקור. הרקע למחקר נובע מהצורך להתמודד עם האתגר של שיבוץ מסמכים למבקרים בצורה שתמקסם את הרלוונטיות והאפקטיביות של השיבוצים.

מתודולוגיה:

גרף דו-צדדי (Bipartite Graph) מדובר במבנה שבו יש שני סטים של צמתים (Nodes), כאשר קשתות (Edges) מחברות בין הצמתים של הסטים השונים, אך לא בתוך אותו סט.



במאמר זה, הצמתים מייצגים את המסמכים ואת המבקרים. הקשתות מייצגות את הקשרים האפשריים בין מסמכים למבקרים, עם משקלים שמייצגים את מידת הרלוונטיות או עלות ההתאמה ביניהם. במאמר, כל מסמך מיוצג כצומת בקבוצה אחת, וכל מבקר מיוצג כצומת בקבוצה השנייה. קשתות מחברות בין כל מסמך למבקר פוטנציאלי, והמשקלים של הקשתות מחושבים לפי מידת הרלוונטיות של המסמך עבור המבקר (למשל, באמצעות ציון TF-IDF).

חישוב ציוני TF-IDF

$\frac{t,df}{\sum_{k,df} k} = TF(t, d)$ TF-IDF (Term Frequency-Inverse Document Frequency) הוא מדד סטטיסטי שמטרתו לשקף את החשיבות של מונח מסוים במסמך, יחסית לשאר המסמכים באוסף.

TF (תדירות): מספר הפעמים שהמונח מופיע במסמך. $\frac{|D|}{|\{d \in D: t \in d\}|} \log = IDF(t, D)$

IDF (תדירות מסמכים הפוכה): מדד הפוך לתדירות שבה המונח מופיע במסמכים אחרים באוסף.

* כאשר t הוא מונח, d הוא מסמך, ו-D הוא אוסף המסמכים

$$IDF(t, D) \times TF(t, d) = IDF(t, d, D) - TF$$

המאמר משתמש ב-TF-IDF כדי לחשב את הרלוונטיות של כל מסמך עבור כל מבקר. כך, אם מבקר מסוים מתמחה בנושא מסוים, המונחים הרלוונטיים לנושא הזה יקבלו משקל גבוה יותר בחישוב הרלוונטיות.

האלגוריתם ההונגרי - האלגוריתם ההונגרי הוא שיטה למציאת התאמה אופטימלית בבעיה של הקצאת משאבים, כמו בגרף דו-צדדי. האלגוריתם פועל על מטריצת עלויות שבה כל שורה מייצגת מסמך וכל עמודה מייצגת מבקר, והמטרה היא למזער את העלות הכוללת או למקסם את הרלוונטיות הכוללת.

שלבים באלגוריתם:

יצירת מטריצת עלויות: חישוב העלות או הרלוונטיות בין כל מסמך למבקר. נרמול המטריצה: ניכוי הערך המינימלי בכל שורה ועמודה. הקצאת אפסים: חיפוש אחר אפסים במטריצה ולהקצות אותם ללא התנגשות (כאשר כל מסמך מקבל רק מבקר אחד). התאמת המשקלים: אם לא ניתן להקצות את כל המשאבים, התאמה של המשקלים במטריצה כך שניתן יהיה לבצע הקצאה. חזרה על השלבים עד שמתקבלת התאמה אופטימלית.

התאמה במשקל מרבי (Maximum Weight Matching) - המטרה היא למצוא את השיבוץ בין שני הסטים (במקרה זה, מסמכים ומבקרים) כך שהמשקל הכולל של ההתאמות יהיה מרבי. כל התאמה בין מסמך למבקר נותנת ערך (רלוונטיות), והאלגוריתם מנסה למקסם את הסכום של כל הערכים הללו. לאחר חישוב הרלוונטיות באמצעות TF-IDF, האלגוריתם ההונגרי משמש למציאת ההתאמה המרבית שתמקסם את הרלוונטיות הכוללת של ההתאמות, תחת המגבלות שנקבעו.

תוצאות המחקר:

המחקר יושם על נתוני אמת מועידה טכנולוגית, והמערכת שהוצעה הוכיחה את עצמה כפתרון מבטיח לשיבוץ אוטומטי של מסמכים למבקרים. התוצאות הראו שהמערכת מצליחה לשמור על שביעות רצון גבוהה של המבקרים תוך עמידה במגבלות כמותיות שהוגדרו מראש. האלגוריתם ההונגרי, ששולב במערכת, מבטיח התאמה אופטימלית בין מסמכים למבקרים, במיוחד בתנאים של מגבלות כמו מספר מינימלי של ביקורות לכל מסמך ומקסימום של מסמכים שכל מבקר יכול לשפוט. השימוש במטריצת TF-IDF מאפשר לקבוע את הרלוונטיות של כל מסמך עבור כל מבקר ולבצע התאמה מיטבית. המודל נבדק והודגם על נתוני אמת, מה שהוכיח את יכולת המערכת לשפר את ההתאמה בין מסמכים למבקרים ולהוביל לשביעות רצון גבוהה יותר. תוצאות המחקר מאשרות את החשיבות של מודלים מתמטיים מתקדמים לצורך אופטימיזציה של תהליכי שיבוץ והתאמה בפרויקטים דומים.



מסקנות :

המסקנות המרכזיות מצביעות על היעילות הגבוהה של האלגוריתם ההונגרי בשילוב זה מטריצת TF-IDF בתהליך ההתאמה. שילוב זה הוביל לשיפור ניכר בהתאמת המשאבים, כמו גם לאוטומציה מוצלחת של תהליך השיבוץ. המערכת לא רק שמפחיתה את העומס על מארגני הכנסים, אלא גם מבטיחה שהשיבוצים יתבצעו בצורה חלקה ואופטימלית. המאמר מצביע על כך שהשיטות המוצעות בו יכולות לשמש גם בתחומים אחרים שבהם יש צורך בהתאמה אופטימלית בין שני סטים של פריטים, כמו בפרויקט שלנו של התאמת מתנדבים למשימות, תוך הבטחת התאמה אופטימלית על פי קריטריונים מוגדרים מראש.

התרומה של המאמר לפרויקט : המאמר מספק תשתית תאורטית לפיתוח מודל התאמה בין מתנדבים למשימות בפרויקט שלנו, תוך שימוש באלגוריתם ההונגרי ובחישוב ציון רלוונטיות מבוסס TF-IDF. יישום שיטות אלו צפוי לשפר את דיוק השיבוץ בתנאים של מגבלות זמן ומשאבים מוגבלים, ולהבטיח התאמה אופטימלית של מתנדבים למשימות בזמן אמת, בהתבסס על קריטריונים כגון מיקום גיאוגרפי וזמינות.

6.3 מאמר 2

Distributed Matching-By-Clone Hungarian-Based Algorithm for Task Allocation of Multiagent Systems

המאמר עוסק בפיתוח שיטה מתקדמת להתאמת משימות במערכות מרובות סוכנים, כאשר מספר הסוכנים קטן ממספר המשימות. המאמר מציע לשלב את האלגוריתם ההונגרי עם טכניקות של שיבוט סוכנים והוספת משימות דמה (*Dummy Tasks*) כדי להבטיח שהמטריצה המשמשת לאופטימיזציה תהיה ריבועית, וכך ניתן יהיה לבצע הקצאה אופטימלית של משימות לסוכנים.

מטרת המאמר היא להבטיח חלוקת משימות אופטימלית תוך מזעור העלויות הכוללות ושיפור זמן ההתכנסות, באמצעות גישה חדשנית לפתרון בעיית ההקצאה.

מתודולוגיה :

האלגוריתם *DMCHBA* מיועד להקצות משימות לסוכנים במערכת מרובת סוכנים, תוך שימוש במטריצת עלויות ריבועית להשגת אופטימיזציה.

נוסחת האלגוריתם 3 במאמר מתארת את השלבים המדויקים של התהליך שבו כל סוכן מבצע את ההקצאה בהתאם למספר המשימות והסוכנים.

Algorithm 3: DMCHBA Phase II: Assignment.

For agent k at time τ ,

1: **Procedure:** (A)ssign_task($N_t, N_a, q_k(\tau), a$)

2: **If** $N_a < N_t$

3: $r = \left\lceil \frac{N_t}{N_a} \right\rceil$. % Operation $\lceil \star \rceil$ rounds \star to the nearest integer greater than or equal to \star .

4: $n = r \cdot N_a$.

5: Add $(r - 1)$ cloned agent sets.

6: Add $(n - N_t)$ pseudo tasks.

7: **Else If** $N_a = N_t$

8: $n = N_a$.

9: **End If**

N_a : מספר הסוכנים הזמינים

N_t : מספר המשימות שיש להקצות

$q_k(\tau)$: הפרמטרים של הסוכן k בזמן τ

$S_{a,n}$: קבוצת האינדקסים של סוכנים ומשימות שנוצרו

$C^k(\tau)$: מטריצת העלויות של הסוכן k בזמן τ

שלב התקשורת : בשלב ההתחלתי, הסוכנים מתקשרים עם שכנותיהם כדי להחליף מידע עד להשגת בסיס ידע גלובלי. התקשורת מתבצעת באופן מבוזר, כך שכל סוכן מעדכן את המידע המקומי שלו בלבד. לאחר ששלב התקשורת מסתיים, כל סוכן יוצר מטריצת עלויות עבור המשימות שהוא מקבל :

מתבצעת בדיקה האם $N_a < N_t$ אם כן, מחשבים את r כשארית חלוקת המשימות בסוכנים ומוסיפים סוכנים משובטים/משימות דמה כדי ליצור מטריצת עלויות ריבועית.

שלב ההקצאה (Assignment Phase): לאחר הוספת משימות הדמה, כל סוכן בונה מטריצת עלויות ריבועית $(n \times n)$ הכוללת את כל הסוכנים והמשימות, על ידי הוספת משימות דמה (dummy tasks) או שיבוט סוכנים (cloned agents), המשמשת כבסיס להפעלת האלגוריתם ההונגרי להקצאת משימות.

הפעלת האלגוריתם ההונגרי: מטריצת העלויות מועברת לאלגוריתם ההונגרי שמבצע את ההתאמה האופטימלית בין הסוכנים למשימות, תוך התחשבות במטרות כמו מזעור עלויות ומקסום רלוונטיות.

(X^k, t) : מטריצת ההקצאה הבינארית הנוצרת על ידי האלגוריתם ההונגרי. כשלאחר מכן, כל סוכן משתמש באלגוריתם תכנון מקומי (כגון TSP) כדי לקבוע את סדר ביצוע המשימות.

איור 3: השוואת ביצועי האלגוריתמים הגרפים המוצגים באיור מציגים את השוואת זמני ההתכנסות של מספר אלגוריתמים בתהליך הקצאת משימות, כאשר מספר המשימות והסוכנים גדל. האלגוריתמים המוצגים (DMCHBA-HLPPA ו-DMCHBA-NLPPA), המיוצגים בקווים הירוקים והורודים, מראים שיפור ניכר בזמן ההתכנסות בהשוואה לאלגוריתם המסורתי (CBBA), המסומן בקו הכחול.

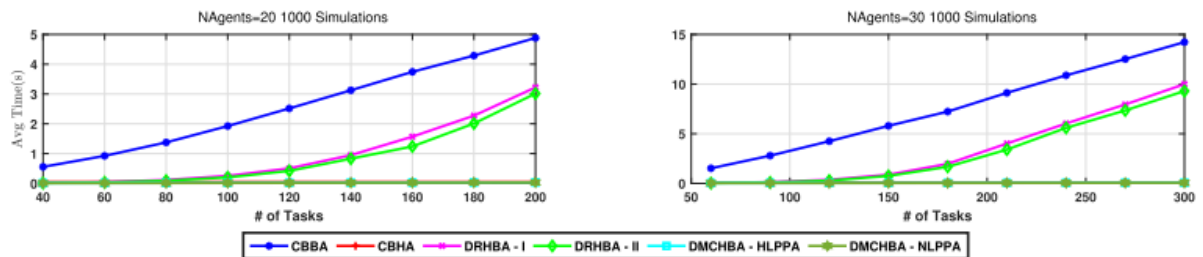


Fig. 3. Averaged computational/converging time of 1000 simulations for the four selected and two proposed algorithms..

הגרפים ממחישים את היתרון הברור של האלגוריתמים המוצגים במצבים מורכבים, שבהם מספר המשימות עולה על מספר הסוכנים. ניתן להבחין כי ככל שמספר המשימות גדל, היתרון של האלגוריתמים המוצגים בולט עוד יותר, עם שיפוע פחות תלול שמעיד על יציבות ויעילות גבוהה יותר. ממצאים אלו מצביעים על כך שהשימוש ב-DMCHBA משפר משמעותית את הביצועים, במיוחד במצבים של מספר רב של משימות, דבר הרלוונטי במיוחד להקצאת משאבים בפרויקט הנוכחי.

תוצאות המחקר: המחקר מראה כי השימוש במשימות דמה וסוכנים משוכפלים בשילוב עם האלגוריתם ההונגרי מוכיח את עצמו כפתרון אופטימלי. השיטה המוצעת, מציגה ביצועים טובים יותר בהשוואה לאלגוריתמים מסורתיים ומצליחה להפחית את הזמן הכולל הנדרש לביצוע המשימות ולהקטין את העלות הכוללת, מה שמדגיש את יתרונם במצבים מורכבים ומבוזרים.

מסקנות: ניכרת החשיבות של יצירת מטריצת עלויות ריבועית, במיוחד במצבים בהם קיים חוסר איזון בין מספר המשימות למספר המתנדבים. היישום של השיטה עשוי להיות קריטי לשיפור האפקטיביות של האלגוריתם ההונגרי שכבר מיושם בפרויקט שלנו. השיטות מאפשרות להתמודד עם בעיות הקצאה לא סימטריות, והשוואת ביצועים על בסיס מדדים מוגדרים. הערך המוסף שהמאמר מביא יכול לעזור לשפר את דיוק השיבוץ ולהפחית את עלויות התפעול, מה שיביא לשיפור משמעותי ביעילות המערכת.

התרומה של המאמר לפרויקט: בפרויקט שלנו צורך בשיבוץ מתנדבים למשימות באופן אופטימלי, המאמר מספק תמיכה תיאורטית למהלך של הוספת משימות דמה כדי להפוך את מטריצת העלויות לריבועית. יישום השיטה המוצעת מהווה צעד חיוני הצפוי לשפר את הדיוק



והיעילות של האלגוריתם ההונגרי המיושם בפרויקט, ולספק פתרונות אופטימליים בהתאמה בין מתנדבים למשימות גם במצבים מורכבים.

6.4 מאמר 3

New Task Oriented Recommendation method Based on Hungarian algorithm in Crowdsourcing Platform

המאמר מציג מחקר חדשני העוסק בפיתוח שיטה להמלצה על משימות חדשות בפלטפורמות קראודסורסינג, בהן משתמשים מבוזרים מתבקשים לבצע משימות באופן חופשי ובהתנדבות.

המחקר מתמודד עם בעיה מרכזית בתחום זה, שהיא התאמת המשימות המתפרסמות למשתמשים המתאימים ביותר לביצוען. לשם כך, המחקר מציע שימוש באלגוריתם ההונגרי, אשר ידוע ביכולתו לפתור בעיות השמה בצורה אופטימלית, ופיתוח מודל אופטימיזציה מרובה מטרות שנועד לשפר את הדיוק והיעילות של תהליך ההמלצה.

המאמר דן באתגרים הקיימים בפלטפורמות קראודסורסינג, במיוחד בתחום התאמת המשימות למשתמשים, כאשר חוסר התאמה זה מוביל לאובדן זמן ומשאבים עבור המשתמשים והפלטפורמה כאחד. מטרת המאמר היא לשפר את היעילות של פלטפורמות קראודסורסינג על ידי התאמה מדויקת יותר של משימות חדשות למשתמשים, תוך שימוש בנתוני עבר ובמודל אופטימיזציה מרובה מטרות כדי למקסם את התאמות אלו בצורה מיטבית.

מתודולוגיה:

במאמר מוצגת מתודולוגיה הכוללת מספר שלבים מרכזיים: איסוף נתונים היסטוריים: איסוף נתונים ממשתמשים בפלטפורמת הקראודסורסינג על המשימות שהם ביצעו בעבר וכן איסוף מידע על העדפות המשתמשים, כגון סוגי המשימות שהם מעדיפים, רמת הקושי המועדפת, הזמן שהם מוכנים להשקיע ועוד.

נתונים אלה כוללים מידע על ביצועים, זמני תגובה, איכות הפתרונות, עלויות וכדומה.

לכל משתמש i ולכל משימה j מחושבת מטריצה יעילות עבור כל אחת משלושת המטרות המרכזיות:

חישוב מטריצת יעילות ($Efficiency Matrix$):

$k=1$ מייצג את מטרת האיכות.

$2=k$ מייצג את מטרת הזמן.

$3=k$ מייצג את מטרת העלות.

מטריצת היעילות מבוססת על מדדים כמותיים כמו מספר המשימות שהושלמו, איכות התוצרים, זמני הביצוע, ועלויות הפתרונות שהוגשו בעבר. כל מדד כמותי מנורמל כך שהערכים יהיו ברי השוואה בין המשתמשים והמשימות השונות.

חישוב דמיון בין משימות ($Task Similarity Calculation$): הדמיון בין משימות מחושב באמצעות גישות מבוססות טקסט, כמו $TF-IDF$, על תיאורי המשימות והכותרות שלהן. חישוב הדמיון בין משימות נועד לקבוע אילו משימות קודמות הן דומות למשימות חדשות שנמצאות במערכת.

השימוש במטריצת הדמיון מאפשר לחזות את היעילות של משתמשים במשימות חדשות על סמך הביצועים שלהם במשימות דומות בעבר.



בניית מודל אופטימיזציה מרובה מטרות: המודל מגדיר מטרות ומתמקד בשלוש מטרות עיקריות:
איכות $F_1(X)$, זמן $F_2(X)$ ועלות $F_3(X)$

עבור כל משתמש i ומשימה j המטריצה ijC משקללת את שלושת המטרות כך:

$${}_{ij}^k w {}_k C \sum_{k=1}^3 = {}_{ij} C$$

כאשר ${}^k w$ הם המשקלים של כל מטרה, הבעיה הופכת לבעיה של אופטימיזציה מרובה מטרות, הנפתרת באמצעות האלגוריתם ההונגרי לאחר שהבעיה מצומצמת לבעיה חד-מטרותית באמצעות שקילת המטרות.

הפעלת האלגוריתם ההונגרי: האלגוריתם ההונגרי משמש כדי למצוא את ההתאמה הטובה ביותר בין המשימות למשתמשים, תוך התחשבות בהעדפות המשתמשים ובמטרות המרובות. האלגוריתם מבצע אופטימיזציה של ההתאמה כך שתושג תוצאה מיטבית בכל המטרות שהוגדרו. האלגוריתם ההונגרי מופעל על מטריצת היעילות המשוקללת ijC , במקרים בהם מספר המשימות קטן ממספר המשתמשים, המודל מבצע הרחבה של מטריצת היעילות על ידי הוספת משימות דמה (*dummy tasks*) כדי להתאים את הבעיה לפורמט הנדרש.

תוצאות המחקר:

לאחר מציאת הפתרון האופטימלי, נבחנים ביצועי המערכת, והמודל נבדק בהשוואה לשיטות אחרות. התוצאות מראות שיפור משמעותי בדיוק ההמלצות וביעילות הביצוע של המשתמשים.

דיון ומסקנות:

החוקרים מצאו שהשיטה החדשה יכולה לשפר את חוויית המשתמשים בפלטפורמות מיקור המונים. המודל מציע פתרון יעיל לבעיית ההתאמה הנמוכה ומספק תוצאות טובות יותר מבחינת איכות, זמן ועלות. המחקר הראה שהשיטה המוצעת משפרת באופן משמעותי את דיוק ההמלצות על משימות חדשות למשתמשים. בטבלה 2 של המאמר מוצגים תוצאות דיוק ההמלצות עבור משימות חדשות בשיטות שונות.

התוצאות מראות שהשיטה המוצעת, המשלבת בין העדפות משתמשים ואופטימיזציה של מטרות מרובות, השיגה דיוק המלצות גבוה יותר בהשוואה לשיטות קודמות שהתמקדו בהעדפות משתמשים בלבד או באופטימיזציה של מטרה יחידה. לדוגמה, דיוק ההמלצות בשיטה המוצעת עמד על ערכים שבין 0.43 ל-0.58, בהשוואה לדיוק שנע בין 0.21 ל-0.46 בשיטות אחרות.

TABLE 2: The recommendations precision of new tasks under different methods

Serial number	User preference	User preferences and single goal efficiency	The method of this paper
1	0.34	0.40	0.58
2	0.21	0.31	0.43
3	0.24	0.29	0.48
4	0.20	0.33	0.50
5	0.38	0.46	0.57

שיפור זה הביא לשיפור היעילות התפעולית של הפלטפורמה, עם תוצאות טובות יותר מבחינת איכות הביצועים וזמני התגובה.

התרומה של המאמר לפרויקט: המאמר מציג מתודולוגיה מתקדמת להמלצה על משימות בפלטפורמות קראודסורסינג, המבוססת על אלגוריתם ההונגרי ומודל אופטימיזציה מרובה מטרות. המתודולוגיה המוצעת במאמר מתאימה במיוחד ללמידה וליישום בארגון "ידידים", שכן הבעיה המרכזית שאנו מתמודדים איתה דומה במהותה – הצורך לשבץ מתנדבים למשימות בצורה אופטימלית, תוך התחשבות במגוון רחב של פרמטרים כמו זמינות, מיקום גיאוגרפי, יכולות, וזמני תגובה.

המתודולוגיה הכמותית המוצגת במאמר, הכוללת חישוב מטריצות יעילות, שקילת מטרות מרובות והפעלת אלגוריתם ההונגרי, מספקת כלים פרקטיים שניתן ליישם בפרויקט שלנו. המאמר מדגים בצורה ברורה כיצד ניתן להשתמש בנתוני עבר ובחישובי דמיון כדי לשפר את הדיוק והיעילות של



התאמת המשימות למתנדבים, מה שיכול לסייע לנו לפתח מערכת שיבוץ יעילה ומותאמת אישית בארגון.

בנוסף, הגישה המתוארת במאמר מספקת תשתית תיאורטית מוצקה לשימוש באלגוריתם ההונגרי, כולל הרחבת מטריצת היעילות במקרים של חוסר התאמה במספר המשימות והמשתמשים, דבר שיכול להיות רלוונטי גם בסביבת העבודה שלנו בארגון ידידים.

יישום בפועל :

המאמר מציג שימוש באלגוריתם ההונגרי, שנמצא יעיל במיוחד בפתרון בעיות הקצאה, כדי להתמודד עם התאמה אופטימלית בין משימות למשתמשים. החלטנו ליישם את האלגוריתם ההונגרי בבחירת המתנדב האופטימלי בפרויקט שלנו, שכן הוא הבחירה המתאימה ביותר ליישום בעמותה ללא מטרות רווח. המודל מצטיין בדיוק גבוה ומהירות חישוב, ומורכבותו הנמוכה יחסית הופכת אותו לאידיאלי להקצאת מתנדבים בצורה יעילה ומדויקת.

נוכח העובדה שהעמותה מתנהלת כיום ללא פלטפורמה המחשבת את בחירת המתנדבים על פי שקלול פרמטרים, אלא רק על פי רצון המתנדב לקחת את האירוע, אנו סבורות שיישום המודל ההונגרי יביא לשיפור משמעותי במדדים של העמותה. השיטה המוצעת תאפשר התאמה מדויקת ומהירה של המתנדבים למשימות, ובכך תשפר את היעילות ותשדרג את ביצועי העמותה.

6.5 מאמר 4

Online Algorithms for Matching Platforms with Multi-Channel Traffic

המאמר עוסק בפיתוח וניתוח של אלגוריתמים מקוונים לפלטפורמות התאמה, בעיקר במצבים שבהם קיימת תעבורה ממספר ערוצים שונים (כגון תנועה פנימית מתוך הפלטפורמה, ותנועה חיצונית ממקורות חיצוניים כמו קישורים ברשתות חברתיות). המחקר מתמקד בהבנת האתגרים הקיימים בשילוב תעבורה ממקורות שונים בפלטפורמות התאמה כמו VolunteerMatch.

מטרת המחקר היא לשפר את היכולת של פלטפורמות התאמה לנהל בצורה אופטימלית את ההתאמה בין מתנדבים להזדמנויות התנדבות. במהלך המחקר הוצע שימוש באלגוריתם חדש AC (Adaptive Capacity) המיועד לשפר את יכולת ההתאמה של פלטפורמות תוך ניצול מקסימלי של המשאבים הזמינים.

מתודולוגיה :

המאמר מציג מספר שיטות ואלגוריתמים חדשניים שנועדו לשפר את יעילות תהליכי ההתאמה בפלטפורמות שונות, במיוחד כאשר קיימת תעבורה ממקורות מגוונים. המתודולוגיה מתמקדת במספר אסטרטגיות עיקריות :

מטריצת עלויות (Cost Matrix): מטריצת העלויות מייצגת את העלות הקשורה להתאמת כל מתנדב לכל משימה בפלטפורמה ונבנית על בסיס פרמטרים שונים המוגדרים מראש. מטריצה זו מהווה את הבסיס לאלגוריתמים השונים, כולל האלגוריתם ההונגרי, לחישוב ההתאמות האופטימליות.

אלגוריתם MSVV (Multiplicative Update Algorithm for Sponsored Search Markets) :

האלגוריתם נועד לטפל במצבים שבהם קיימת תנועה חיצונית משמעותית לפלטפורמה, ומאפשר התאמה דינמית בין משאבים לפי משקל משימה. הוא משתמש בנוסחת משקלים כדי לקבוע את החשיבות היחסית של כל משימה ומשתמש בעדכוני כפולים כדי להשיג איזון מיטבי בין משאבים ותוצאות התאמה.

אלגוריתם AC (Adaptive Capacity) : מתמקד בניהול דינמי של קיבולת המשאבים, תוך התאמת הקיבולת בהתאם לתנאים משתנים של תעבורה במערכת. אלגוריתם זה חשוב במיוחד במצבים שבהם קיימת תנועה חיצונית משמעותית בפלטפורמה.



יחס תחרותי (Competitive Ratio): מטרתו היא להעריך את היעילות של האלגוריתם בהשוואה לאלגוריתם אופטימלי, שיודע מראש את סדר ההגעה של המתנדבים.

$$\min \left(1, \frac{\sum_{i=1}^n v_i}{\sum_{j=1}^m C_{ij}} \right) = CR$$

vi הוא ערך המשימה i.

Cij היא עלות התאמת המתנדב i למשימה j.

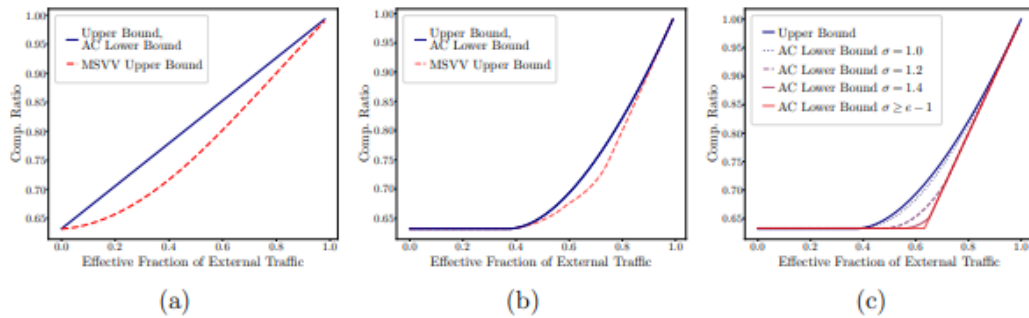


Figure 2 In the asymptotic regime, we present bounds for settings where (a) all external traffic arrives first, (b) arrivals are general and sign-ups are deterministic, and (c) arrivals and conversion probabilities are general.

האיורים מציגים את הביצועים של האלגוריתם AC בהשוואה לאלגוריתם MSVV בתרחישים שונים של תנועה חיצונית ופנימית. גרפים אלו מדגימים כיצד האלגוריתם משפר את יחס התחרותי בתנאים מסוימים, ומציגים את הגבולות העליונים והתחתונים של ביצועי האלגוריתמים, תוך השוואה בין התרחישים השונים.

תוצאות המחקר:

תוצאות המחקר מצביעות על כך שהשימוש בסינון גרידי והגישה ההיברידית הביאו לשיפור של כ-20% ביעילות האלגוריתם ההונגרי, עם יחס תחרותי של כ-1.1 בממוצע, מה שמעיד על עלייה ניכרת בדיוק ההתאמות ללא תוספת משמעותית של זמן ריצה.

מסקנות:

השיטות והאלגוריתמים שהוצגו במאמר, במיוחד הסינון הגרידי והגישה ההיברידית, מאפשרים להתמודד בצורה יעילה יותר עם התאמות במצבים של תעבורה ממקורות מגוונים. השימוש בגישה ההיברידית, שבה אלגוריתם מתקדם משולב עם שיטות סינון ראשוניות, מוביל לשיפור בזמני תגובה ודיוק בהתאמות. המאמר מציג תוצאות אמפיריות המראות כי האלגוריתם AC משיג שיפור משמעותי בביצועים בהשוואה לאלגוריתמים קיימים. לדוגמה, נצפה שיפור ניכר במספר ההתאמות המוצלחות, בעלויות ההתאמה ובניצול המשאבים בתנאי תעבורה מגוונים. כמו כן, יחס תחרותי (Competitive Ratio) מראה שהשיפורים מצמצמים את עלויות ההתאמה ומעלים את אחוזי ההצלחה בהשוואה לאלגוריתמים קיימים, תוך ניצול מיטבי של המשאבים.

התרומה של המאמר לפרויקט: המאמר תרם לפרויקט בכך שהציג כלים ושיטות יעילים לשיפור תהליך ההתאמה בין מתנדבים למשימות. הטמעת הסינון הגרידי והגישה ההיברידית בפרויקט הביאה להפחתת העומס החישובי והעלתה את הדיוק בהתאמות, מה שהוביל לשיפור משמעותי בזמני התגובה ובאחוזי ההצלחה של ההתאמות בין המתנדבים למשימות. בנוסף, השימוש ביחס תחרותי איפשר למדוד את היעילות של האלגוריתם ההונגרי ולהוכיח את השיפור לעומת שיטות פשוטות יותר.



An Optimization Model for Volunteer Assignments in Humanitarian Organizations

המאמר מציג מודל אופטימיזציה רב-קריטריונים לניהול שיבוץ מתנדבים בארגונים הומניטריים. המאמר מתמקד באתגרים המיוחדים של ניהול מתנדבים ושיבוץ משימות למתנדבים תוך התחשבות במגבלות תפעוליות והעדפות אישיות. המאמר שימש אותנו כבסיס הבנתי לפיתוח המערכת.

המטרה:

מטרת המאמר היא לפתח כלי שיסייע לארגונים הומניטריים לשבץ מתנדבים בצורה אופטימלית, כך שניתן יהיה לאזן בין הצורך למלא את המשימות בצורה מלאה לבין שמירה על שביעות הרצון וההעדפות של המתנדבים.

פירוט המתודולוגיה:

המודל שהוצע במאמר הוא מודל אופטימיזציה מבוסס תכנות שלם (*Integer Programming*). המודל מבצע תהליך של אופטימיזציה על מנת למצוא את הפתרון הטוב ביותר עבור הארגון, ומשקלל מספר קריטריונים ומטרות כדי להבטיח הקצאת מתנדבים אופטימלית למשימות בארגון. פונקציית המטרה נועדה למקסם את שביעות רצון המתנדבים, תוך מזעור עלויות הקשורות לאי-מילוי משימות והפחתת השיבוצים הלא רצויים.

$$i_j c_{ij} x \sum_{j=1}^m \sum_{i=1}^n = \text{Maximize } Z$$

- העדפות המתנדבים: המודל לוקח בחשבון את העדפות המתנדבים לסוגי המשימות ולמועדים שבהם הם זמינים. בכך הוא שואף לשבץ כל מתנדב למשימה שמתאימה לו בצורה הטובה ביותר, מה שמשפר את שביעות הרצון שלהם.
- עלות אי-מילוי משימות: פונקציות המטרה כוללות גם את העלויות הנובעות מאי-מילוי משימות, כגון השפעות על יעילות הארגון ואובדן אפשרי של הכנסות או משאבים אחרים.
- אופטימיזציה כוללת: המודל מאפשר אופטימיזציה של הקצאת מתנדבים באופן כוללני, תוך שקלול גורמים כמו שביעות רצון, יעילות תפעולית, ועלויות. הוא נועד למזער את אי-ההתאמות בין משימות למתנדבים ולמקסם את הביצועים הכוללים של הארגון.

מגבלות:

- המודל כולל מגבלות שונות, כגון זמינות המתנדבים, סוגי המשימות, מיקומם הגיאוגרפי, והעדפותיהם האישיות.
- מגבלות נוספות כוללות את הצורך להקצות מתנדבים למשימות קריטיות, והשאיפה למזער את השיבוצים הלא רצויים.



מגבלות על ההקצאה :

$$\forall i, 1 \geq \sum_{j=1}^m x_{ij}$$

כל מתנדב יכול להיות משובץ למשימה אחת לכל היותר.

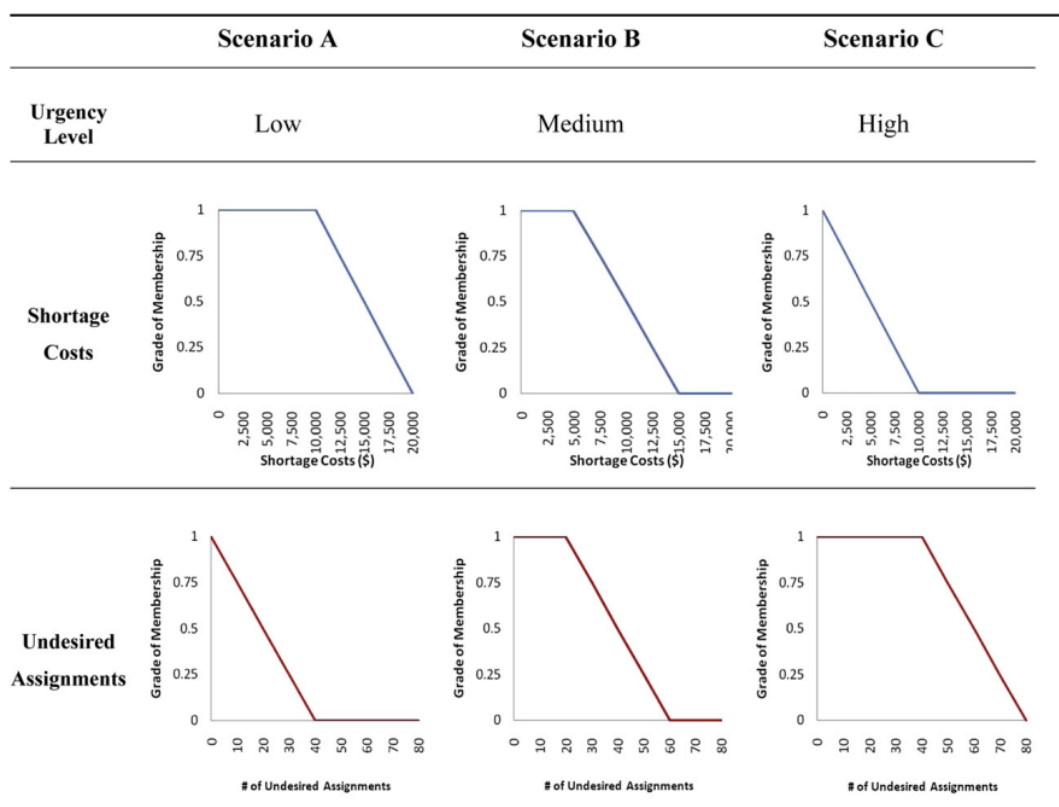
מגבלות של זמינות המתנדבים :

$$Availability_i, \forall i, j \geq t_{ij} \times x_{ij}$$

t_{ij} הוא הזמן שנדרש לביצוע המשימה j על ידי המתנדב i .

תוצאות המחקר :

בטבלה 3 של המאמר מוצגות התוצאות המרכזיות המראות שיפור במדדים :



1. שיפור של 15% במילוי משימות : הקטנת מספר המשימות שלא מולאו בכ-15%.
2. שיפור של 20% בשביעות רצון המתנדבים : הודות להתאמה טובה יותר של המשימות להעדפות האישיות שלהם.
3. הפחתת שיבוצים לא רצויים ב-25% : הפחתת שיבוצים שאינם תואמים את העדפות המתנדבים.



דיון ומסקנות:

המאמר מציג מודל אופטימיזציה רב-קריטריונים לניהול מתנדבים בארגונים הומניטריים, המתמודד עם אתגרים כמו ניהול מתנדבים במספר מיקומים שונים ושילוב משימות קבוצתיות. החוקרים הדגישו את החשיבות של איזון בין ניצול משאבים ומילוי משימות, במיוחד במצבי חירום הומניטריים, והדגישו כיצד המודל מאפשר למקבלי החלטות לשקול את הפשרות הנדרשות בתהליכי הניהול.

המאמר מדגיש את הצורך לשלב את הידע והמומחיות של מקבלי החלטות במודל האופטימיזציה כדי לייצר תוצאות יעילות יותר, ומראה שהשימוש במודל מאפשר גמישות בניהול מתנדבים ומסייע לצמצם את החוסרים בכוח אדם. החוקרים מציעים לחקור בעתיד פונקציות חברות לא-לינאריות ושילוב בין תכנון אסטרטגי ללוח זמנים, כדי לשפר עוד יותר את הדיוק והיעילות של המודל.

התרומה של המאמר לפרויקט: המאמר הציג מגבלות ומדדים מרכזיים שהנחו אותנו בתהליך פיתוח המודל שלנו. במהלך כתיבת המודל בפרויקט "ידידים", התייחסנו בקפידה למגבלות והקריטריונים שתוארו במאמר, תוך שילובם במשוואות המותאמות לצרכים שלנו. כך, הבטחנו שהמודל יתאים לתנאים בשטח ויביא לתוצאות אופטימליות בשיבוץ מתנדבים וביעילותם.

במסגרת המודל, שמנו דגש על:

1. צמצום אי-התאמה בין מתנדב למשימה: מינימיזציה של השיבוצים הלא רצויים על ידי התאמת המתנדב למשימה ולזמן ההגעה, כדי להבטיח ביצוע מיטבי של המשימה.
2. התחשבות במגבלות המתנדב: שקלול זמינות המתנדבים בקבלת ההחלטות, עם העדפה למתנדבים בעלי זמינות גבוהה יותר למשימות הדורשות מענה מהיר, לשיפור היעילות התפעולית.
3. אופטימיזציה כוללת: ביצוע אופטימיזציה מקיפה המתחשבת בכל הקריטריונים הרלוונטיים, במטרה לשבץ את המתנדב המתאים ביותר לכל משימה. כך נשפר את איכות השירות, נבטיח שביעות רצון גבוהה של הפונים, ונחזק את תחושת הסיפוק של המתנדבים, מה שיתרום לשימורם ולהמשך התנדבותם בארגון.

6.7 מאמר 6

Disaster preparedness knowledge and its relationship with triage decision-making among hospital and pre-hospital emergency nurses

המאמר עוסק בקשר שבין הידע בהכנה לאסונות לבין קבלת החלטות במיון בקרב אחיות חירום בבתי חולים ובסביבות פרה-הוספיטליות באיראן.

מטרת המחקר הינה לבדוק כיצד הידע של אחיות בתחום ההכנה לאסונות, משפיע על יכולתן לקבל החלטות מיון במצבי חירום, תוך בחינת משתנים דמוגרפיים ותפעוליים נוספים. המחקר נועד לזהות את הגורמים המקדמים או המעכבים קבלת החלטות מיון מוצלחת במהלך אסונות.

מתודולוגיה:

המחקר השתמש במגוון שיטות סטטיסטיות מתקדמות כדי לבחון את הקשרים בין הידע בהכנה לאסונות לבין קבלת החלטות במיון. בין השיטות והמודלים שהשתמשו בהם:

- מבחני T: ניתוח השוואתי שנועד לבדוק את ההבדלים בין קבוצות שונות (למשל, בין אחיות עם הכשרה לאסונות לעומת אחיות ללא הכשרה).
- ANOVA (Analysis of Variance): ניתוח שונות לבדיקת הבדלים מובהקים בין קבוצות רבות.
- רגרסיה לינארית מרובה: מודל רגרסיה לבחינת ההשפעה של משתנים מרובים (כגון ניסיון, הכשרה, מגורים) על הידע בהכנה לאסונות.
- מבחני קורלציה: לבדיקת הקשרים בין משתנים שונים (כגון קשר בין ידע בהכנה לאסונות לבין החלטות מיון).



המחקר התבסס על נתונים שנאספו מ-472 אחיות חירום, ובחן משתנים כמו מגדר, ניסיון בעבודה, הכשרה, ומיקום העבודה. המתודולוגיה נועדה להבטיח שכל ההשפעות האפשריות יילקחו בחשבון תוך כדי ניתוח הנתונים.

תוצאות המחקר:

המחקר מצא כי ידע מוקדם בהכנה לאסונות, ניסיון בעבודה והכשרה מיוחדת באסונות הם הגורמים המרכזיים המשפיעים על יכולת קבלת ההחלטות של אחיות חירום בזמן אסון. ניתוחים סטטיסטיים תומכים בממצאים אלו. לדוגמה, מבחן T-Test הראה הבדלים מובהקים סטטיסטית בביצועי קבלת ההחלטות בין אחיות עם הכשרה לאסונות לבין אלו ללא הכשרה. ניתוח ANOVA חשף כי רמת הידע בהכנה לאסונות משתנה באופן מובהק בין קבוצות של אחיות עם דרגות ניסיון שונות, והממצאים מצביעים על כך שניסיון רב יותר מוביל לשיפור בקבלת ההחלטות.

דיון ומסקנות:

החוקרים מסיקים כי הכשרה ממוקדת ותכנון מקדים יכולים לשפר באופן ניכר את יכולת ההתמודדות של אחיות במצבי אסון. זה רלוונטי בהקשר של הפרויקט שלנו, שבו שימוש בשיטות סטטיסטיות לבחינת השפעת האלגוריתם ההונגרי על זמני התגובה של המתנדבים, יכול לתרום לשיפור משמעותי בביצועי הארגון.

התרומה של המאמר לפרויקט:

המאמר מדגיש את החשיבות של הכשרה והכנה בתנאים מורכבים וקבלת החלטות אופטימליות, מה שמתחבר בצורה ישירה לפרויקט שלנו, שבו אנו מיישמים אלגוריתמים מתקדמים להקצאת מתנדבים למשימות בהתבסס על מדדים כמו זמני תגובה וניסיון. שיטות הניתוח במאמר, כמו מבחני T ו-ANOVA, דומות לאלו שבהן השתמשנו בפרויקט לצורך הערכת שיפור הביצועים לאחר הטמעת האלגוריתם ההונגרי. השיטות הסטטיסטיות המפורטות במאמר מספקות גיבוי תיאורטי חזק לאופן שבו ביצענו את הניתוחים שלנו.

יישום מתוכנן:

מבחן T-Test: ישמש לבדיקת השפעת האלגוריתם על זמני התגובה של המתנדבים, באמצעות השוואת ממוצעים של זמני תגובה לפני ואחרי ההטמעה.

ANOVA: יאפשר לנו לבדוק האם השיפורים בזמני התגובה הם עקביים בין קבוצות שונות של מתנדבים, כמו מתנדבים באזורים גיאוגרפיים שונים או בזמני פעילות שונים.

6.8 סיכום סקר ספרות

בפרק זה נערכה סקירה מקיפה של הספרות המחקרית בתחום ניהול מתנדבים ואופטימיזציה של משאבים, בדגש על יישום אלגוריתמים מתקדמים כגון האלגוריתם ההונגרי. הסקירה כוללת ניתוח מעמיק של שיטות שונות שנבחנו במחקרים קודמים לשם התאמת משאבים בצורה אופטימלית, תוך שילוב משקלים פרמטריים ונוסחאות מדויקות לחישוב ציוני מתנדבים. כמו כן, נדונו שיטות סטטיסטיות להערכת ביצועים, לבחינת השפעת השינויים והתהליכים שהוטמעו במסגרת הפרויקט.

הסקירה נועדה לבחון את התאמתן של השיטות והמודלים השונים לפרויקט הנוכחי, ולהעריך את התרומה האפשרית של כל אחת מהן לשיפור תהליכי העבודה והביצועים הכוללים. בטבלה המוצגת בהמשך, מופיעים הממצאים העיקריים מהספרות המחקרית שנבחרה, תוך פירוט המטרות, המסקנות והתרומה של כל מאמר לפרויקט, כמו גם היישום בפועל של השיטות שנבחרו במסגרת הפרויקט.



להלן סיכום הממצאים והיישום בפועל כפי שנבחנו מסקירת הספרות:

שם המאמר	מטרת המאמר	תרומה לפרויקט	יישום בפועל
A bipartite graph matching approach for relevance scoring and combinatorial optimization	פיתוח מערכת לשיבוץ אופטימלי של מסמכים למבקרים בכנסים באמצעות גרף דו-צדדי ואלגוריתם הונגרי	המאמר מציע גישה שיכולה להיות מיושמת לא רק בכנסים, אלא גם בתחומים נוספים בהם נדרשת התאמה בין שני סטים של פריטים, תוך הבטחת התאמה אופטימלית על פי קריטריונים מוגדרים מראש. המאמר מציג את האלגוריתם ההונגרי כפתרון יעיל לבעיות הקצאה אופטימלית, במיוחד כאשר קיימות מגבלות כמותיות כמו מינימום ומקסימום של פריטים לשיבוץ. זה רלוונטי לפרויקט שלנו בכך שהוא מדגים כיצד ניתן ליישם את האלגוריתם ההונגרי בצורה מיטבית, באמצעות הרחבת הקריטריונים להתאמה. השימוש בנתוני אמת במאמר מוכיח את יעילותו של האלגוריתם בשיפור התאמה, מה שמפחית את העומס ומבטיח אוטומציה חלקה של התהליך.	חישוב הציון באמצעות TF-IDF לצורך דירוג רלוונטיות של המשימות בהתאמה למתנדבים. לאחר מכן, הפעלת האלגוריתם ההונגרי לביצוע הקצאת משאבים אופטימלית בהתבסס על הקריטריונים שנקבעו, כגון זמינות, מיקום גיאוגרפי ודירוגים היסטוריים. לבסוף, הכנסנו את הערך המנורמל של כל מתנדב למטריצת העלויות כדי לשפר את השיבוץ.
Distributed Matching-By-Clone Hungarian-Based Algorithm for Task Allocation of Multiagent Systems	פיתוח שיטה מתקדמת להתאמת משימות במערכות מרובות סוכנים כאשר מספר הסוכנים קטן ממספר המשימות. השיטה כוללת שילוב של אלגוריתם הונגרי עם משימות דמה וסוכנים משובטים כדי ליצור מטריצת עלויות ריבועית ולהבטיח הקצאה אופטימלית.	השימוש במשימות דמה וסוכנים משובטים מאפשר למערכת להתמודד עם בעיות הקצאה לא סימטריות בצורה יעילה יותר, ולהפחית את העלות הכוללת וזמן ההתכנסות של המערכת. המאמר מספק גישה תיאורטית ותמיכה להוספת משימות דמה במטריצת העלויות בפרויקט שלנו, מה שיכול לשפר את דיוק השיבוץ ולהפחית את עלויות התפעול.	יישמו את הגישה המוצעת במאמר של הוספת משימות דמה במערכת הקיימת שלנו כדי ליצור מטריצת עלויות ריבועית עבור האלגוריתם ההונגרי. השיטה מאפשרת לבצע התאמות משימות אופטימליות גם במצבים בהם מספר המשימות גדול ממספר הסוכנים.
New Task Oriented Recommendation method Based on Hungarian algorithm in Crowdsourcing Platform	המאמר מציג פיתוח שיטה להמלצה על משימות חדשות בפלטפורמות קראודסורסינג, בהן משתמשים מבוזרים מתבקשים לבצע משימות באופן חופשי ובהתנדבות. המחקר מתמודד עם בעית התאמת המשימות המתפרסמות למשתמשים המתאימים ביותר	המאמר מציג את היתרון של שימוש באלגוריתם ההונגרי בשילוב אופטימיזציה מרובה מטרות, דבר שמביא לשיפור דיוק ההמלצות ותפקוד המערכת. הוא מדגים כיצד ניתן לשפר את דיוק השיבוץ על ידי חישוב מטריצות יעילות ושילוב דמיון בין משימות על, מתקשר לפרויקט בתחום השיבוץ האופטימלי של מתנדבים למשימות, תוך התחשבות במספר פרמטרים כמו זמינות, מיקום גיאוגרפי, וזמן תגובה.	שימוש באלגוריתם ההונגרי לביצוע אופטימיזציה של שיבוץ מתנדבים למשימות על בסיס מטריצת יעילות המחושבת לפי זמינות, מיקום גיאוגרפי, ואף את הדיוק של המערכת הקיימת. המודל מצטיין בדיוק גבוה ומהירות חישוב, ומורכבותו הנמוכה יחסית הופכת אותו לאידיאלי



להקצאת מתנדבים בצורה יעילה ומדויקת.		לביצוע. לשם כך, המחקר מציע שימוש באלגוריתם ההונגרי, אשר ידוע ביכולתו לפתור בעיות השמה בצורה אופטימלית.	
הטמעת הגישה ההיברידית בפרויקט, הכוללת סינון גרידי מקדים להפחתת העומס החישובי והפעלת האלגוריתם ההונגרי על קבוצה מצומצמת של מתנדבים. יישום יחס תחרותי למדידת יעילות האלגוריתם בהשוואה לשיטות בסיסיות. התוצאה היא שיפור בזמני התגובה, בעלות הכוללת ובאחוזי ההצלחה של התאמות המתנדבים למשימות.	המאמר מדגיש את חשיבות השימוש באלגוריתמים היברידיים, המשלבים סינון גרידי מקדים עם אלגוריתמים מתקדמים כמו האלגוריתם ההונגרי, כדי להתמודד עם תעבורה ממקורות מגוונים ולשפר את הדיוק והיעילות של ההתאמות. השימוש ביחס תחרותי מאפשר למדוד את הביצועים של האלגוריתם החדש בהשוואה לשיטות קודמות, ומראה כי השיטה החדשה משיגה תוצאות טובות יותר במונחי עלויות התאמה, דיוק וניצול משאבים.	פיתוח וניתוח אלגוריתמים מקוונים לשיפור תהליך ההתאמה בפלטפורמות, במיוחד במצבים של תעבורה ממספר ערוצים שונים. המאמר מתמקד באופטימיזציה של התאמות במערכות מבוזרות תוך שימוש באלגוריתם ההיבריד AC ויחס תחרותי.	Online Algorithms for Matching Platforms with Multi-Channel Traffic
יישמו את הגישה המוצעת במאמר על ידי הוספת עמודה המצביעה על זמינות המתנדב, כדי להבטיח שההקצאה תתבצע לא רק על בסיס היכולות של המתנדב, אלא גם על פי זמינותו. כך, המערכת מתחשבת קודם כל בזמינות המתנדב ולאחר מכן במיקומו ביחס לאירוע, מה שמפחית את העומס וזמן החישוב של המערכת, הנדרש לבחירת המתנדב המתאים. בנוסף, ביצענו חישוב של ציון המתנדב על פי מספר האירועים המוצלחים שהוא ביצע בקטגוריה ספציפית. גישה זו מבטיחה שהמתנדב הנבחר יהיה בעל ניסיון מוכח ויכולת גבוהה לפתור את הבעיה הנדרשת בצורה האופטימלית.	המאמר הציג מגבלות ומדדים מרכזיים שהנחו אותנו בתהליך פיתוח המודל שלנו. במהלך כתיבת המודל בפרויקט "ידידים", התייחסנו בקפידה למגבלות והקריטריונים שתוארו במאמר, תוך שילובם במשוואות המותאמות לצרכים שלנו. בכך הבטחנו שהמודל יתאים לתנאים בשטח ויביא לתוצאות אופטימליות בשיבוץ מתנדבים וביעילותם.	המטרה המרכזית של המאמר היא לשפר את היעילות והאפקטיביות של תהליך הקצאת המתנדבים לאירועים או משימות שונות באמצעות שימוש במודלים אופטימיזציה מתקדמים.	An Optimization Model for Volunteer Assignments in Humanitarian Organizations



השיטות הסטטיסטיות המוזכרות במאמר, כגון מבחני T-Test ו-ANOVA, שימשו בפרויקט להערכת השיפורים בזמני התגובה של המתנדבים לאחר הטמעת האלגוריתם ההונגרי. הניתוחים אפשרו לאמוד בצורה מדעית את השיפור שחל בביצועים בעקבות המודל.	המחקר מראה כי ידע בהיערכות לאסונות משפיע באופן מובהק על יכולת קבלת ההחלטות של צוותי החירום. ממצאים אלו מדגישים את החשיבות של הכשרה ממוקדת במצבי חירום ושימוש במבחני T-Test ו-ANOVA להערכת השפעות שונות על ביצועים. תרומה זו רלוונטית לפרויקט שלנו בהערכת השיפור בביצועי המתנדבים לאחר יישום האלגוריתם.	המאמר חוקר את ההשפעה של ידע בהיערכות לאסונות על קבלת החלטות במצבי טריאז' בקרב אחיות חדר מיון ורפואת חירום. מטרת המחקר היא להעריך את הקשר בין ידע זה לבין יכולת קבלת החלטות בזמן חירום, תוך שימוש במודלים סטטיסטיים להערכת משתנים שונים.	Disaster preparedness knowledge and its relationship with triage decision-making among hospital and pre-hospital emergency nurses
---	--	---	---

מגוון מחקרים עסקו בבעיות ניהול מתנדבים ואופטימיזציה של משאבים ארגוניים, והציעו מודלים ושיטות הנדסיות מתקדמות לשיפור תהליכי הקצאה ותפעול בארגונים. הספרות המחקרית מציגה גישות מגוונות להתמודדות עם אתגרי ניהול מתנדבים, תוך התחשבות בפרמטרים כמו זמינות, מיקום גיאוגרפי, ודירוגים היסטוריים, על מנת להבטיח דיוק ויעילות בתהליכי הקצאה.

לדוגמה, מחקרם של Knoll ו-Schauer, מציג [1] מודל לניהול משאבים בארגון באמצעות אלגוריתם הונגרי, המשולב בחישוב הציון באמצעות TF-IDF לצורך דירוג רלוונטיות של המשימות בהתאמה למתנדבים. מודל זה מסייע להרחיב את הקריטריונים להתאמה, כגון זמינות, מיקום גיאוגרפי ודירוגים היסטוריים, דבר שהינו חיוני לשיפור הדיוק בשיבוץ. בפרויקט שלנו, אומצה גישה זו לשיפור תהליך השיבוץ של מתנדבים למשימות, תוך דגש על דיוק בהתאמה ושיפור ביצועים.

מאידך, מחקרו של Law, [2] מציע שיטה ייחודית להתמודדות עם מצבים שבהם יש אי-התאמה במספרים בין מתנדבים למשימות באמצעות הוספת משימות דמה ליצירת מטריצות עלויות ריבועיות, המבטיחות שההקצאה תהיה אופטימלית גם במצבים של חוסר איזון. גישה זו אומצה בפרויקט שלנו כשיטה מרכזית לטיפול בבעיות אי-התאמה, והיא הביאה לשיפור ניכר ביכולתנו לבצע הקצאות מדויקות ויעילות יותר.

מצד שני, מחקרם של Shi ושותפיו [3] מספק תובנות על השימוש באלגוריתם ההונגרי להקצאת משימות בפלטפורמות קראודסורסינג, תוך אופטימיזציה מרובת מטרת. בהשראתו, יישמנו את הגישה המוצעת בפרויקט, במטרה להבטיח שהשיבוץ יתבצע תוך התחשבות בפרמטרים קריטיים כמו זמינות ומיקום. השיטה המוצעת תאפשר התאמה מדויקת ומהירה של המתנדבים למשימות, ובכך תשפר את היעילות ותשדרג את ביצועי העמותה.

בנוסף, המחקר של Gunasekaran ושותפיו [5] הדגיש את החשיבות של זמינות וניסיון המתנדבים בניהול משאבי אנוש בארגונים הומניטריים. התובנות ממחקר זה השפיעו ישירות על מבנה המודל שלנו, בכך שהוספנו שכבת בחינה של זמינות המתנדב לפני כל הקצאה, מה ששיפר את היעילות התפעולית ואת ביצועי המערכת שלנו.

מחקרם של Xu ושותפיו [4] מדגיש את היתרון של שילוב סינון גרידי עם אלגוריתם ההונגרי, המאפשר להתמודד עם עומסי תעבורה ממקורות שונים ולשפר את הדיוק והיעילות של ההקצאות. גישה זו, ששולבה בפרויקט שלנו, סייעה בשיפור זמני התגובה ובדיוק השיבוצים, במיוחד במצבים מורכבים שבהם עומסי התעבורה מגוונים.

לבסוף, מחקרם של Mahoney ושותפיו [6] מציע כלים סטטיסטיים חיוניים כמו מבחני T-Test ו-ANOVA להערכת ביצועים, אשר שימשו אותנו להערכת השיפור בזמני התגובה ובדיוק ההקצאות לאחר יישום האלגוריתם ההונגרי.

לפי סקירת הספרות, ניתן לראות כי מספר שיטות אופטימיזציה, כגון השימוש באלגוריתם ההונגרי בשילוב עם משימות דמה והערכה סטטיסטית, תרמו באופן משמעותי לפרויקט שלנו. הגישות



שהוצגו במאמרים השונים סייעו לנו לפתח פתרון מקיף ויעיל, התואם לצרכים הספציפיים של ארגון ידידים, ומספק שיפורים משמעותיים בתהליכי ניהול המתנדבים והמשאבים.

*יישום המתודולוגיות בעזרת הסקירה בפועל מפורט בנספחים

6.9 ביבליוגרפיה

1. B. Knoll and A. Schauer, "A bipartite graph matching approach for relevance scoring and combinatorial optimization," Expert Systems with Applications, vol. 93, pp. 1-9, March 2018. DOI: 10.1016/j.eswa.2017.09.045
2. E. H. Law, "Distributed matching-by-clone Hungarian-based algorithm for task allocation of multiagent systems," IEEE Transactions on Automation Science and Engineering, vol. 12, no. 4, pp. 1309-1322, Oct. 2015. DOI: 10.1109/TASE.2015.2471019
3. Z. Shi, D. Gong, X. Yao, and M. Yang, "New task oriented recommendation method based on Hungarian algorithm in crowdsourcing platform," in 2020 IEEE World Congress on Services (SERVICES), 2020, pp. 134-144. DOI: 10.1109/SERVICES48979.2020.00040
4. C. Xu, Z. Shi, and D. Gong, "Online algorithms for matching platforms with multi-channel traffic," Proceedings of the IEEE, vol. 108, no. 3, pp. 399-414, 2020. DOI: 10.1109/JPROC.2019.2946253
5. A. Gunasekaran, N. Subramanian, and P. Rahman, "An optimization model for volunteer assignments in humanitarian organizations," BMC Health Services Research, vol. 22, no. 1, p. 309, 2022. DOI: 10.1186/s12913-022-08311-9
6. E. J. Mahoney, J. Murphy, and T. Wilson, "Disaster preparedness knowledge and its relationship with triage decision-making among hospital and pre-hospital emergency nurses," BMC Health Services Research, vol. 19, p. 559, 2019. DOI: 10.1186/s12913-022-08311-9

7. תיאור הנדסי של המצב הקיים

7.1 תיאור המצב הקיים :

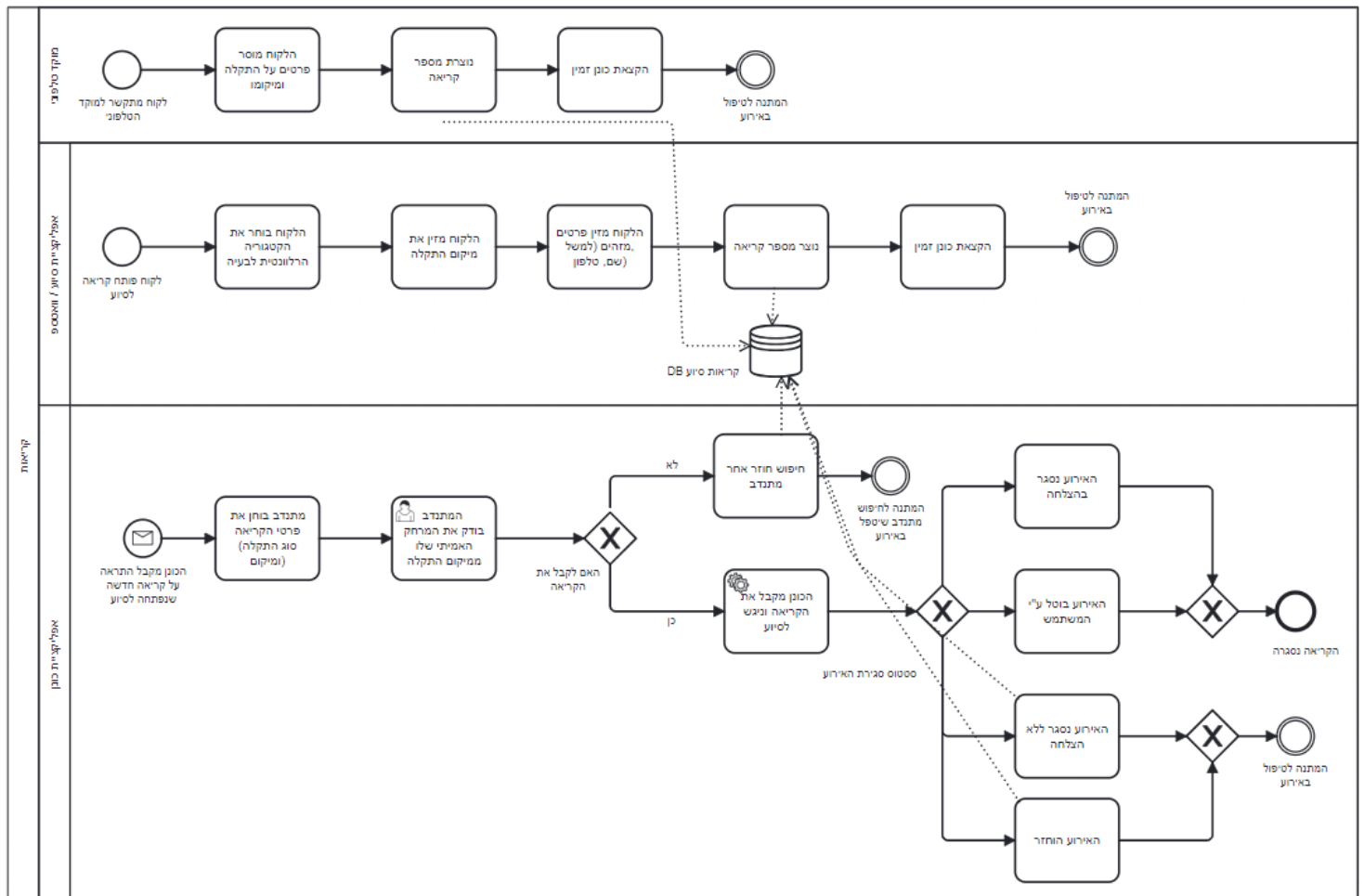
כפי שצוין, לארגון ישנן שתי דרכים מרכזיות לקלוט קריאה חדשה.

● דרך א'- בעזרת אפליקציית 'ידידים' - סיוע' שדרכה אזרחים הזקוקים לעזרה מזינים את הנתונים הנדרשים (כגון סוג התקלה, מיקום גיאוגרפי ופרטי קשר) והבקשה נקלטת במערכות העמותה.

● דרך ב'- בעזרת שיחת טלפון למוקד הארצי של העמותה, אזרחים המתקשרים למוקד מועברים למוקדן אשר רושם את הפרטים הנדרשים במערכת המידע של העמותה.

בכל אחת מהדרכים הללו, הציבור יכול להזמין סיוע באמצעות בחירת הבעיה או הקטגוריה שבה נדרשת עזרה, והכנסת מיקום התקלה יחד עם הפרטים המזהים שלו. הקריאה הנכנסת מועברת לאפליקציית 'ידידים - כונן', המיועדת לשימוש המתנדבים בלבד ומאפשרת להם לקבל פרטים אודות הקריאה בזמן אמת.

המתנדבים יכולים לראות את סוג התקלה, מיקום התקלה ופרטי הקשר של מבקש הסיוע. על סמך המידע הזה, הם מחליטים אם לקחת על עצמם את המשימה. כל תהליך ההחלטה מתקיים ברמת המתנדב הבודד, ללא שקילה של פרמטרים אובייקטיביים כמו זמינות מתנדבים אחרים, קרבת מיקום או ניסיון קודם בטיפול בבעיה דומה.





חסרונות בתהליך הנוכחי:

- תהליך אינו אופטימלי: תהליך קבלת ההחלטות נעשה בצורה ידנית על ידי המתנדבים עצמם, מה שעלול להוביל לשיבוצים לא אופטימליים ולזמני תגובה ארוכים.
- חוסר אוטומציה: אין מערכת שמבצעת את השיבוץ על פי קריטריונים אובייקטיביים כמו קרבת המתנדב לתקלה או זמינותו המיידית.
- יכולת התרחבות מוגבלת: במצב הנוכחי, ככל שמספר הקריאות עולה, הקצאת המשאבים הופכת לפחות יעילה, מה שיכול להוביל לעיכובים בטיפול בתקלות ולפגיעה בשירות.

7.2 חקר מדדים הקדמה:

לאחר מיפוי וזיהוי הפערים בתהליכי העבודה הקיימים, ביצענו ניתוח מעמיק של מדדים כמותיים ואיכותניים, שמטרתו להעריך את המצב הנוכחי של פעילות הארגון ולזהות תחומים לשיפור. במהלכו, התגלו פערים בתהליך הבחירה שמובילים לעיכובים בזמני הטיפול ולעיתים אף לכשלים בטיפול באירועים.

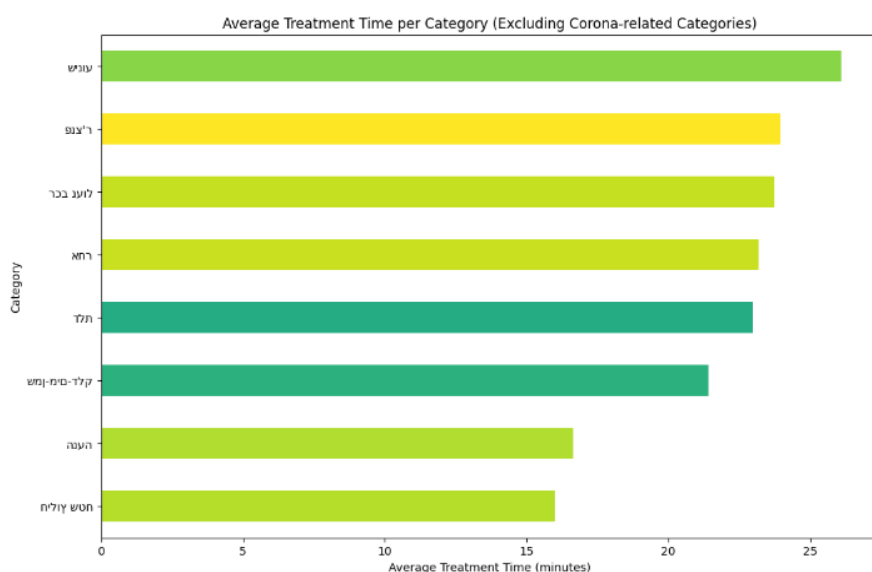
במסגרת התהליך, התמקדנו בזיהוי ובחינת מדדים מרכזיים שיאפשרו לנו לאמוד את היעילות התפעולית של הארגון במצבו הנוכחי, טרם ההטמעה של מערכת האופטימיזציה המוצעת. מדדים אלו נבחרו בקפידה בעקבות ניתוח מעמיק של תהליכי העבודה ושל הנתונים שנאספו ממערכות הארגון, תוך סיעור מוחות עם נציגי הארגון. תהליך זה נועד להבטיח שהמדדים הנבחרים משקפים נאמנה את המציאות בשטח, ויהיו נקודת ייחוס מדויקת להשוואה עם התוצאות הצפויות לאחר יישום המערכת.

זמן טיפול ממוצע (Average Treatment Time):

*זמן טיפול מתאר את הזמן בו המתנדב קיבל על עצמו את הטיפול בתקלה ועד לסיום הטיפול.

הארגון מתמודד עם בעיות בטיפול מהיר באירועים שונים. זמן הטיפול הממוצע הוא אחד המדדים החשובים ביותר להערכת יעילות התגובה לאירועים. ניתוח הנתונים הקיימים מצביע על חסמים שמובילים לזמני תגובה ארוכים ולעיתים אף לכשלים בהקצאת המתנדב הנכון לאירוע. לדוגמה, הקטגוריות עם זמן הטיפול הממוצע הגבוה ביותר משויכות לתקופת הקורונה, שאינה רלוונטית לתיחום הפרויקט שלנו ולכן לא נכללה בניתוח.

נשתמש בגרף עמודות אופקי כדי להציג את זמן הטיפול הממוצע לכל קטגוריה בצורה ויזואלית:



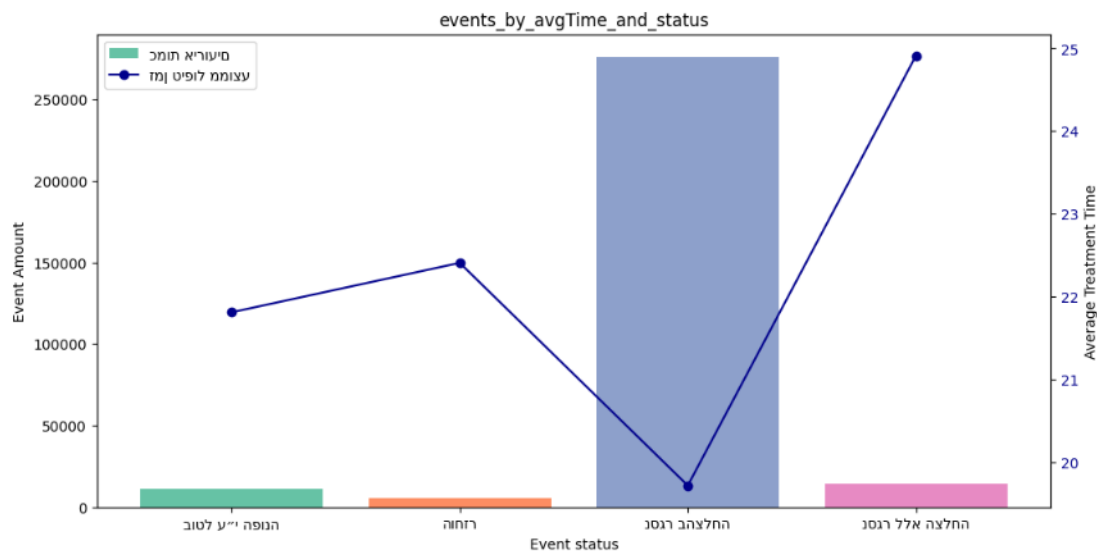
קטגוריה	אחר
25.280388	דלת
24.568449	הנעה
18.064785	חילוץ שטח
16.000000	פנצ'ר
25.305340	רכב נעול
24.669174	שינוע
31.228216	שמן-מים-דלק
23.869202	



מהניתוח שבוצע, עולה כי קטגוריות מסוימות מציגות זמני טיפול ממוצעים גבוהים יותר מאחרות. הקטגוריה "שינוע" מציגה את זמן הטיפול הממוצע הגבוה ביותר, עם מעל 26 דקות בממוצע לאירוע. לאחריה נמצאות הקטגוריות "רכב נעול" ו"פנצ'ר", עם כ-23 דקות בממוצע. לעומת זאת, קטגוריות כמו "הנעה" ו"חילוץ שטח" מציגות זמני טיפול ממוצעים נמוכים יותר, עם כ-16 דקות או פחות.

בכדי לבחון האם קיים קשר בין זמן הטיפול לבין הצלחת האירוע, השתמשנו בגרף פיזור (Scatter Plot), המציג את המגמות השונות בזמני הטיפול עבור כל סטטוס אירוע:

ציר X: מייצג את סטטוס האירוע ("נסגר בהצלחה", "נסגר ללא הצלחה", "בוטל", "הוחזר").
ציר Y: מייצג את זמן הטיפול הממוצע באירועים בסטטוס הנתון.



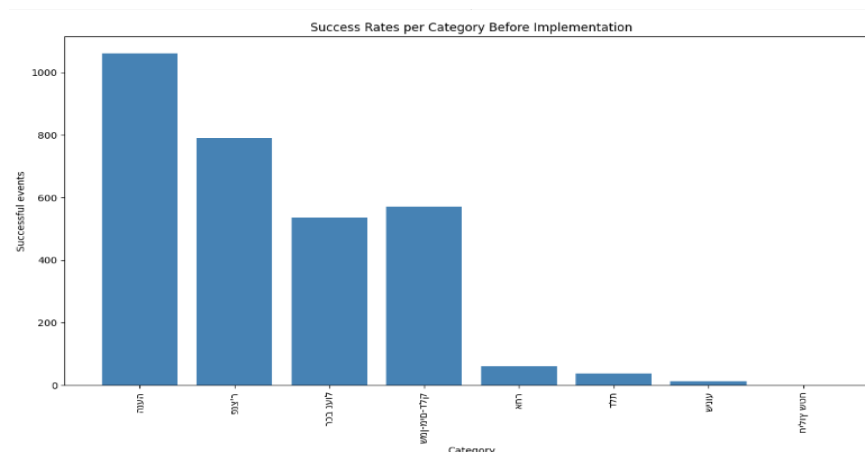
בבחינת פיזור הנתונים ב Scatter Plot, ניתן לראות פערים משמעותיים בזמני הטיפול בין האירועים השונים המציגים פיזור רחב יותר. פערים אלו מעידים על קושי בקביעת מתנדב מתאים ובניהול יעיל של תהליך הסגירה של האירועים.

הגרף מראה כי בסטטוס "נסגר בהצלחה" זמן הטיפול נמוך יותר בהשוואה לסטטוסים אחרים, מה שמעיד על כך שככל שהמתנדב מנוסה יותר ומתאים לאירוע, זמן הטיפול קצר יותר והסבירות להצלחה עולה. לעומת זאת, בסטטוס "הוחזר" נצפה זמן טיפול מעט גבוה יותר, מה שיכול לנבוע מחוסר מיומנות או חוסר יכולת לטפל באירוע.

- נראה כי במצב הנוכחי, זמן הטיפול הממוצע באירועים עומד על כ-23 דקות, עם הבדלים משמעותיים בין הקטגוריות. היעד לאחר ההטמעה הוא להקטין את הפערים הללו ולשפר את היעילות הכוללת.

אחוזי הצלחה (Success Rate):

בניתוח מדד אחוזי ההצלחה, זוהו בעיות בטיפול יעיל באירועים שונים, כאשר כמות משמעותית של אירועים לא נסגרו בהצלחה או בוטלו. מדד זה חיוני להערכת איכות השירות והתוצאה הסופית של תהליך הטיפול באירועים בארגון.



הקטגוריה
"הנעה" מציגה
את מספר

האירועים המוצלחים הגבוה ביותר, עם כ-1,200 אירועים שנסגרו בהצלחה כשאחריה, הקטגוריה "פנצ'ר" מציגה מספר גבוה של אירועים מוצלחים, עם כ-800 אירועים שנסגרו בהצלחה. קטגוריות כמו "שינוע", "דלת" ו"אחר", מציגות מספר אירועים מוצלחים נמוך במיוחד.

על מנת לשפר את יכולת הארגון ולוודא שהאירועים נסגרים בהצלחה, ניתחנו את כמות האירועים לפי הסטטוסים השונים:

אירועים שנסגרו ללא הצלחה: נשאף לנתח את הסיבות לכישלונות אלו ולפעול לתיקון על מנת לשפר את השירות.

אירועים שבוטלו ע"י הפונה: נבחן את הסיבות לביטולים ולשפר את התקשורת עם מבקשי הסיוע, על מנת לנסות ולהקטין את כמות הביטולים.

אירועים שהוחזרו: נחקור את הסיבות ונפעל למניעת מקרים דומים בעתיד

סטטוס אירוע
276099
14377
11312
5738

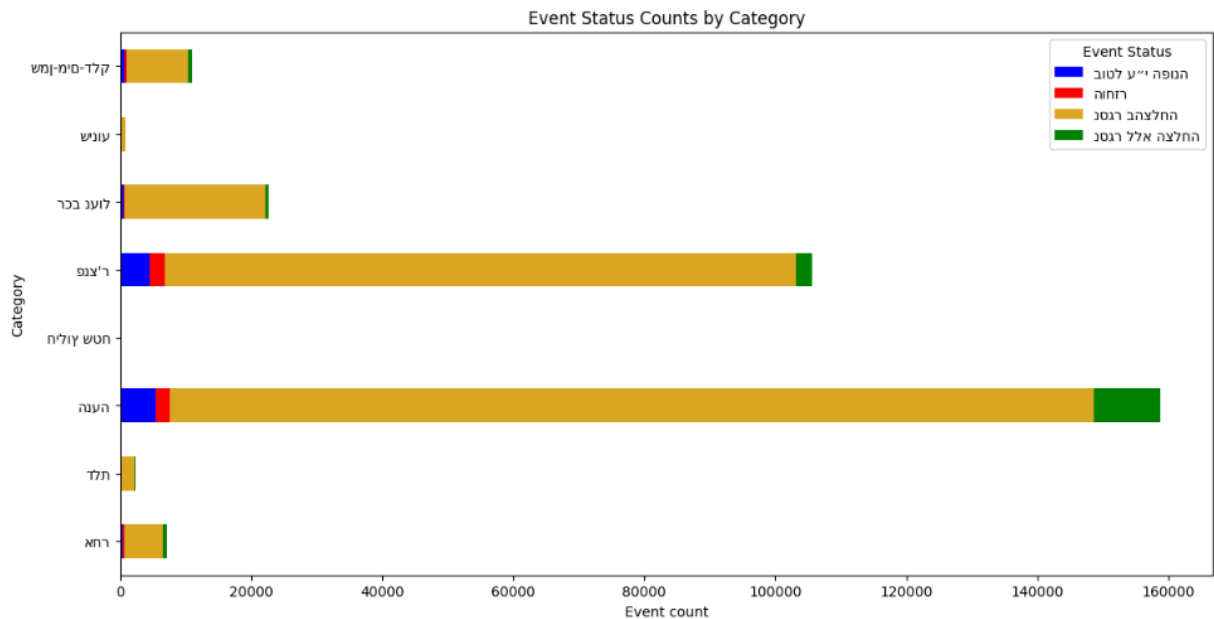
נסגר בהצלחה
נסגר ללא הצלחה
בוטל ע"י הפונה
הוחזר

Event Status Counts by Category (Excluding Corona-related Categories)

קטגוריה	סטטוס אירוע	בוטל ע"י הפונה	הוחזר	נסגר בהצלחה	נסגר ללא הצלחה
אחר	303	272	5870	655	
דלת	83	71	1939	157	
הנעה	5339	2275	140997	10225	
חילוץ שטח	2	6	11	0	
פנצ'ר	4424	2400	96312	2412	
רכב נעול	459	233	21496	430	
שינוע	68	85	597	31	
שמן-מים-דלק	623	381	9349	504	



הגרף הבא מציג את כמות האירועים לפי קטגוריה וסטטוס אירוע. הוא מסייע להבין את חלוקת האירועים בכל קטגוריה בצורה מהירה וברורה:

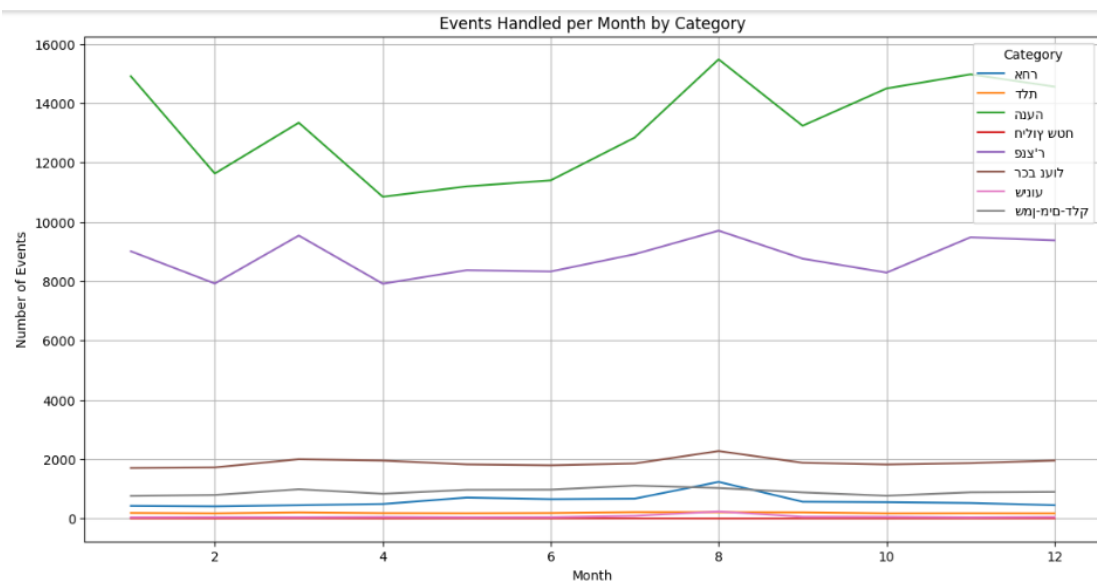


הגרף מראה כי קטגוריות מרכזיות כמו "הנעה", "רכב נעול", ו-"פנצ'ר" מציגות מספר גבוה של אירועים מוצלחים (סטטוס "נסגר בהצלחה"), עם אחוז קטן יחסית של אירועים שנסגרו ללא הצלחה, הוחזרו או בוטלו. קטגוריות כמו "שמן-מים-דלק" ו"חילוץ שטח" מציגות מספר נמוך יותר של אירועים, אך עדיין עם אחוזי הצלחה גבוהים יחסית.

לאחר הגדרת היעד, מנהל אגף טכנולוגיות מהארגון פנה אלינו בבקשה לבחון את כמות האירועים שטופלו בפרקי זמן שונים, כשהמטרה היא לזהות דפוסי פעילות עונתיים ושעות שיא במהלך היום, כדי לייעל את ניהול המתנדבים ולשפר את התהליכים.

בגרפים המוצגים להלן ניתן להבחין בתבניות חשובות שמציעות תובנות לגבי דפוסי הפעילות והשירות בארגון, ונשענות על תצפיות שנאספו לאורך זמן.

גרף חלוקת כמות האירועים לפי חודשים (Events Handled per Month by Category)

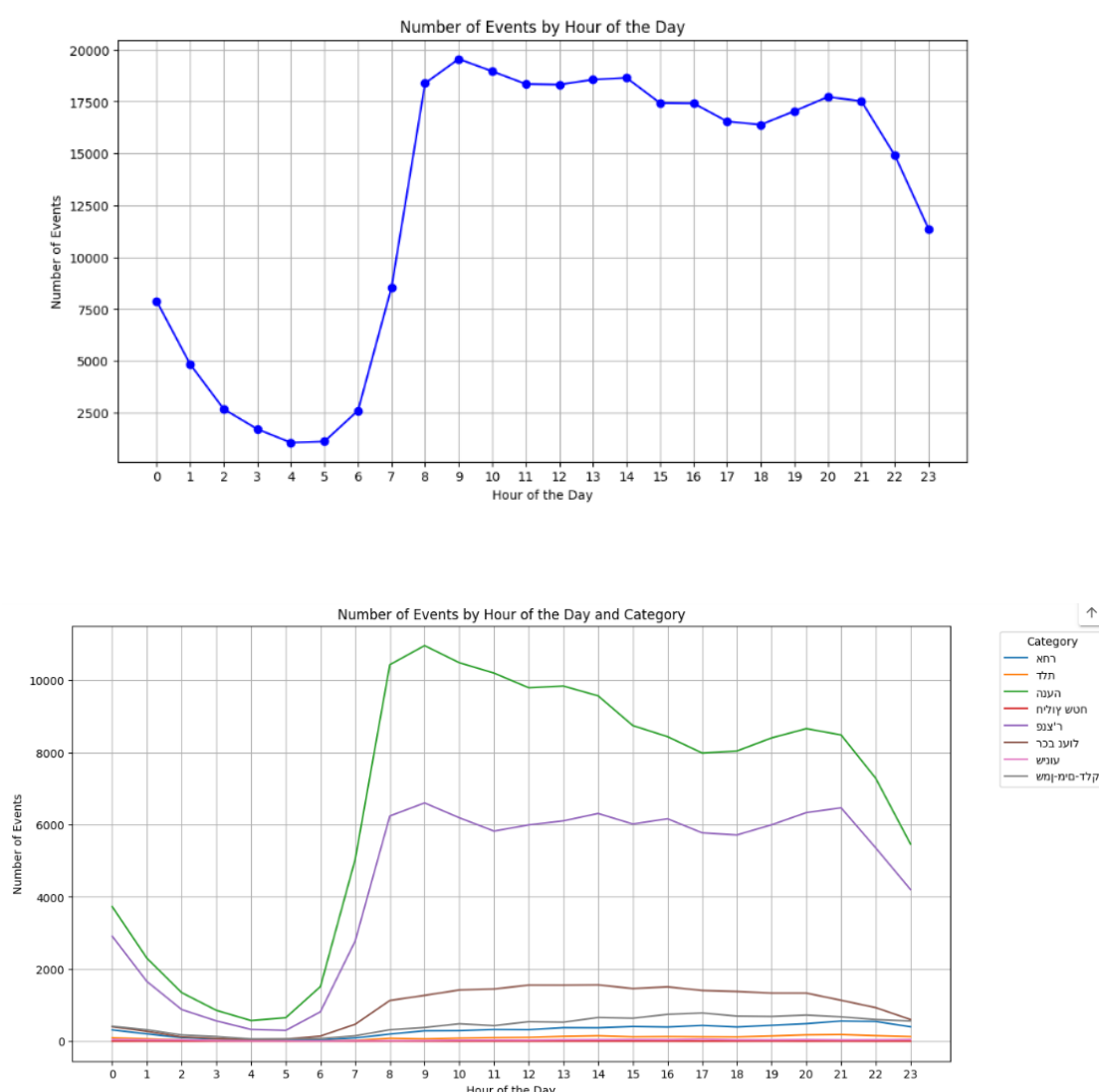




מהגרף ניתן להסיק דפוסי עונתיות העשויים להצביע על תקופות של עומס או חוסר יעילות. הקטגוריות "הנעה" ו-"פנצ'ר" מציגות את כמות האירועים הגבוהה ביותר בכל חודש, ייתכן כי תנאי מזג האוויר בחודשי הקיץ והחורף משפיעים ישירות על התקלות ברכבים. ניכרת ירידה בכמות האירועים בחודשים ינואר ומרץ, ייתכן שזו קשורה למזג האוויר הקר והשפעתו על המערכות המכניות של הרכבים. עלייה נרשמה לאחר מכן, ככל הנראה עקב הכנה לחורף או גשמים ראשוניים.

קטגוריות כמו "רכב נעול" ו-"שמן-מים-דלק" מציגות כמות אירועים נמוכה ויציבה לאורך השנה. קטגוריות אלו אינן דורשות תנודתיות או שינוי משמעותי במבנה הארגוני או בהקצאת משאבים, ולכן הן יכולות להיות נקודת התייחסות. יעילות הטיפול בקטגוריות אלו ניכרת בכך שהן דורשות פחות משאבים, ולכן ניתן למקד את המאמצים בשיפור הקטגוריות המרכזיות הדורשות יותר תשומת לב, כמו "הנעה" ו-"פנצ'ר".

גרף כמות האירועים לפי שעות ביום (Number of Events by Hour of the Day and Category):



הגרף מדגיש את שעות השיא בפעילות הארגון, עם עלייה חדה בכמות האירועים בשעות הבוקר המוקדמות (06:00-10:00), כאשר שיא הפעילות נרשם בין 08:00-09:00. קטגוריות כמו "הנעה" ו-"פנצ'ר" מציגות את העלייה המשמעותית ביותר בשעות אלו. לאחר שעות העומס, מספר האירועים נשאר יציב יחסית, עם ירידה ניכרת לאחר השעה 16:00 ובמהלך שעות הערב והלילה. דפוס זה מצביע על כך שבעיות רכב פחות שכיחות ככל שהיום מתקדם.



ניתוח זה מצביע על כך שרבים נתקלים בבעיות רכב בדרכם לעבודה או בשעות העומס של הבוקר, וכתוצאה מכך נדרש צורך גבוה במתנדבים בזמנים אלו. לאחר שעות העומס, מספר האירועים נשאר יציב יחסית, במיוחד בקטגוריות כמו "רכב נעול", מה שמעיד על ביקוש קבוע לעזרה לאורך היום.

גרף אחוזי הצלחה לפי שעה ביום (Success Rate by Hour of the Day)

הגרף מראה כי אחוזי ההצלחה הגבוהים ביותר הם בשעות הבוקר המאוחרות, מה שמתאים למסקנה שהמתנדבים מתפקדים טוב בשעות השיא. נראה כי ישנה עלייה חדה בשעות 7:00-9:00, שבהן אחוזי ההצלחה עולים ל-91%.



בשעות אחר הצהריים והערב ניכרת יציבות יחסית, עם ירידה קלה בשעות הבוקר המוקדמות (4:00-6:00) ובשעות הערב המאוחרות (21:00-23:00), ככל הנראה בשל מספר מועט של אירועים בשעות אלו.

מתוך הניתוח שביצענו, ניתן להסיק כי תכנון נכון של אופן הקצאת המתנדבים בשעות העומס המשליך על עומס בקריאות כמובן באותם זמנים, יכול להביא לשיפור משמעותי באחוזי ההצלחה של האירועים.

הצעת פיצור חדש: במהלך חקר המצב הקיים, עלה הצורך לאמוד את טיב ביצועי המתנדבים באמצעות פיצור חדש שיאפשר למבקשי הסיוע לדרג את איכות השירות שקיבלו. פיצור זה יאפשר מעקב ושיפור מתמיד של רמת השירות הניתן על ידי המתנדבים.

שביעות רצון המתנדבים-

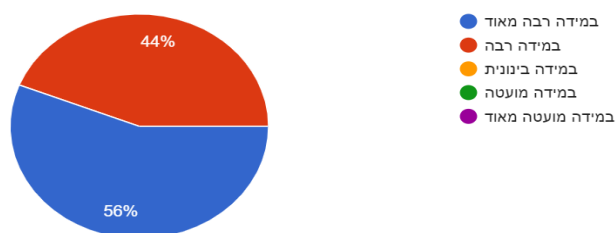
מדד נוסף שאנו מבקשים לבחון הוא שביעות רצון המתנדבים, מדד איכותני בעל חשיבות רבה לארגון. שביעות רצון המתנדבים משקפת את רצונם להמשיך ולהשתתף בפעילות הארגון לאורך זמן, ומהווה גורם מכריע בהבטחת הצלחתו. מדד זה חיוני להפעלת מתנדבים בעלי ביצועים וכישורים טובים, ומצביע על המוטיבציה של המתנדב להמשיך לפעול בארגון ולטפל באירועים בצורה מהירה ויעילה.

המדד נבדק באמצעות שאלונים, בהם השתתפו כ-50 מתנדבים שהתבקשו למלא אותם.



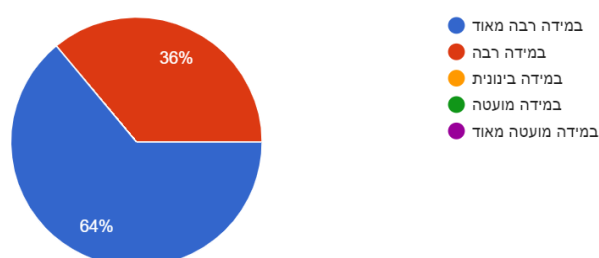
אם היית יודע שהעמותה עושה ככל שביכולתה כדי לשלוח אותך לאירועים המתאימים ביותר, האם היית רוצה לתרום יותר?

50 תגובות



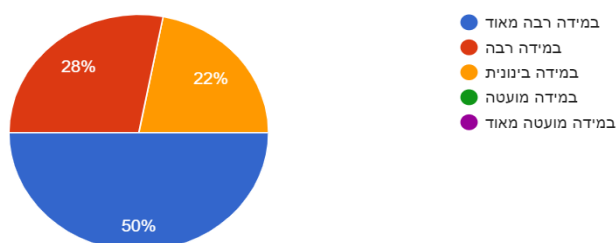
האם אתה מרגיש שכאשר אחוזי ההצלחה שלך יעלו, זה יגרום לך לרצות להתנדב יותר?

50 תגובות



אם היית יכול לבחור את הקטגוריה שבה תרצה לקבל פניות למשימות, האם זה היה משפיע על בחירתך לקבל יותר אירועים?

50 תגובות

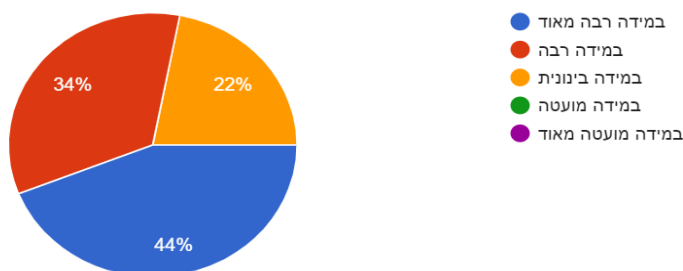


מניתוח שאלה זו עולה כי המתנדבים אכן מביעים רצון לקחת חלק פעיל בבחירת הקטגוריות שבהן הם יקבלו פניות. בחירה זו עשויה להגדיל את מעורבותם ואת נכונותם לקחת על עצמם יותר אירועים.



אם היית מקבל פידבק על הביצועים שלך לאחר כל משימה, האם זה היה מעודד אותך להשתפר?

50 תגובות



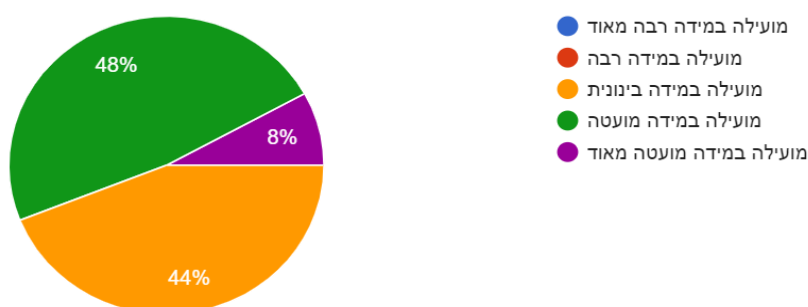
מסיכום ניתוח השאלון ניתן להסיק כי המתנדבים אכן מעוניינים בשינוי המצב הקיים. הם מביעים רצון לקבל פידבק על עבודתם ולקחת חלק פעיל יותר בקטגוריות שבהן הם מרגישים בעלי ניסיון רב יותר. בנוסף, הוספת האפשרות לבחור את הקטגוריות מעוניין לקחת חלק, צפוי לעודד את המתנדבים להשתתף באירועים באופן פעיל יותר, לשפר את תחושתם, ובכך לתרום להישארותם בארגון לטווח ארוך.

שביעות רצון מבקשי הסיוע-

מדד נוסף שאנו מבקשים לבחון הוא שביעות רצון מבקשי הסיוע מארגון ידידים, מדד איכותני בעל חשיבות רבה לארגון. אחת המטרות המשניות של הארגון היא להרחיב את היכרות הציבור עם פעילותו ולעודד כמה שיותר אזרחים להשתמש בשירותיו. לשם כך, חשוב מאוד לארגון לקבל חוות דעת ממשתמשי השירות. בנינו שאלון שהופץ בקרב משתמשים חוזרים, במטרה לסייע לארגון להתפתח ולהשתפר, ניתוח התשובות יסייע לארגון להעריך את הצורך בשינוי המצב הקיים ובהטמעת המודל החדש.

כיצד היית מעריך את משך הזמן הנדרש לטיפול בפניות?

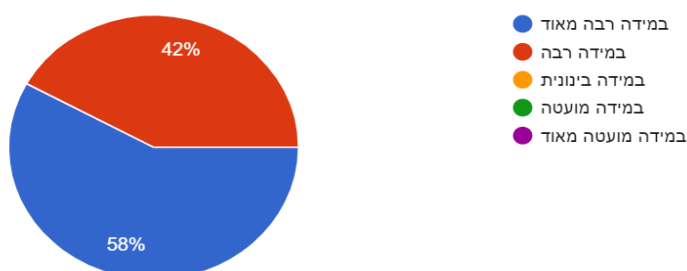
50 תגובות





אם היית יודע שהפנייה שלך תטופל על ידי צוות עם ניסיון בתחום, האם זה היה מגביר את נכונותך להשתמש בשירות שוב?

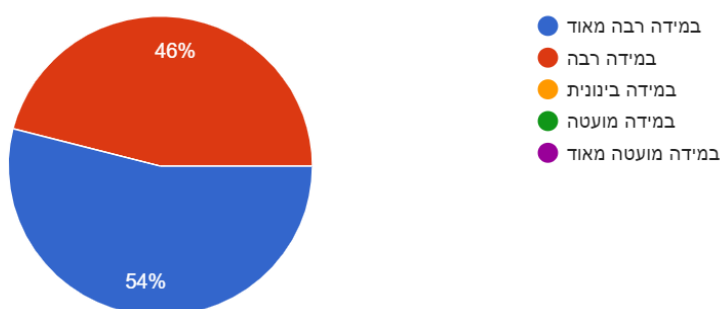
50 תגובות



השאלה נשאלה שנית אך בצורה מעט שונה בכדי לוודא אמינות התוצאות.

אם היית מקבל אפשרות למעקב אחרי סטטוס הפנייה שלך בזמן אמת, האם זה היה משפר את חוויית השימוש שלך באפליקציה?

50 תגובות



7.2 המצב הרצוי לאחר ההטמעה

הניתוח המעמיק שבוצע במצב הקיים של פעילות ארגון "ידידים" העלה מספר פערים משמעותיים בתהליכי העבודה הנוכחיים, במיוחד בכל הנוגע לזמני תגובה, התאמת מתנדבים לאירועים, וניצול המשאבים. המדדים שנבחנו, כולל זמן הטיפול הממוצע, אחוזי ההצלחה, וזמינות המתנדבים בשעות השיא, הצביעו על הצורך בשיפור תהליכי השיבוץ וההקצאה של המתנדבים.

בהתאם לכך, הוחלט ליישם מודל אופטימיזציה מתקדם על בסיס האלגוריתם ההונגרי במערכת השיבוץ. מודל זה, הכולל פרמטרים כמו זמינות המתנדב, קרבתו למיקום התקלה, ניסיון קודם בטיפול בבעיות דומות ודירוגים היסטוריים, ישפר את הדיוק והאפקטיביות של תהליך השיבוץ.

אוטומציה ושיפור תהליך השיבוץ: האלגוריתם ההונגרי יחליף את תהליך קבלת ההחלטות הידני הנוכחי, ויבצע את השיבוץ בצורה אוטומטית ומבוססת על קריטריונים אובייקטיביים. תהליך זה יאפשר שיבוץ מהיר, מדויק ואפקטיבי של מתנדבים למשימות, תוך התחשבות במידע זמין על המתנדבים כמו זמינותם ומיקומם בזמן אמת.

שיפור דיוק ההתאמה: באמצעות המודל החדש, המתנדבים המתאימים ביותר מבחינת זמינות, קרבה גיאוגרפית וניסיון יוקצו למשימות בצורה אופטימלית. כתוצאה מכך, זמני התגובה של

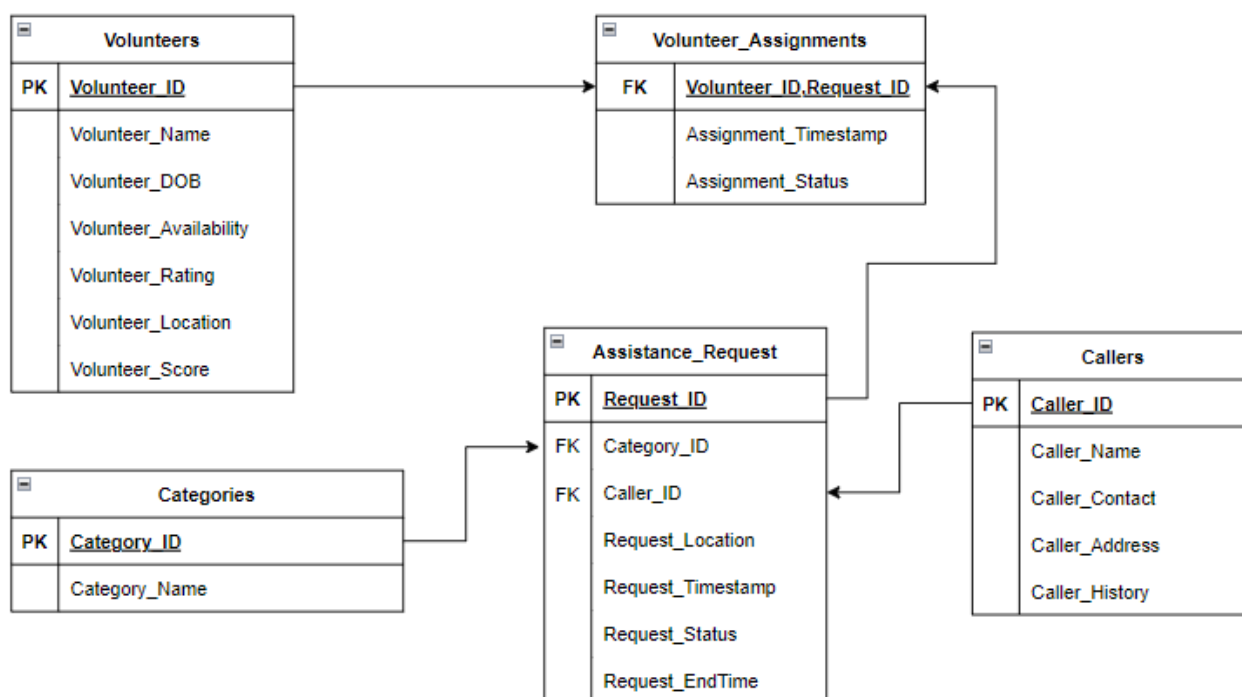


המתנדבים יתקצרו, מה שישפר את איכות השירות הניתן למבקשי הסיוע ויתרום ליעילות גבוהה יותר של המערכת כולה.

יכולת התרחבות: המערכת המשודרגת תוכל להתמודד ביעילות גם עם עומסים מוגברים בקריאות, באמצעות אופטימיזציה של הקצאת המשאבים, כך שגם בתקופות של עומס גבוה תוכל להבטיח מתן שירות ברמה גבוהה ובזמן תגובה קצר.

המצב הרצוי לאחר ההטמעה יבחן על ידי השוואת המדדים שנקבעו מראש, כמו זמן טיפול ממוצע ואחוזי הצלחה, בין המצב הנוכחי למצב המשופר. משוב מהארגון והערכות שוטפות יאפשרו זיהוי של תחומים נוספים לשיפור ויבטיחו שהמערכת החדשה תמשיך להתאים לצרכים המשתנים של הארגון.

לאחר שהוגדר המצב הרצוי והשינויים הצפויים בעקבות הטמעת האלגוריתם ההונגרי, מוצג להלן תרשים ERD המשקף את המבנה החדש של מערכת ניהול המתנדבים וקריאות הסיוע בארגון "ידידים". תרשים זה מדגים את הקשרים בין היישויות המרכזיות במערכת, כגון מתנדבים, קריאות הסיוע, והפונים, ואת אופן ניהול הנתונים במערכת המשופרת. התרשים מציג את השיפורים שנעשו במבנה הנתונים כדי לתמוך באופטימיזציה המוצעת ולהבטיח מענה מהיר ומדויק יותר לקריאות הסיוע המתקבלות.





8. מתודולוגיה

8.1 מבוא למתודולוגיה-

המתודולוגיה בפרויקט זה מתארת את תהליך הפיתוח, הכלים והשיטות שנקטו כדי ליצור מערכת אוטומיזציה אוטומטית להקצאת כווננים בעמותות "ידידים - סיוע בדרכים". הפרויקט מתמקד בפיתוח מערכת מבוססת נתונים היסטוריים שתוכל לקבל החלטות בזמן אמת באמצעות שימוש באלגוריתם ההונגרי ובפרמטרים שנבחרו בתשומת לב כדי לשפר את תהליך הקצאת הכווננים.

יש לציין שפרויקט זה הינו פרויקט מחקרי, ולכן שלבי המתודולוגיה מבוצעים באופן מחזורי ולעיתים במקביל זה לזה. זאת בשל גילוי ממצאים חדשים לאורך כל תהליך הפרויקט אשר ראינו לנכון שיסייעו לנו לבנות את המערכת הנכונה ביותר.

8.2 השגת הנתונים-

כפי שצינו, כלל הנתונים התקבלו ממערכת CiviCRM שמוטמעת במערכות העמותה והובאו לידינו כקובץ Excel מותמם שמציג את אירועי העבר שהיו לעמותה. בכדי להגיע לתוצאה הסופית, ולאחר התייעצות עם מנחה הפרויקט, השתמשנו בנתונים סינתטיים שאותם יצרנו בעצמנו באמצעות Python. ההמלצות הרלוונטיות הועברו לידי העמותה ומוצגות בסוף דו"ח זה. בנוסף, בכדי לקבל מידע על נתונים רלוונטים להשגת מטרת הפרויקט, שוחחנו עם סמנכ"ל העמותה אשר חידד והעשיר את הידע לבחינת נתונים נוספים. במהלך הפרויקט, בוצע שימוש בקובץ נתונים גולמי יחיד שדרכו הגענו לתוצאות הפרויקט. קובץ זה נקרא "קריאות עבר - ידידים" ומכיל בתוכו נתונים אודות מזהה הפונה, כתובת התקלה, שעת פתיחת קריאה, סוג תקלה, מזהה כוון, שעת סגירת תקלה וסטטוס גמר טיפול עבור כל תקלה.

להלן חלק מטבלת הנתונים אותה קיבלנו-

סטטוס אירוע	זמן סגירה	זמן לקיחה	מזהה כוון	מזהה פינה	תת קטגוריה	קטגוריה רכב	כתובת	זמן
נסגר בהצלחה	1/1 18:16	1/1 17:39	גיל 1123	ניסים 9043	כבלים חזקים	הנעה	הנביאים 56, בת ים, ישראל	0 1/1 17:39
נסגר בהצלחה	1/1 18:37	1/1 18:27	שלמה 4278	אבי 0574	פנצ'ר + כלים	פנצ'ר	קדיש לוז 22, קרית מוצקין, ישראל	1 1/1 17:39
נסגר בהצלחה	NaN	1/1 18:28	משה 4838	שמעון 7308	פנצ'ר + כלים	פנצ'ר	דקר 21, לוד, ישראל	2 1/1 17:39
נסגר בהצלחה	NaN	1/1 19:51	גרשון 8697	שמרית 2629	פנצ'ר	פנצ'ר	אבנר בן יהודה 31, נס ציונה, ישראל	3 1/1 17:41
נסגר בהצלחה	1/1 18:06	1/1 17:56	אורי 1940	יגאל 0363	כבלים	הנעה	תחנת עוז נתניה, הרצל 102, נתניה, ישראל	4 1/1 17:43

8.3 עיבוד הנתונים-

שלב עיבוד הנתונים במתודולוגיה הינו השלב שהקצנו לו הכי הרבה זמן בביצוע הפרויקט. הנתונים אותם קיבלנו לא עלו בקנה אחד עם הדרך אותה התווננו להצלחת הפרויקט.

אי לכך, נדרשו פעולות איטרטיביות על הפעולות השונות בעיבוד הנתונים. התוצאה הסופית בשלב עיבוד הנתונים הייתה להפוך טבלת הנתונים שקיבלנו מהעמותה, למספר טבלאות שונות שבחלקן מכילות נתונים סינטטיים, ולכל אחת מהטבלאות יש מרכיב עיקרי וחשוב להצלחת הפרויקט. נציין שבשלב זה של הפרויקט, נתקלנו במספר בעיות בנתונים שכללו: נתונים החורגים מטווח הנורמה, אנומליות בתאריכי פתיחת וסגירת תקלה, נתונים חסרים ועוד. בכדי להתמודד עם אתגר זה, ביצענו מניפולציות שונות על הנתונים בהבנה ותיאום עם מפתח המכיר את אופי המערכת, ובהתאם לחקר והבנת הנתונים, נימוק לכך ניתן למצוא בנספחים. כמו כן, ביצענו Feature Engineering (=תהליך של יצירה, בחירה ושיפור של מאפיינים מנתוני הגלם במטרה לשפר את ביצועי המודלים האלגוריתמיים) בכדי ליצור משתנים חדשים נוספים אשר נבעו ממניפולציות שונות על המשתנים הקיימים.



הערות-

- הנתונים שהוצגו לעיל הן לאחר התממה מצד העמותה וקיבלנו אישור מהגורמים הרלוונטיים להציג בדו"ח זה נתונים אלה.
- סימון עמודות-
*עמודה גולמית- עמודה שלא בוצע עליה שום מניפולציה או שינוי. מציגה את הנתונים כמו שהם התקבלו מהעמותה.
**עמודה שהתווספה: עמודה זו נוצרה מתוך עמודות קיימות בטבלת העמותה, תוך שימוש במניפולציות שונות כדי להציג נתונים בצורה המתאימה למטרות הפרויקט.
***עמודה סינטטית: עמודה שנוצרה לצורך השגת מטרת העל של הפרויקט, ומכילה נתונים רנדומליים שהופקו בעזרת פייתון.

ראשית, נציג את המניפולציות והעבודה שביצענו על הדאטה הגולמית ('קריאות עבר - ידידים'):

- ניקוי שורות עם ערכים חסרים - מחיקת שורות שבהן שני השדות 'זמן סגירה' ו'זמן לקיחה' חסרים לצורך שמירה על איכות הנתונים.
`df.dropna(subset=['זמן לקיחה', 'זמן סגירה'], how='all', inplace=True)`

- המרת עמודות לסוג של זמן - המרת תאריכים בטקסט לפורמט זמן תקני והחלפת שנים שגויות בשנה הרלוונטית להבטחת נתונים תקינים.
`df['זמן לקיחה'] = pd.to_datetime(df['זמן לקיחה'], format='%d/%m %H: %M',
errors='coerce').apply(lambda x: x.replace(year=2022) if x.year == 1900 else x)
df['זמן סגירה'] = pd.to_datetime(df['זמן סגירה'], format='%d/%m %H: %M',
errors='coerce').apply(lambda x: x.replace(year=2022) if x.year == 1900 else x)
df['זמן'] = pd.to_datetime(df['זמן'], format='%d/%m %H: %M',
errors='coerce').apply(lambda x: x.replace(year=2022) if x.year == 1900 else x)`

- חישוב זמן הטיפול עם התחשבות במעברים ליום הבא - חישוב משך זמן הטיפול כולל תיקון במקרים בהם הטיפול נמשך אל היום הבא.
`df['זמן טיפול'] = df.apply(lambda row: row['זמן לקיחה'] - row['זמן סגירה'] if row['זמן סגירה'] >= row['זמן לקיחה'] else row['זמן סגירה'] + pd.Timedelta(days=1) - row['זמן לקיחה'], axis=1)`

- איתור ערכים שליליים בעמודות זמן טיפול - זיהוי ומחיקת שורות עם ערכים שליליים בזמן הטיפול כדי להסיר נתונים שגויים.
`negative_treatment_time = df[df['זמן טיפול'] < pd.Timedelta(0)]
df = df[~df.index.isin(negative_treatment_time.index)]`

- זיהוי ערכים שליליים ושינוי שנת 'זמן סגירה' - תיקון ערכי זמן סגירה במקרים של חישובי זמן טיפול שגויים, על ידי הוספת שנה לערך השגוי.
`df.loc[negative_values_mask, 'זמן סגירה'] += pd.DateOffset(years=1)`

- חישוב הזמן הממוצע לתקלה לפי הקטגוריה - חישוב זמן הטיפול הממוצע עבור כל קטגוריה, לשימוש בהשלמת נתונים חסרים.
`average_time_per_category = df.groupby('קטגוריה').apply(lambda x: (x['זמן סגירה'] - x['זמן לקיחה']).mean())`

- הוספת זמן סגירה ממוצע לשורות חסרות - השלמת זמן סגירה חסר באמצעות הזמן הממוצע בקטגוריה המתאימה.
`def add_average_time_close(row):`



```
if pd.isna(row['זמן סגירה']):
    new_closure_time_close = row['זמן לקיחה'] +
average_time_per_category[row['קטגוריה']]
    return new_closure_time_close
return row['זמן סגירה']
df['זמן סגירה'] = df.apply(add_average_time_close, axis=1)
```

- הוספת זמן לקיחה ממוצע לשורות חסרות - השלמת זמן לקיחה חסר בהתאם לזמן הממוצע בקטגוריה המתאימה.

```
def add_average_time_open(row):
    if pd.isna(row['זמן לקיחה']):
        new_closure_time_open = row['זמן סגירה'] -
average_time_per_category[row['קטגוריה']]
        return new_closure_time_open
    return row['זמן לקיחה']
df['זמן לקיחה'] = df.apply(add_average_time_open, axis=1)
```

- חישוב מחדש של זמן הטיפול - חישוב מחדש של זמן הטיפול לאחר השלמת הזמנים החסרים.
df['זמן טיפול'] = df.apply(lambda row: row['זמן סגירה'] - row['זמן לקיחה'] if row['זמן לקיחה'] >= row['זמן סגירה'] else row['זמן סגירה'] + pd.Timedelta(days=1) - row['זמן לקיחה'], axis=1)

- המרת זמן טיפול לדקות - המרת זמן הטיפול למספר דקות, כדי להקל על ניתוחים סטטיסטיים.

```
if isinstance(td, pd.Timedelta):
    return td.total_seconds() / 60
else:
    return td # Return the original value if it's not a timedelta

# המרת עמודת 'ממוצע זמן טיפול בתקלה' לשעות
df['זמן טיפול'] = df['זמן טיפול'].apply(convert_to_minutes)

df.groupby('קטגוריה')['זמן טיפול'].transform('mean')
```

- חישוב זמן טיפול ממוצע לכל קטגוריה - חישוב ממוצע זמן הטיפול לכל קטגוריה, לטובת ניתוח ביצועים ושיפור תהליכים.

```
df['זמן טיפול ממוצע'] = df.groupby('קטגוריה')['זמן טיפול'].transform('mean')
```

- בדיקת זמן טיפול חריג וסינון ערכים חריגים - זיהוי וסינון של מקרים חריגים, שבהם זמן הטיפול גבוה בצורה לא פרופורציונלית.

```
# פונקציה לבדיקת זמן טיפול
def check_treatment_time(row):
    if pd.isna(row['זמן טיפול']) or pd.isna(row['זמן טיפול ממוצע']): # Check for NaT values
        return False
    if row['זמן טיפול'] > 10 * row['זמן טיפול ממוצע']:
        return True
```




```
return False
```

```
# החלת הפונקציה על כל שורה ובדיקת התנאי
```

```
df['זמן טיפול'] = df.apply(check_treatment_time, axis=1)
```

```
# סינון השורות שבהן זמן הטיפול חריג
```

```
outliers_df = df[df['זמן טיפול'] == True]
```

```
df = df[df['זמן טיפול'] == False]
```

- הוספת דירוג לקוח רנדומלי - יצירת עמודת דירוג לקוח רנדומלית בין 1 ל-5, לשימוש בניתוחים סטטיסטיים נוספים.

```
df['דירוג לקוח'] = np.random.randint(1, 6, size=len(df))
```

כעת נציג את הטבלאות שיצרנו במסגרת פרויקט זה אשר נגזרות מהטבלה הגולמית:
א. טבלת 'מאגר כוננים' - לצורך הטמעה נכונה של האלגוריתם ההונגרי, היה צורך בטבלה שתאגד את פרטי הכוננים שיצאו לכל הפחות לקריאה אחת ונכנסו למאגר העמותה. טבלה זו מאגדת את המידע הבסיסי ביותר (באפשרות העמותה להרחיב את הטבלה, כראות עיניה ולהוסיף פרטים מזהים נוספים כמו מספר טלפון, כתובת מגורים, מצב משפחתי וכו'. עקב חסיון העמותה, לנו לא הייתה גישה לפרטים אלה). מטרת הטבלה היא לאגד בתוכה את המידע אודות המתנדב, הנתונים אותם היא מכילה הוזנו במשוואה עליה נפרט בהמשך.
הטבלה מכילה את העמודות הבאות:

- 'מזהה כונו' - עמודת מפתח.
- 'קטגוריה' - עמודת מפתח. מציגה את סוג התקלה.
- 'ממוצע זמן טיפול בתקלה' - עמודה זו מציגה את הזמן טיפול בדקות בתקלה של כונו מסוים עבור סוג תקלה מסוימת. לכן יש לנו שני מפתחות בטבלה זו.
- 'כמות אירועים' - סך כל הקריאות אליהן יצא כונו מסוים עבור סוג תקלה מסוימת.
- 'אירועים מוצלחים' - סך כל הקריאות שהסתיימו בהצלחה מתוך כלל הקריאות מסוג מסוים אליהן יצא הכונו.
- 'אחוזי הצלחה' - עמודה שמסתמכת על העמודות 'כמות אירועים' ו'אירועים מוצלחים' ומחשבת את שיעור ההצלחה של כל מתנדב בקריאות אליהן הוא יצא עבור תקלה מסוימת.
- 'דירוג לקוח' - עמודה שמציגה את הדירוג הממוצע שקיבל המתנדב מכלל מבקשי העזרה שסייע להם.

ממוצע דירוג לקוח	אחוזי הצלחה	אירועים מוצלחים	כמות אירועים	ממוצע זמן טיפול בתקלה	קטגוריה	מזהה כונו
2.000000	100.000000	1.0	1	13.000000	הנעה	אבי 0000
2.500000	100.000000	2.0	2	9.000000	הנעה	אביחי 0000
2.000000	0.000000	0.0	1	18.046653	הנעה	אביטל 0000
3.094595	86.956522	60.0	69	16.069465	הנעה	ישראל 0000
3.094595	100.000000	3.0	3	37.430566	פנצ'ר	ישראל 0000



להלן הקוד לבניית הטבלה -
*הערה- טבלה זו נבנתה כנגזרת מהטבלה הגולמית לאחר השינויים שביצענו עליה.

```
# שינוי ערכים בעמודה סטטוס אירוע למספרים על פי התנאים הנתונים
status_mapping = {
    'ניסגר בהצלחה': 1,
    'ניסגר ללא הצלחה': 0,
    'בוטל': 0,
    'הוחזר': 2
}
df['סטטוס אירוע'] = df['סטטוס אירוע'].map(status_mapping)

# יצירת עמודה חדשה 'הצלחה' על פי הערכים בעמודה 'סטטוס אירוע'
success_mapping = {
    1: 1,
    0: 0,
    2: 0
}
df['הצלחה'] = df['סטטוס אירוע'].map(success_mapping)

# פונקציה לחישוב ממוצע זמן טיפול
def calculate_average_time(group):
    total_time = group.sum()
    average_time = total_time / len(group)
    return average_time

# קביעת הגישה לקבוצות לפי מזהה כונן וקטגוריה
grouped = df.groupby(['מזהה כונן', 'קטגוריה'])

# חישוב ממוצע הזמן לטיפול בתקלה לכל קטגוריה וכונן
average_time = grouped['זמן טיפול'].apply(calculate_average_time)

# חישוב כמות האירועים והאירועים המוצלחים
event_counts = df.groupby(['מזהה כונן', 'קטגוריה']).size().reset_index(name='כמות אירועים')
successful_events = df[df['סטטוס אירוע'] == 1].groupby(['מזהה כונן', 'קטגוריה']).size().reset_index(name='אירועים מוצלחים')

# 1 פילטר על הערכים 'הצלחה' 0 או 1
filtered_df = df[df['הצלחה'].isin([0, 1])]

# חישוב אחוזי ההצלחה של כל כונן עבור כל קטגוריה
success_rates = filtered_df.groupby(['מזהה כונן', 'קטגוריה']).agg(
    סך_הכל_אירועים=pd.NamedAgg(column='הצלחה', aggfunc='count'),
    סך_הכל_הצלחות=pd.NamedAgg(column='הצלחה', aggfunc='sum')
).reset_index()
```



```
# הוספת עמודה של אחוזי הצלחה
success_rates['אחוזי הצלחה'] = (success_rates['יסך_הכל_הצלחות'] /
success_rates['יסך_הכל_אירועים']) * 100

# חישוב ממוצע הדירוג לקוח לכל כוץ באופן גלובלי
global_average_rating = df.groupby('מזהה כוץ')['דירוג לקוח'].mean().reset_index(name='ממוצע
דירוג לקוח')

# סופי לפי הדרישות DataFrame בניית
final_df = pd.DataFrame({'ממוצע זמן טיפול בתקלה': average_time}).reset_index()

# מיזוג עם נתוני כמות האירועים והאירועים המוצלחים
final_df = final_df.merge(event_counts, on=['מזהה כוץ', 'קטגוריה'], how='left')
final_df = final_df.merge(successful_events, on=['מזהה כוץ', 'קטגוריה'], how='left')

# מיזוג עם נתוני אחוזי ההצלחה
final_df = final_df.merge(success_rates[['מזהה כוץ', 'קטגוריה', 'אחוזי הצלחה']], on=['מזהה כוץ',
'קטגוריה'], how='left')

# מיזוג עם ממוצע דירוג לקוח גלובלי
final_df = final_df.merge(global_average_rating, on=['מזהה כוץ'], how='left')
```

- ב. טבלת 'זמינות כוץ', מיקומם וציון סופי - טבלה זו נבנתה גם היא מתוך הצורך לייעל את הפצתה של קריאה חדשה שנכנסת למערכת. כפי שצוין, קריאה חדשה מופצת לכלל הכוץ שמוזנים במאגר העמותה מבלי לקחת בחשבון את מצב הזמינות שלהם נכון לרגע קבלת הקריאה. בנוסף, הוספנו (בדרך סינטטית) את מיקומם של הכוץ דבר שישיר את אותנו בהמשך בניית האלגוריתם כדי למצוא את המרחק בין הכוץ למיקום הקריאה. כמו כן, טבלה זו הינה אחת משתי הטבלאות שנכנסות לאלגוריתם ההונגרי אותו בנינו ומסייעת לנו להגיע אל המתנדב האופטימלי. להלן העמודות שמופיעות בטבלה ופירוט עליהן:
- 'מזהה כוץ' - עמודת מפתח.
 - 'קטגוריה' - עמודת מפתח, מציגה בתוכה את סוג התקלה. גם כאן בחרנו להוסיף את העמודה הזאת מתוך רצון לתת גמישות למתנדב לבחור באילו סוגי תקלות הוא יכול לסייע באותו רגע נתון ולא לחייב אותו לתת מענה לכל סוגי התקלות.
 - 'ציון סופי' - הציון אותו קיבל המתנדב מהשוואה אותה בנינו. (פירוט אודות המשוואה והפרמטרים שהיא מכילה בהמשך הסעיף).
 - 'זמינות' - מצב עדכני אודות הזמינות של המתנדב עבור קטגוריה מסוימת.
 - 'מיקום מתנדב' - מיקומו העדכני של המתנדב.

מיקום כוץ	זמינות	final score	קטגוריה	מזהה כוץ
31.100727938063795,34.88118528215465	זמין	7.285183	שמן-מים-דלק	9930 דוד
33.062332214096905,35.60478424234899	זמין	17.855717	שמן-מים-דלק	9930 עדי
None	לא זמין	10.142084	שמן-מים-דלק	9933 יהודה
31.57131210410382,34.91123539517559	זמין	17.284337	שמן-מים-דלק	9936 עמרם
None	לא זמין	8.713633	שמן-מים-דלק	9941 בנצי
None	לא זמין	7.570873	שמן-מים-דלק	9945 לאון



```
להלן הקוד שבעזרתו בנינו את העמודות הסינטטיות 'מיקום מתנדב' ו'זמינות'
# יצירת עמודה רנדומלית 'זמינות' עם ערכים 'זמין' או 'לא זמין'
calling_pool['זמינות'] = [random.choice(['זמין', 'לא זמין']) for _ in range(len(calling_pool))]

# פונקציה ליצירת קורדינטות רנדומליות בשטחי ישראל
def random_coordinates_in_israel():
    lat = random.uniform(29.5, 33.3) # תחום קווי רוחב של ישראל
    lon = random.uniform(34.2, 35.9) # תחום קווי אורך של ישראל
    return f'{{lat}},{{lon}}'

# הוספת עמודת 'מיקום כוץ' על פי העמודה 'זמינות'
calling_pool['מיקום כוץ'] = calling_pool['זמינות'].apply(
    lambda x: random_coordinates_in_israel() if x == 'זמין' else None)
```

ראינו לנכון לתת חשיבות למגבלה שבכל רגע נתון חלק מהכוונים לא יהיו במצב 'זמין', ולכן גם כאן החלטנו להוסיף את הפיצ'ר של זמינות המתנדב כדי להציג תמונת מצב מציאותית ככל שניתן. כאשר כוץ מעוניין לסייע, כל שעליו לעשות הוא להיכנס לאפליקציה 'ידידים - כוץ' שכבר קיימת עבורם, וללחוץ על הפיצ'ר 'זמין' והמערכת תדע לקחת אותו בחשבון בכל הקריאות הבאות שיכנסו למערכת עד אשר הוא יסמן את עצמו כ'לא זמין'.

נוכח סינון המתנדבים עקב זמינותם, המודל רץ על כמות נתונים מופחתת המובילה לזמן ריצה קצר יותר והורדת עומס על המערכת.

*הערה- הוספת פיצ'ר זה לאפליקציית הכוונים היא באחריות העמותה ולא נכללת תחת מסגרת האחריות של פרויקט זה. המלצה זו ועוד אחרות הועברו לעמותה ונרשמו בסעיף המתאים לכך.

כמו כן, שמנו דגש שמיקום המתנדב יהיה מציאותי ככל שניתן, ולכן השתמשנו בספרייה מובנת בפיתוח בשם geodesic שאיפשרה לנו להציג קורדינטות בשטחי ישראל בלבד. הסיבה לכך שבחרנו להשתמש בספרייה זו ולא בדרך משוכללת יותר שנועדה לנטר מיקום מדויק כגון API היא חוסר במשאבים כלכליים הן בקרב העמותה והן בקרב הפרויקט שלנו.

ניתן לראות שכונן אשר מוגדר כ'לא זמין' לא נקבל את המיקום שלו מתוך חשש לחדירה לפרטיות. רק כאשר כוץ יגדיר את עצמו כ'זמין' יתקבל מיקומו.

ג. טבלת 'קריאות חדשות' - טבלה שמציגה את הקריאות החדשות שמתקבלות הן באפליקצית 'ידידים - סיוע' והן במוקד הטלפוני.

הטבלה מכילה את העמודות הבאות-

- 'קטגוריה'
- 'מזהה פונה'
- 'מיקום תקלה'*** - הצגת מיקום התקלה בעזרת קורדינטות.

מיקום תקלה	מזהה פונה	קטגוריה
31.383973270650856,34.39726043737276	עדי 4884	הנעה
30.190631027438513,35.67907162861701	סתיו 1169	רכב נעול
30.366574080734456,35.81251613968826	איילה 7448	פנצ'ר
32.379704187394,34.54707136346327	אוריאל 0059	הנעה
30.552261608725885,35.545364757747016	משה 8481	הנעה
32.71340317373189,34.7634565541651	ישראל 1336	פנצ'ר



בדומה לעמודה 'מיקום כונו', גם כאן השתמשנו בקוד בעזרת פונקציית geodesic.

ד. טבלת 'ציון כונונים' - מהות הטבלה היא הצגת הציון הסופי אותו קיבל המתנדב בעבור הפרמטרים השונים עליהם התבקשו לשים דגש בבניית המודל.

עמודות הטבלה -

- 'מזהה כונו' - מפתח.
- 'קטגוריה' - מפתח.
- 'ממוצע זמן טיפול בתקלה' - ממוצע זמן טיפול בתקלה של מתנדב מסוים עבור סוג קטגוריה ספציפית.
- 'כמות אירועים' - סך האירועים מסוג מסוים בהם השתתף הכונו.
- 'אירועים מוצלחים' - סך האירועים שהסתיימו בהצלחה מתוך כלל האירועים מסוג מסוים אליהם יצא הכונו.
- 'אחוזי הצלחה' - עמודה שמסתמכת על העמודות 'כמות אירועים' ו'אירועים מוצלחים' ומחשבת את שיעור ההצלחה של כל מתנדב בקריאות אליהן הוא יצא עבור תקלה מסוימת.
- 'ממוצע דירוג לקוח' - עמודה שמבוססת על עמודות 'דירוג לקוח' שנמצאת בטבלת 'מאגר כונונים' ומכילה בתוכה את הממוצע כלל דירוגי הלקוחות שקיבל הכונו עבור סיוע התקלה מסוג מסוים.
- העמודות - t_ratio, r_ratio, w_ratio, t_score, r_score, w_score (פירוט מורחב על בניית המשוואה - בהמשך הסעיף)

מזהה כונו	קטגוריה	ממוצע זמן טיפול בתקלה	כמות אירועים	אירועים מוצלחים	אחוזי הצלחה	ממוצע דירוג לקוח	t_ratio	r_ratio	w_ratio	t_score	r_score	w_score	rank	final score
0001 שלמה	אחר	24.000000	1	1.0	100.0	2.944444	0.975470	0.980425	1.159445	3	3	4	2.356700	10.182565
0003 שלומי	אחר	14.000000	1	1.0	100.0	3.250000	1.672234	1.082168	1.159445	5	4	4	3.119162	14.574226
0004 אליהו	אחר	1.000000	1	1.0	100.0	3.454545	23.411283	1.150276	1.159445	5	4	4	3.119162	14.574226
0009 דדי	אחר	25.280388	1	1.0	100.0	1.000000	0.926065	0.332975	1.159445	3	1	4	2.218071	9.384081
0009 יוסי	אחר	12.640194	6	5.0	100.0	2.869565	1.852130	0.955493	1.159445	5	3	4	8.562005	45.924147

קוד לבניית הטבלה-

חישוב הממוצעים של כל עמודה לפי קטגוריות עבור העמודות הרלוונטיות בלבד

```
columns_to_average = ['ממוצע זמן טיפול בתקלה', 'ממוצע דירוג לקוח', 'אחוזי הצלחה', 'ממוצע זמן טיפול בתקלה']
```

```
averages = user_data_df.groupby('קטגוריה')[columns_to_average].transform('mean')
```

חישוב היחסים בין כל ערך לממוצע של העמודה המתאימה בקטגוריה

הערה: בשביל ממוצע זמן טיפול בתקלה, נשתמש ביחס הפוך

```
user_data_df['t_ratio'] = averages['ממוצע זמן טיפול בתקלה'] / user_data_df['ממוצע זמן טיפול בתקלה']
```



```
user_data_df['r_ratio'] = user_data_df['ממוצע דירוג לקוח'] / averages['ממוצע דירוג לקוח']

user_data_df['w_ratio'] = user_data_df['ממוצע הצלחה'] / averages['ממוצע הצלחה']

# מתן ציונים לפי הסולם שניתן

def score(value):

    if 0 <= value <= 0.4:

        return 1

    elif 0.4 < value <= 0.7:

        return 2

    elif 0.7 < value <= 1:

        return 3

    elif 1 < value <= 1.3:

        return 4

    else:

        return 5

# לשיפור היעילות vectorization - שימוש ב

score_vectorized = np.vectorize(score)

user_data_df['t_score'] = score_vectorized(user_data_df['t_ratio'])

user_data_df['r_score'] = score_vectorized(user_data_df['r_ratio'])

user_data_df['w_score'] = score_vectorized(user_data_df['w_ratio'])
```

עקרונות בניית הקוד-

1. כדי להתחשב במספר האירועים בהם המתנדב השתתף בכל קטגוריה, הוספנו משקל נוסף לציון שניתן לכל מתנדב בהתאם למספר האירועים בהם השתתף. בקוד זה השתמשנו בעמודת מספר האירועים כדי להשפיע על הציון הכולל כך שמתנדב שהשתתף במספר גדול יותר של אירועים יקבל יתרון מסוים על פני מתנדב שהשתתף במספר קטן יותר של אירועים. הוספנו את העמודה הזו לחישוב הציונים והשתמשנו בנוסחה החדשה שתתחשב במספר האירועים. נוספה למשוואה פונקציית הלוגריתם הטבעי `np.log1p`, כדי לתת משקל נוסף לציון בהתאם למספר האירועים (עם שיפור לוגריתמי). פונקציה זו מאפשרת להפחית את ההשפעה של משתתפים במספר מועט של אירועים, מבלי לפגוע במידה רבה במשתתפים רבים.

2. נרמול- כדי לבצע נרמול עבור כל קטגוריה בנפרד, עלינו חושב הציון המנורמל לכל קטגוריה בנפרד והתווספו הערכים המנורמלים לעמודה `'normalized_rank'`.



3. כדי להתחשב בעובדה שזמן טיפול גבוה יותר צריך להפחית את הציון הכולל, ביצענו שינוי בכיוון של החישוב של t_ratio . והשתמשנו ביחס הפוך עבור t_ratio כך שמתנדב שזמן הטיפול שלו גבוה יותר יקבל ציון נמוך יותר בפרמטר הזה.

8.4 הכנה לבניית האלגוריתם ההונגרי-

בכדי למצוא את הכון האופטימלי עבור תקלה חדשה שנכנסת למערכת, נדרשנו להזין לאלגוריתם ההונגרי נתונים מספריים. בתחילת התהליך העבודה שלנו, התייעצנו עם גורמים בכירים בעמותה וביקשנו מהם להציג בפנינו את הפרמטרים אותם הם היו מעוניינים להביא לידי ביטוי בתהליך מציאת הכון. בשיחה זו הובהר לנו שבראש ובראשונה הם רוצים לקצר לשפר את זמני הטיפול בתקלה ובנוסף הם רוצים לתת דגש על חווית המשתמש ואיכות השירות אף על פי שמדובר בעמותה והסיוע של הכוננים נעשה ללא רווחים כאלה או אחרים עדין העמותה מעוניינת לשפר את איכות השירות שלה.

לאחר חשיבה ממוקדת, החלטנו לפצל את הנתונים לשני חלקים.

1. נתונים אודות הכון - יחושב בעזרת נוסחא.
2. מרחק פיזי בין הכון לתקלה

8.4.1 נוסחא לחישוב ציון המתנדב-

לפי הצורך של העמותה בכל הקשור לחשיבות הפרמטרים העיקריים אותם הגדירו לנו מנהלי העמותה, בנינו משוואה המציגה לנו ציון המתנדב קיבל על סמך אירועי העבר של אותו מתנדב. האלגוריתם יודע לעדכן את הציון של המתנדבים כאשר הם יוצאים לתקלות חדשות ובכך אנחנו מבטיחות תמונת מצב עדכנית ואמינה.

ע"מ לבנות את המשוואה הנכונה ביותר שמדרגת את דירוג המתנדב, נחלק את המשוואה לשלושה פרמטרים שלכל אחד מהפרמטרים האלה יש חשיבות רבה בעיני העמותה, וברצונם שתחומים אלה יילקחו בחשבון בעת בחירת המתנדב האופטימלי עבור שלל הקריאות. הפרמטרים שיבואו לידי ביטוי במשוואה –

- **ממוצע זמן טיפול בתקלה –** היעד העיקרי בעמותה הוא לצמצם את זמני הטיפול בתקלה מרגע כניסתה של קריאה חדשה במערכת ועד לסגירתה. לכן, פרמטר זה יקבל את המשקל הגבוה ביותר במשוואה.
- **דירוג הלקוחות –** דגש נוסף שהעמותה מעוניינת לשים אליו לב הוא איכות השירות של המתנדב.
- **אחוזי הצלחה –** מקצועית. העמותה מעוניינת שהכוננים שלה יהיו מקצועיים ככל שניתן ולכן נבחן אחוזי ההצלחה שלהם בקריאות השונות.

לכל אחד מהפרמטרים נבנה משוואה שמציגה מספר מסוים, אותו נמיר בדירוג (1-5). ההמרה לדירוג מתבצעת באמצעות מד מספרים רציף שמתחיל מהערך המינימלי של אותו הפרמטר עד לערך המקסימלי, נחלק את רצף הערכים האלה ל-5 וכל ערך יקבל דירוג מ-1-5. כך בעצם נקבל את הציון של המתנדב ביחס ליתר המתנדבים. את הדירוג שקיבלנו נכפול במשקל אשר מציג את הכובד של אותו פרמטר במשוואה הסופית. משקלי הפרמטר הוגדרו ע"י העמותה בהתאם לחשיבות הפרמטרים. ככל שהפרמטר חשוב יותר, הוא יקבל משקל גבוה יותר ולחלופין. את המשקל הזה יהיה ניתן לשנות בהתאם לצורכי העמותה המשתנים.

ממוצע זמן טיפול בתקלה-

נגדיר-

$$\frac{ti}{\bar{t}} = t_ratio$$

ti - ממוצע זמן טיפול בתקלה I עבור מתנדב J

\bar{t} - ממוצע הממוצעים של זמני טיפול בתקלה I עבור כל המתנדבים
נקבל את המשוואה-



נגדיר מד T שבעזרתו נוכל לתת דירוג למתנדב עבור הפרמטר 'זמן טיפול בתקלה'. בצד הימני יש את הדירוג אותו מקבל המתנדב עבור פרמטר זה.

$0 < t_ratio \leq 0.4$	->	1
$0.4 < t_ratio \leq 0.7$	->	2
$0.7 < t_ratio \leq 1$	->	3
$1 < t_ratio \leq 1.3$	->	4
$1.3 < t_ratio$	->	5

T_score - המשקל אותו יקבל הפרמטר 'זמן טיפול בתקלה' בציון הסופי של דירוג המתנדב. ציון זה יכנס למשוואה הסופית.

$$\frac{r_i}{\bar{r}} = r_ratio$$

דירוג לקוחות -
r_i - ממוצע דירוג לקוחות עבור מתנדב J לתקלות I.
r̄ - ממוצע הממוצעים עבור דירוגי לקוחות.
נקבל את המשוואה-

גם כאן נגדיר מד r_ratio שבעזרתו נוכל לתת דירוג למתנדב עבור הפרמטר 'דירוג לקוחות'.

$0 < r_ratio \leq 0.4$	->	1
$0.4 < r_ratio \leq 0.7$	->	2
$0.7 < r_ratio \leq 1$	->	3
$1 < r_ratio \leq 1.3$	->	4
$1.3 < r_ratio$	->	5

r_score - המשקל אותו יקבל הפרמטר 'ממוצע דירוג לקוח' בציון הסופי של דירוג המתנדב. ציון זה יכנס למשוואה הסופית.

$$\frac{w_i}{\bar{w}} = W_ratio$$

אחוזי הצלחה -
נגדיר -
w_i - אחוז האירועים i שהסתיימו בהצלחה עבור מתנדב j.
w̄ - ממוצע אחוזי הצלחה.
נקבל את המשוואה-

גם כאן נגדיר מד W_ratio שבעזרתו נוכל לתת דירוג למתנדב עבור הפרמטר 'אחוזי הצלחה'.



$0 < W_ratio \leq 0.4$	->	1
$0.4 < W_ratio \leq 0.7$	->	2
$0.7 < W_ratio \leq 1$	->	3
$1 < W_ratio \leq 1.3$	->	4
$1.3 < W_ratio$	->	5

ונגדיר w_score - המשקל אותו יקבל הפרמטר 'אחוזי הצלחה' בציון הסופי של דירוג המתנדב.

לאחר שהגדרנו את שלושת הפרמטרים שיהוו חלק במשוואה הסופית שמטרתה לתת לנו דירוג לקוח, נקבל את המשוואה-

$$\text{Final score} = t_score * 0.5 + r_score * 0.1 + w_score * 0.4$$

להלן הקוד לבניית הנוסחא ונרמול הציון שהתקבל-

```
# חישוב הציון הכולל של כל מתנדב לפי המשוואה
user_data_df['rank'] = (user_data_df['t_score'] * 0.5 + user_data_df['r_score'] * 0.1 +
user_data_df['w_score'] * 0.4) * np.log1p(user_data_df['כמות אירועים'])

# נרמול הציונים בין 1 ל-100 עבור כל קטגוריה בנפרד
def normalize_category(group):
    min_rank = group['rank'].min()
    max_rank = group['rank'].max()
    group['final score'] = 1 + ((group['rank'] - min_rank) * (100 - 1) / (max_rank -
min_rank))
    return group
```

8.4.2 מציאת מרחק פיזי בין הכוון לתקלה-

בעזרת העמודה 'מיקום כוון' שנמצאת בטלה 'זמינות הכוונים', מיקומם וציון סופי' והעמודה 'מיקום תקלה' שנמצאת בטבלה 'קריאות מתעדכנת' נקבל את המרחק המדויק בין השניים.

להלן הקוד למציאת המרחק בין שני המיקומים-

```
for category, call_df in calling_by_category:
    if category not in volunteer_by_category.groups:
        continue

    vol_df = volunteer_by_category.get_group(category)
```



```
call_df.loc[:, 'מיקום תקלה'] = call_df['מיקום מתנדב'].apply(eval)
vol_df.loc[:, 'מיקום מתנדב'] = vol_df['מיקום מתנדב'].apply(eval)
```

8.5 בניית האלגוריתם ההונגרי-

כחלק מתהליך אופטימיזציה שיבוץ המתנדבים לבקשות, הבעיה הוגדרה כמטריצת עלויות, שבה כל שורה מייצגת בקשה לסיוע וכל עמודה מייצגת מתנדב זמין. האלגוריתם ההונגרי נבחר לפתרון בעיית השיבוץ האופטימלית, תוך מינימום העלות הכוללת (הכוללת את המרחק בין המתנדב לבקשה ואת ציון המתנדב מהמשוואה).

8.5.1 תהליך יישום האלגוריתם

האלגוריתם ההונגרי יושם בשלבים הבאים:

א. הכנת הנתונים:

1. פונקציה לנרמול ציונים-

תחילה, נבנתה פונקציה לנרמול ציוני המתנדבים, כך שציון גבוה יהפוך לעלות נמוכה:

```
def normalize_scores(scores):
```

```
    max_score = max(scores)
```

```
    return max_score - scores
```

2. הגרלת נתונים רנדומליים-

בהמשך, נבחר מספר רנדומלי של בקשות מתוך טבלת 'calling_pool', ולאחר מכן בוצע פיצול של הטבלה לפי קטגוריות הבקשות. בחלק זה רצינו לדמות סיטואציות מהמציאות שבכל רגע נתון נכנסות מספר שונה של קריאות חדשות למערכת. היה לנו חשוב שהמודל לא יעבוד רק בסביבת עבודה 'סטריילית' וב'תנאי מעבדה' שבכל רגע יש אך ורק קריאה אחת כי במציאות זה לא קורה.
num_rows = random.randint(1, 20)

3. סינון מתנדבים זמינים-

מתוך טבלת המתנדבים ('volunteer_pool'), סוננו רק המתנדבים שסומנו כזמינים. המתנדבים סווגו לקטגוריות, כך שניתן יהיה לשייך אותם לבקשות באותה קטגוריה. גם חלק זה נועד כדי לדמות מצב אמת שבו לא כל הכוונים תמיד זמינים.

```
available_volunteers = Volunteer_pool[Volunteer_pool['זמינות'] == 'זמין']
```

ב. חישוב מטריצת העלויות:

4. חישוב המרחקים-

חישוב המרחק בין כל מיקום של תקלה למיקום של מתנדב בוצע באמצעות פונקציית 'geodesic', שהחזירה את המרחק בקילומטרים.

```
results = []
```

```
for category, call_df in calling_by_category:
```

```
    if category not in volunteer_by_category.groups:
```

```
        continue
```

```
    vol_df = volunteer_by_category.get_group(category)
```

```
    call_df.loc[:, 'מיקום תקלה'] = call_df['מיקום מתנדב'].apply(eval)
```

```
    vol_df.loc[:, 'מיקום מתנדב'] = vol_df['מיקום מתנדב'].apply(eval)
```



5. בניית מטריצת עלויות-

לכל קטגוריה, נבנתה מטריצת עלויות המבוססת על המרחקים בין מיקומי התקלות למיקומי המתנדבים, כשהציונים המשוקללים מהנוסחא של המתנדבים נוספו לכל מרחק, לאחר נרמול הציונים.

```
cost_matrix = []
for call_index, call_row in call_df.iterrows():
    row = []
    call_location = call_row['מיקום תקלה']
    for vol_index, vol_row in vol_df.iterrows():
        vol_location = vol_row['מיקום מתנדב']
        distance = geodesic(call_location, vol_location).kilometers
        row.append(distance)
    cost_matrix.append(row)

cost_matrix = np.array(cost_matrix)
```

6. וידוא ריבועיות המטריצה-

כדי לאפשר יישום תקין של האלגוריתם ההונגרי, היה צורך להבטיח שמטריצת העלויות תהיה ריבועית (מספר השורות והעמודות שווה). לכן, נוספו שורות ועמודות עם עלויות גבוהות מאוד (dummy) כדי להשלים את המטריצה לריבועית במידת הצורך. העלויות שהתווספו היו בעלות ערך גבוה מאוד בכדי להבטיח שהם לא יבחרו (האלגוריתם מחפש את הערך המינימלי).

```
n, m = cost_matrix.shape
max_dim = max(n, m)
if n < max_dim:
    cost_matrix = np.vstack([cost_matrix, np.full((max_dim - n, m), 1e6)])
if m < max_dim:
    cost_matrix = np.hstack([cost_matrix, np.full((max_dim, max_dim - m), 1e6)])
```

ג. יישום האלגוריתם ההונגרי:

7. הרצת האלגוריתם ההונגרי-

לאחר בניית מטריצת העלויות עבור כל קטגוריה, יושם האלגוריתם ההונגרי באמצעות הפונקציה 'linear_sum_assignment' ממודול 'scipy.optimize'. הפונקציה החזירה את ההתאמות האופטימליות בין המתנדבים לבקשות.

```
row_ind, col_ind = linear_sum_assignment(cost_matrix)
```

8. שיבוץ המתנדבים:

התוצאות של האלגוריתם הצביעו על השיבוץ האופטימלי של מתנדבים לבקשות תוך התחשבות בעלות המינימלית. שיבוצים אלה נשמרו בטבלה מסכמת.

ד. הצגת התוצאות

```
assignment = [(category, call_df.iloc[row]['מזהה פונה'], vol_df.iloc[col]['מזהה כונית'],
cost_matrix[row, col]) for row, col in zip(row_ind, col_ind) if row < len(call_df) and col < len(vol_df)]
results.extend(assignment)
```



```
results_df = pd.DataFrame(results, columns=['קטגוריה', 'מזהה כונו', 'מזהה פונה', 'קטגוריה'])
```

9. תוצגות תוצאות:

כל השיבוצים האופטימליים הוצגו בטבלה 'results_df', שכללה את הקטגוריה, מזהה הפונה, מזהה המתנדב והעלות המשוקללת לשיבוץ. טבלה זו משמשת להערכת ביצועי האלגוריתם ולהבנת תוצאות השיבוץ.

עלות	מרחק בק"מ	קטגוריה	מזהה כונו	מזהה פונה	מספר קריאה	0
42.730280	7.209749	אחר	יעקב 4877	רויטל 3311	63203	0
33.631885	26.699277	דלת	שרון 0831	שרה 3853	40723	1
39.722319	31.056338	הנעה	כהן 1376	טאובר 5336	11284	2
31.029550	11.894293	הנעה	יוסף 0038	אסף 4743	56903	3
33.367015	18.658036	הנעה	יוסף 0231	אוריאל 2094	69736	4
31.712467	2.358158	הנעה	יונתן 5795	חגי 3130	25271	5
47.337707	13.866622	הנעה	יהודה 3915	אברהם 1118	16050	6
45.200062	22.951321	הנעה	אלי 4017	טל 6261	81170	7
26.971358	22.654979	הנעה	מרדכי 1740	רינה 1119	84942	8
30.061291	30.061291	הנעה	יוני 2972	שלמה 6907	14520	9
45.627431	10.412979	פנצ'ר	יוסף 3454	שנאור 0292	17456	10



9. הצגת חלופות

המודל ההונגרי, הידוע גם בשם אלגוריתם ההקצאה של קוניג, הוא אלגוריתם מתמטי המשמש לפתרון בעיות הקצאה אופטימליות. המודל ההונגרי מיועד למציאת הקצאה אופטימלית של משאבים (כגון עובדים, מתנדבים, מכונות וכו') למשימות (כגון אירועים, פרויקטים, פעילויות וכו') באופן שממזער את העלות הכוללת או ממקסם את הרווח הכולל. בעיות אלו מיוצגות בדרך כלל באמצעות מטריצת עלויות או מטריצת רווחים.

שלב ראשון במודל הוא להכין את הנתונים: על המודל לקבל מטריצת עלויות מרובעת, כלומר מספר העמודות שווה למספר השורות. במידה והמטריצה אינה מרובעת, המודל מוסיף שורות או עמודות פקטיביות עם ערכים גבוהים מאוד (או נמוכים מאוד במקרה של רווחים) כדי להפוך אותה למרובעת.

בשלב השני יש לבצע נרמול של הנתונים, שלב זה כולל שני חלקים חשובים - נרמול העמודות והשורות.

נרמול שורות - מטרת הנרמול בשלב זה היא להבטיח שבכל שורה יהיה לפחות ערך אפס אחד. זה נעשה על ידי חיסור הערך המינימלי של כל שורה מכל אחד מהערכים באותה שורה. לדוגמא:

נניח שיש לנו את מטריצת העלויות הבאה (4x4):

6	3	1	4
7	5	0	2
5	2	2	3
6	4	3	4

עבור השורה הראשונה, הערך המינימלי הוא 1.

מחסרים 1 מכל הערכים בשורה הראשונה:

$$(5, 2, 0, 3) \Rightarrow (1 - 6, 1 - 3, 1 - 1, 1 - 4)$$

וכן הלאה.

נרמול העמודות: מטרת הנרמול בשלב זה היא להבטיח שבכל עמודה יהיה לפחות ערך אפס אחד. זה נעשה על ידי חיסור הערך המינימלי של כל עמודה מכל אחד מהערכים באותה עמודה.

הנרמול חשוב כי הוא מאפשר לאלגוריתם ההונגרי לזהות את ההקצאה האופטימלית בצורה יעילה יותר. הנרמול מכין את המטריצה לשלב הכיסוי וההקצאה, בו משתמשים באפסים ליצירת ההקצאה המינימלית.

בשלב השלישי יש לבצע כיסוי אפסים: מכסים את כל האפסים במטריצה באמצעות מספר מינימלי של קווים (אופקיים ואנכיים). אם מספר הקווים שווה למספר השורות (או העמודות), עוברים לשלב ההקצאה. אם לא, ממשיכים לשלב הבא.

במידה ומספר הקווים אינו שווה למספר השורות או העמודות נעבור לשלב של התאמת מטריצת העלויות:

מחסרים את הערך המינימלי שאינו מכוסה מכל הערכים שאינם מכוסים. מוסיפים את הערך המינימלי לכל הערכים שנמצאים בהצטלבות של שני קווים. חוזרים לשלב הקודם עד שמספר הקווים שווה למספר השורות (או העמודות).

השלב החמישי והאחרון: שלב ההקצאה: בוחרים אפסים במטריצה כך שלא יבחרו יותר מאחד בכל שורה ועמודה. האפסים שנבחרו מייצגים את ההקצאה האופטימלית. יש לסכום את הערכים שנמצאים מיקום של האפסים במטריצה המנורמלת, אלו יתנו לנו את ההקצאה האופטימלית.



יתרונות מול חסרונות של המודל

יתרונות	חסרונות
פתרון אופטימלי: השיטה מבטיחה מציאת הפתרון הטוב ביותר עבור בעיית ההקצאה הנתונה.	מורכבות: השיטה דורשת מספר שלבים ויכולה להיות מורכבת להבנה ולביצוע, במיוחד עבור מטריצות גדולות.
יעילות חישובית: השיטה יעילה מבחינת זמן ריצה ויכולה להתמודד עם מטריצות גדולות יחסית.	דרישות לזיכרון וחישוביות במטריצות גדולות.
מתאים למגוון רחב של בעיות הקצאה: ניתן להשתמש בשיטה במגוון רחב של בעיות הקצאה במגזרים שונים.	דורש התאמות נוספות למטריצות לא ריבועיות.

המודל ההונגרי יעיל בלכידת התאמות אופטימליות בין שני סטים של פריטים, כמו מתנדבים ואירועים, בצורה מדויקת. המודל מתאים במיוחד להקצאות שבהן יש למזער עלויות או למקסם יעילות, ולכן הוא אידיאלי למקרים בהם יש להקצות את המתנדבים המתאימים ביותר לאירועים בהתבסס על פרמטרים כמו דירוג מתנדב ומרחק מהאירוע. יתרון מרכזי של המודל ההונגרי הוא היכולת שלו להבטיח פתרון אופטימלי ולמנוע בחירה שאינה אופטימלית, מה שמקנה אמינות גבוהה יותר בביצוע הקצאות מדויקות.

מודל זה מאפשר גם לזהות את המקרים שבהם יש צורך במתנדבים נוספים או באירועים נוספים, בכך שהוא מראה את היחס בין המתנדבים לאירועים בצורה ברורה. למרות שהמודל עשוי להיראות מורכב, הוא מבוסס על אלגוריתם מתמטי חזק המבטיח פתרון מדויק ואופטימלי תוך זמן ריצה סביר. יתרון נוסף הוא שהמודל אינו מצריך התאמות מרובות לפרמטרים, מה שמקל על היישום וההטמעה שלו בכלים וספריות מודרניות של מדעי הנתונים.

השימוש במודל ההונגרי מאפשר למנהלי מערכות הקצאה לקבל תובנות על חשיבותם של פרמטרים שונים בתהליך ההקצאה, ומסייע בזיהוי הגורמים המשפיעים ביותר על יעילות הקצאת המשאבים. יתר על כן, בזכות מבנהו המתמטי המובנה, המודל קל יחסית ליישום והוא מספק תוצאות עקביות ואמינות במגוון רחב של יישומים תפעוליים.

המודל ההונגרי מתאים באופן אידיאלי לפתרון בעיית ההקצאה בארגון ידידים, שדורשת התאמה אופטימלית של מתנדבים למשימות חילוץ בהתבסס על דירוג המתנדבים ומרחקם מהאירוע. להלן הסיבות לכך:

1. הקצאה אופטימלית: המודל ההונגרי מתמקד במציאת פתרון אופטימלי לבעיות הקצאה, ומבטיח שכל משימה תקבל את המתנדב המתאים ביותר על פי שילוב של דירוג ומרחק. זהו יתרון קריטי במיוחד בארגון ידידים, שבו הקצאה יעילה של מתנדבים יכולה להציל חיים ולהבטיח תגובה מהירה ואמינה לאירועים.

2. איזון בין קריטריונים מרובים: המודל ההונגרי מאפשר שקלול בין ציון המתנדב ומרחקו מהאירוע, כך שהמתנדב עם הציון הטוב ביותר ובמרחק הקצר ביותר יוקצה למשימה. היכולת למצוא את האיזון האופטימלי בין מיומנות המתנדב לבין זמינותו המהירה משפרת את היעילות הכוללת של המערכת.

3. מניעת התאמות לא אופטימליות: באמצעות המודל ההונגרי נמנעים מצבים שבהם מתנדבים פחות מתאימים מוקצים למשימות קריטיות, מה שעלול לקרות בשיטות הקצאה פשוטות יותר. זה מבטיח שכל משימה תקבל את המתנדב האידיאלי ביותר, תוך התחשבות בכל הקריטריונים הרלוונטיים, ובכך מעלה את איכות השירות.



4. יעילות חישובית: המודל ההונגרי ידוע ביעילות החישובית שלו, מה שמאפשר להתמודד עם מספר רב של מתנדבים ומשימות בצורה מהירה. במצבים שבהם יש צורך בהקצאה מהירה בזמן אמת, היעילות החישובית של המודל מהווה יתרון משמעותי.

5. גמישות והתאמה: המודל ההונגרי גמיש וניתן להתאמה למגוון רחב של קריטריונים נוספים במידת הצורך, מה שמאפשר להוסיף פרמטרים נוספים מעבר לדירוג ומרחק בעתיד ובהתאם לצורך הנלווה לעמותה. תכונה זו הופכת את המודל לכלי רב עוצמה שיכול לגדול ולהתפתח יחד עם צורכי הארגון המשתנים.

באמצעות שימוש במודל ההונגרי, ארגון ידידים יוכל להבטיח שכל משימת חילוץ תוקצה למתנדב המתאים ביותר בצורה מהירה ויעילה, ובכך למקסם את היכולת להתמודד עם מצבי חירום בצורה הטובה ביותר. מודל זה משפר את תהליך ההקצאה, מספק תגובה מהירה יותר, ומעלה את שביעות הרצון הכללית של המתנדבים והלקוחות כאחד.

לבקשת מנחה העבודה התבקשנו לבצע השוואת חלופות של השיטה הקיימת כעת בארגון אל מול המודל ההונגרי שבחרנו להטמיע בעמותה ואל מודלים שונים שיוכלו לפתור את הבעיה. בחרנו לבצע השוואה בין מודלי AHP, GA.

פתרון 1- מודל (AHP - Analytic Hierarchy Process)

הסבר : AHP הוא כלי לקבלת החלטות רב-קריטריוניות. המודל משמש להערכת ובחירת אלטרנטיבות על ידי פירוק הבעיה להיררכיה של קריטריונים ותת-קריטריונים, ולאחר מכן השוואה ביניהם בזוגות תוך שימוש בסולם ליקרט. כל קריטריון וכל אלטרנטיבה מקבלים משקל יחסי בהתאם לחשיבותם, מה שמאפשר קבלת החלטה מושכלת ומובנית. המודל מאפשר לשקלל ולהשוות בין מספר רב של קריטריונים, כמו דירוג המתנדב, מרחק מהאירוע, ניסיון קודם, זמינות ועוד. היכולת לשלב קריטריונים שונים בצורה היררכית מאפשרת להעריך את המתנדבים בצורה הוליסטית ומעמיקה.

פתרון 2- מודל ה-GA (Genetic Algorithm)

הסבר : האלגוריתם הגנטי (Genetic Algorithm, GA) הוא טכניקת אופטימיזציה המבוססת על עקרונות האבולוציה והברירה הטבעית בטבע. מטרתו היא למצוא פתרון מיטבי לבעיות מורכבות על ידי שיפור הדרגתי של קבוצת פתרונות אפשריים. האלגוריתם מתחיל בקבוצת פתרונות אקראיים הנקראת "אוכלוסייה" ומעריך את האיכות של כל פתרון באמצעות פונקציית מטרה. הפתרונות הטובים ביותר נבחרים להמשך, משולבים ביניהם (רבייה) ומעורבים בהם שינויים קטנים אקראיים (מוטציות). התהליך הזה חוזר על עצמו במספר דורות, כאשר כל דור שואף להניב פתרונות טובים יותר מהדור הקודם.

עבור הפרויקט שלנו, שמטרתו לשבץ את המתנדבים המתאימים ביותר לאירועים בהתחשב בזמן הגעתם ובדירוגם, האלגוריתם הגנטי יכול להיות כלי מצוין. הוא מסוגל להתמודד עם בעיות שבהן יש מספר רב של משתנים ופתרונות אפשריים, ומשפר את הפתרונות בצורה אינקרמנטלית עד שהוא מוצא את האיזון האופטימלי בין הקריטריונים השונים. יתרונו הגדול של ה-GA הוא ביכולתו לחפש פתרונות באזורים רחבים של המרחב האפשרי, מה שמאפשר לו למצוא פתרונות טובים גם כאשר המערכת מורכבת ויש לה פוטנציאל לקונפליקטים בין הקריטריונים. נשתמש בסולם של 1-10 (כאשר 1 הוא הנמוך ביותר ו-10 הוא הגבוה ביותר)



9.1 טבלה מסכמת

קריטריונים	משקל קריטריון	G.A	AHP	המודל ההונגרי
גמישות והתאמה	10%	גמיש מאוד ויכול להיות מותאם בקלות לשינויים בקריטריונים ובמטרות הארגון. (9)	דורש עדכון המשקלים וההיררכיות במקרה של שינוי בצרכים, מה שעלול להוות תהליך מסורבל. (4)	פחות גמיש, שכן הוא מיועד לבעיות הקצאה ספציפיות עם קריטריונים קבועים. (3)
דיוק חיזוי	30%	גבוהה- מבצע חיפוש אקראי עם אלמנטים של אופטימיזציה, ולכן ייתכן שיגיע לפתרון קרוב לאופטימלי. עם זאת, לעיתים התוצאות עשויות להיות פחות מדויקות עקב אופי החיפוש. (7)	בינוני- מתאים היטב להחלטות מורכבות ומרובות קריטריונים, אך הדיוק תלוי במידה רבה בקביעת המשקלים והעדיפויות. (6)	גבוהה- המודל ההונגרי מוצא פתרון אופטימלי לבעיית הקצאה דו-צדדית ולכן מספק את הדיוק הגבוה ביותר בהקצאה. (10)
מהירות חישוב	25%	יכול להיות איטי יחסית, במיוחד אם האוכלוסייה והדורות מוגדרים לגדלים גדולים. (5)	תהליך קבלת ההחלטות ב-AHP יכול להיות מורכב וגוזל זמן, במיוחד כשיש מספר רב של קריטריונים ואלטרנטיבות. (6)	מהיר ויעיל מאוד עבור בעיות הקצאה, במיוחד כשהמידות הן קטנות עד בינוניות. (10)
רמת מורכבות יישומית	20%	דורש הבנה מעמיקה של האלגוריתם ושל הפרמטרים השונים (כמו גודל אוכלוסייה, סיכוי למוטציה וכדומה), מה שהופך אותו למורכב יותר ליישום ולתחזוקה. ולכן פחות רלוונטי ליישם אותו בעמותה שהיא ללא מטרות רווח. (4)	דורש הגדרה מדויקת של היררכיות ומשקלים, ויכול להיות מורכב לביצוע באופן ידני, אך קיים מגוון של כלים ותוכנות שמפשטות את היישום. (7)	קל יחסית ליישום באמצעות אלגוריתם מובנה ומבוסס היטב, עם דרישות תחזוקה מינימליות. (9)
סקלאביליות	15%	ניתן להתאמה למערכות גדולות ומורכבות, אך ככל שהמערכת גדלה, זמן החישוב עשוי לגדול. (8)	פחות מתאים לסקלות גדולות, שכן העלות החישובית עולה באופן מעריכי עם מספר האלמנטים שיש להשוות ביניהם. (5)	מצטיין בהקצאות דו-צדדיות בגדלים קטנים עד בינוניים, אך יכול להתמודד עם מערכות גדולות יותר באמצעות אופטימיזציות. (7)



סה"כ		6.25	5.85	8.65
------	--	------	------	------

בחירת המשקלות לכל קריטריון נובעת מהשיקולים האסטרטגיים ולפי הצרכים שהגדרנו של ארגון ידידים- סיוע בדרכים, כמו גם מהאופן שבו המודלים השונים מתאימים לתרחיש הקיים. הנה ההסבר לבחירת המשקלות לכל קריטריון:

- דיוק החיזוי (30%) -הדיוק בהתאמה הוא קריטי עבור ארגון ידידים סיוע בדרכים, כיוון שהמטרה העיקרית היא להקצות מתנדבים בצורה אופטימלית לאירועים על פי הקריטריונים החשובים (כגון קרבה גיאוגרפית ודירוג המתנדב). כל טעות או חוסר דיוק בהקצאה יכול להוביל לפגיעה בשירות שניתן לאזרחים וביכולת לספק מענה מהיר ואפקטיבי. לכן, משקל זה הוא הגבוה ביותר.
- מהירות חישוב (25%) - זמן התגובה הוא פקטור מרכזי בשירות כמו ידידים סיוע בדרכים, שבו יש צורך להקצות מתנדבים במהירות למקרי חירום. מודל שיכול לספק תוצאות במהירות גבוהה יותר יאפשר תגובה מהירה יותר בשטח. לכן, מהירות החישוב קיבלה משקל גבוה אך נמוך במעט מהדיוק.
- רמת מורכבות יישומית (20%) - כעמותה ללא מטרות רווח, ייתכן שלא תהיה לארגון ידידים היכולת להעסיק צוותים טכנולוגיים מתקדמים או משאבים כספיים לטיפול במודלים מורכבים. לכן, חשוב שהמודל יהיה קל ליישום ולתחזוקה. בשל כך, רמת המורכבות קיבלה משקל משמעותי, שכן מודל פשוט יותר יהיה קל יותר ליישום ותפעול.
- סקלאביליות (15%) - הארגון אמנם יכול לגדול בעתיד, אך נכון להיום הצורך בסקלאביליות פחות קריטי מאשר הדיוק, המהירות והפשטות ביישום. עם זאת, חשוב לשקול את יכולת המודל להתרחב עם הארגון, ולכן הקריטריון הזה עדיין קיבל משקל מסוים.
- גמישות והתאמה לצרכים משתנים (10%) - בעוד שגמישות היא תכונה חשובה, היא פחות קריטית במצב הנוכחי של ארגון ידידים סיוע בדרכים, שבו ההקצאה היא משימה ברורה ומוגדרת. לכן, גמישות קיבלה משקל נמוך יותר, משום שהיא עשויה להיות רלוונטית יותר בעתיד או בתנאים משתנים, אך כרגע היא פחות בעדיפות.

- המודל ההונגרי קיבל את הציון המשוקלל הגבוה ביותר (8.65), בזכות הדיוק הגבוה שלו והפשטות ביישום. הוא מתאים במיוחד לארגון כמו ידידים שמחפש פתרון יעיל, מדויק, ומעשי.

- המודל G.A קיבל ציון של 6.25, בעיקר בזכות הגמישות הרבה שהוא מציע, מה שהופך אותו לפתרון שמתאים במיוחד למערכות מורכבות או לתרחישים שבהם יש צורך להתאים את הקריטריונים או המטרות באופן תדיר. עם זאת, עבור ארגון ללא מטרות רווח כמו ידידים, שזקוק למערכת פשוטה, יעילה וקלה ליישום, המודל הזה פחות מתאים. עם זאת, הוא עשוי להתאים בעתיד ככל שהארגון יתפתח ויצטרך להתמודד עם אתגרים מורכבים יותר.

-מודל AHP קיבל ציון של 5.85 ומציע איזון מוצלח בין דיוק וגמישות, מה שמאפשר לו להתמודד היטב עם החלטות מרובות קריטריונים. היכולת שלו לשקלל ולהעריך גורמים שונים בהחלטה הופכת אותו למתאים במיוחד בסיטואציות שבהן יש חשיבות למשקלים שונים של קריטריונים. עם זאת, בהשוואה למודל GA, ה-AHP פחות גמיש כאשר הצרכים משתנים לעיתים קרובות. בעוד ש-GA יכול להתאים את עצמו במהירות לשינויים, ה-AHP דורש עדכונים מורכבים של המשקלים וההיררכיות, מה שעלול להיות פחות יעיל ויותר מסורבל ליישום ולתחזוקה בארגון כמו ידידים.

בהתאם לצרכים הספציפיים של ארגון ידידים ולקריטריונים שנבחנו, ניתן להסכים כי המודל ההונגרי הוא הבחירה המתאימה ביותר ליישום בעמותה ללא מטרות רווח. המודל מצטיין בדיוק גבוה ומהירות חישוב, ומורכבותו הנמוכה יחסית הופכת אותו לאידיאלי להקצאת מתנדבים בצורה יעילה ומדויקת. בעמותה, לא בהכרח יש את מירב האנשים הטכנולוגיים הנדרשים לשילוב מודלים מסובכים יותר, ולכן המודל ההונגרי מתאים במיוחד. בנוסף, מכיוון שהעמותה מתמודדת עם אתגרי תקציב, יש עדיפות למודל פשוט שקל ליישום ולטיפול שוטף.



בעוד ש-GA יכול להוות פתרון גמיש במערכות מורכבות, במידה והעמותה תתפתח ותגדל בעתיד, הם יוכלו לשקול לשפר את המערכת ולהטמיע את מודל ה-GA כדי להשיג גמישות והתאמה למערכות מורכבות יותר. בסיכומי של דבר, המודל ההונגרי מציע יתרונות ברורים כשמדובר בהקצאה ישירה ומהירה, ולכן הוא הבחירה המועדפת לצרכים של ארגון ידידים.

10. מימוש פתרון

10.1 הצגת פתרון

בפריקט זה, מטרתנו הייתה לבצע אופטימיזציה של הקצאת כווננים (מתנדבים) לקריאות שירות עבור ארגון "ידידים - סיוע בדרכים". הפתרון שנבחר היה שימוש באלגוריתם ההונגרי לייעול הקצאת הכווננים בהתאם למיקום הגיאוגרפי, ציוני המתנדבים וזמינות מתנדבים.

תיאור הפתרון - אלגוריתם הונגרי :

השתמשנו באלגוריתם זה כדי למזער את העלות הכוללת בהקצאת מתנדבים לקריאות שירות. האלגוריתם פתר בעיית הקצאה אופטימלית על ידי חישוב מטריצת עלויות לכל קריאה והשוואת התאמה בין המתנדבים לקריאות.

מטרת הניסוי :

הניסוי נועד להעריך את השפעת האלגוריתם על זמן הטיפול בקריאות שירות. השווינו בין ממוצע זמן הטיפול לפני ואחרי יישום האלגוריתם.

תכנון הניסוי :

- קבוצת נתונים : השתמשנו בנתוני קריאות שירות ומתנדבים לפני ואחרי יישום האלגוריתם.
- מדדים : זמן טיפול ממוצע, עלות הקצאה, מרחק בין המתנדב לקריאה.

תרחישים שנבדקו :

תרחיש 1 : השוואת זמני טיפול לפני ואחרי האלגוריתם כאשר כל התנאים נורמליים.
תרחיש 2 : השוואת זמני טיפול תחת תנאים של עומס גבוה (יותר קריאות או מתנדבים).

ממצאי הרצות :

ANOVA : ה-F-statistic וה-p-value מעידים על הבדל מובהק סטטיסטית בין קבוצות.
T-Test : ה-T-statistic וה-p-value מראים הבדל משמעותי בין זמן טיפול לפני ואחרי האלגוריתם.

10.2 הממשק והליך העבודה

המערכת כוללת ממשק משתמש אינטואיטיבי וידידותי שפותח באמצעות Streamlit. הממשק מאפשר למשתמשים להזין את הנתונים הדרושים, כגון מיקום התקלה, מזהה הפונה וקטגוריית התקלה. הממשק מחובר ישירות למאגר הנתונים של המתנדבים, המאוחסן בקובץ CSV ב-GitHub, ומאפשר טעינה דינמית של נתונים אלו לצורך עיבוד בזמן אמת. עם הזנת הנתונים, הממשק מפעיל את האלגוריתם ההונגרי לחישוב השיבוץ האופטימלי של המתנדבים בהתבסס על מרחקים גיאוגרפיים, זמינות, וציון המתנדבים. הנתונים מחושבים באמצעות ספריות פיתוח כמו NumPy, Pandas, ו-Geopy, לאחר העיבוד, התוצאות מוצגות בטבלה מסודרת שמפרטת את מזהה המתנדב שנבחר, המרחק מהתקלה והקטגוריה המתאימה. הממשק נבנה בצורה כזו שהוא מספק למשתמש חוויה פשוטה אך מקצועית, תוך שמירה על גמישות בהזנת הנתונים והצגת התוצאות. בכך, המערכת מספקת פתרון יעיל ומותאם אישית לצורכי ארגון "ידידים".



אחד הקבצים המרכזיים שמנוהל ב-GitHub הוא קובץ ה-`requirements.txt`, המכיל רשימה של כל הספריות הנדרשות להפעלת הפרויקט בגרסה הרלוונטית. קובץ זה מאפשר למשתמשים להוריד ולהתקין את כל התלויות הדרושות בלחיצת כפתור, תוך שימוש בספריות כמו Pandas לעיבוד נתונים, NumPy לחישובים מתמטיים, Geopy לחישובי מרחקים גיאוגרפיים, SciPy לניתוחים סטטיסטיים, ו-Requests לתקשורת עם שירותים חיצוניים. כך ניתן להבטיח שהקוד יפעל בצורה חלקה וללא תקלות.

```
Code Blame 5 lines (5 loc) · 82 Byt
1 pandas == 2.1.4
2 numpy == 1.26.4
3 geopy == 2.4.1
4 scipy == 1.13.1
5 requests == 2.32.3
```

הקוד עושה שימוש בקובץ `volunteer_pool.csv`, המכיל את מאגר המתנדבים של ארגון "ידידים". המאגר נטען באמצעות Pandas ומומר ל-DataFrame, שמשמש כבסיס לחישובי משקלים לכל מתנדב, המבוססים על פרמטרים כמו מרחק גיאוגרפי, זמינות ודירוגים קודמים. לאחר חישוב המשקלים, האלגוריתם ההונגרי מבצע אופטימיזציה למציאת השיבוץ האופטימלי בין המתנדבים למשימות.

תהליך העבודה

1. הזנת נתונים: המשתמש מזין את פרטי הקריאה לסיוע, כגון מיקום התקלה, מזהה הפונה, וקטגוריית התקלה.
2. עיבוד הנתונים: הקוד מבצע עיבוד נתונים ומחשב את המשקלים עבור כל מתנדב פוטנציאלי, תוך שימוש במרחקים גיאוגרפיים, זמני תגובה, וקריטריונים נוספים. המידע מעובד באמצעות האלגוריתם ההונגרי, שמטרתו למצוא את השיבוץ האופטימלי.
3. תוצגת התוצאות: לאחר חישוב המשקלים וביצוע האלגוריתם, מוצגת תוצאה למשתמש בצורה ברורה ומובנת, כולל פרטי המתנדב הנבחר והמרחק באותו פונה המבקש סיוע.

ביצועי הממשק והערכה

הממשק שפותח מספק פתרון מקצועי ויעיל לתהליך שיבוץ המתנדבים בארגון "ידידים". השימוש בטכנולוגיות מתקדמות ובאלגוריתמים אופטימיזציה מאפשר להציג למשתמש תוצאות מדויקות ואמינות בזמן אמת, תוך שמירה על פשטות השימוש ונוחות מקסימלית. הממשק תורם לשיפור תהליך קבלת ההחלטות בארגון, ומספק תשתית יציבה להמשך פיתוח ויישום כלים נוספים בעתיד.



10.3 אילוצים

הארגון מתמודד עם מספר אתגרים, ביניהם :

מגבלה תקציבית המונעת שימוש בתוכנות זיהוי מיקום: ארגון "ידידים" מתבסס על תרומות ומתנדבים, דבר המכתיב מגבלות תקציביות שמונעות שימוש בטכנולוגיות מתקדמות לאיתור מיקום בזמן אמת באמצעות API ייעודי. כתוצאה מכך, אין אפשרות לאתר את מיקומם המדויק של המתנדבים בזמן אמת, ולכן הפתרון הנבחר מציע גישה ידנית להזנת קורדינטות אורך ורוחב.

בממשק המערכת, המשתמשים נדרשים להזין באופן ידני את מיקום התקלה ואת מיקום המתנדבים על פי קורדינטות גיאוגרפיות (קו אורך וקו רוחב). גישה זו מאפשרת לארגון לנצל את הכלים הקיימים בצורה מיטבית, תוך שמירה על הפשטות והנגישות של המערכת, מבלי להכביד על התקציב.

על אף האילוץ הטכני, המערכת מצליחה לספק מענה מקצועי ואמין, בזכות האלגוריתם ההונגרי המבצע את התאמת המתנדבים למשימות על בסיס הקורדינטות שהוזנו. כל הנתונים מעובדים באמצעות ספריות פיתוח כמו NumPy, Pandas, ו-Geopy, המאפשרות לבצע חישובים מדויקים ולהציג תוצאות רלוונטיות בממשק המשתמש.

תכנון זה מאפשר לארגון "ידידים" לשמור על גמישות ועלות נמוכה, תוך מתן פתרון פרקטי ויעיל לצורכי ניהול המתנדבים ושיבוץ המשימות, בהתאם למגבלות הקיימות.

ניהול נתונים רגישים ושימוש במזהים אנונימיים: במסגרת הפרויקט, ארגון "ידידים" בחר לנקוט באמצעי זהירות משמעותיים על מנת להגן על פרטיותם של הפונים והמתנדבים. עקב חשש לפגיעה בפרטיות ולשימוש בלתי הולם במידע רגיש, הארגון החליט שלא לספק נתונים אישיים מזהים ישירים כמו שמות מלאים, מספרי טלפון או מספרי תעודת זהות של המתנדבים והפונים.

במקום זאת, הנתונים שסופקו למערכת עברו תהליך של התממה (Anonymization), שבו הוחלפו המזהים האישיים במזהים אנונימיים. לדוגמה, במקום שמות אמיתיים או מספרי תעודת זהות, המזהים האנונימיים כוללים שם (או כינוי) וארבע ספרות רנדומליות, הנבחרות מתוך תעודת הזהות או ממקור אחר.

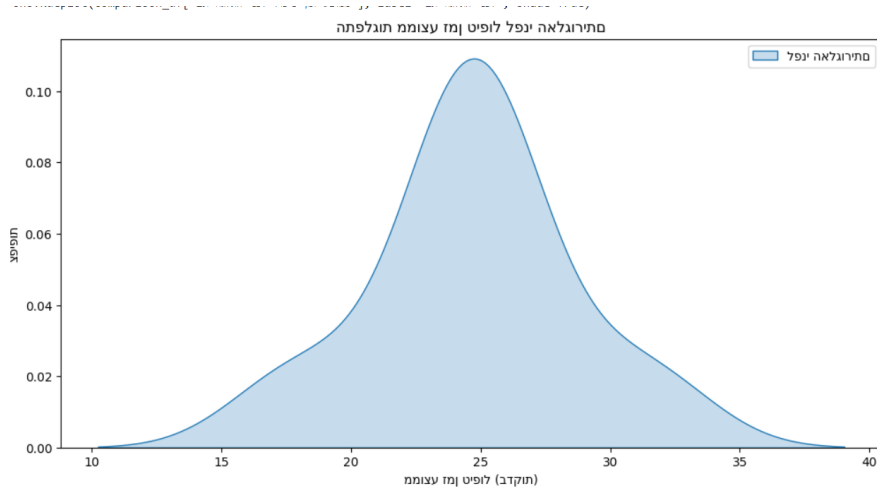
פתרון זה מאפשר לנו לעבוד עם נתונים שומרים על פרטיות המשתמשים, מבלי לוותר על היכולת לבצע ניתוחים ולקבל תובנות משמעותיות לצורכי השיבוץ והאופטימיזציה של המערכת. האלגוריתמים המיושמים במערכת מבוססים על המזהים האנונימיים, מה שמבטיח כי תהליך עיבוד הנתונים מתבצע בצורה בטוחה ומוגנת, בהתאם לדרישות הארגון ולתקני הגנת פרטיות.



11. הערכת הפתרון

כדי לבדוק את איכות האלגוריתם שלנו, נשתמש במבחני סטטיסטיים.
ע"מ להגיע לתוצאה המדויקת ביותר נבצע את הפעולות הבאות-

- ראשית נבדוק בעזרת מבחנים סטטיסטיים האם ממוצע זמני הטיפול לפני האלגוריתם מתפלגים התפלגות נורמלית- נתונים אלה הינם קבועים ולכן מספיק לבדוק פעם אחת בלבד את סוג ההתפלגות



בדיקת שפפה: $p\text{-value} = 0.39364044266813314$

בדיקת קולמוגורוב-סמירנוב: $p\text{-value} = 0.6407305410077333$

עבור שני המבחנים, התקבל $P\text{-value}$ גדול מ-0.05 ולכן נקבל את השערת האפס שאומרת שהנתונים מתפלגים התפלגות נורמלית.

- לפי משפט הגבול המרכזי, אנחנו מניחות התפלגות נתונים נורמלית עבור הנתונים המתקבלים לאחר האלגוריתם. זאת מכיוון שגודל המדגם (n) גדול מ-30.
לכן, נוכל להניח ששני וקטורי הנתונים מתפלגים נורמלית.

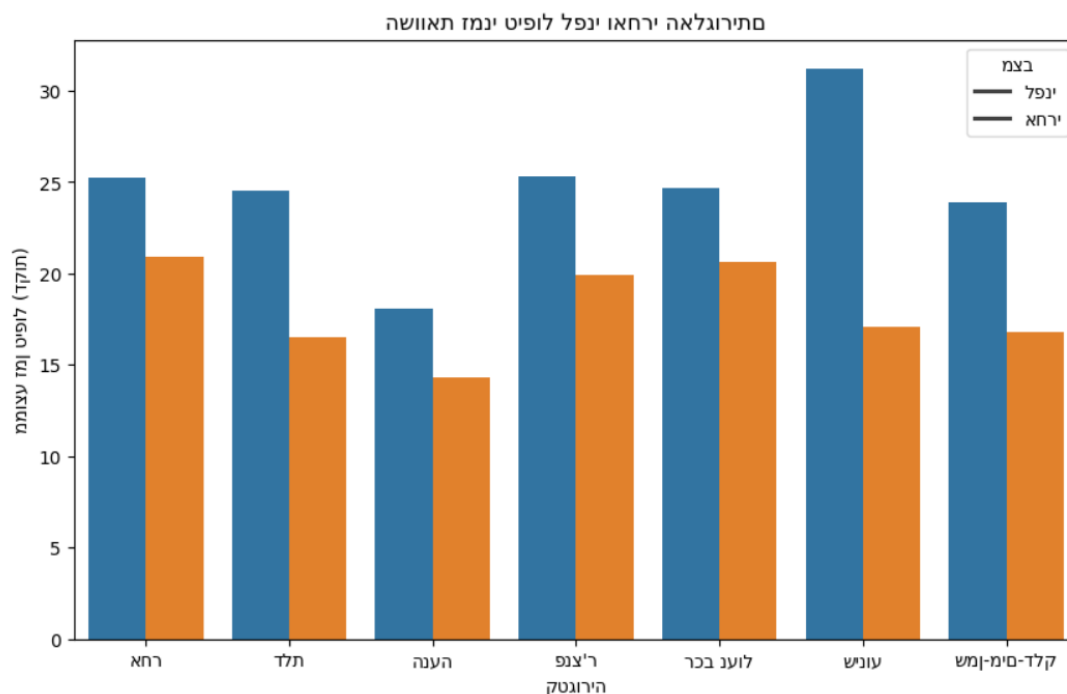
אי לכך, כדי לבדוק את איכות האלגוריתם נשתמש בשני מבחנים סטטיסטיים-

ANOVA test	T test	
0.0022571104	0.002904769	P-value
-	4.831854268	T-statistic
14.921296665	-	F-statistic
ה-p-value קטן מ-0.05, מה שמעיד שיש הבדל מובהק סטטיסטית בין ממוצעי הקבוצות השונות. זה אומר שלפחות אחת מהקבוצות נבדלת באופן משמעותי מהאחרות מבחינת ממוצע זמן הטיפול. אם מדובר בזמנים לפני ואחרי האלגוריתם, זה מצביע על כך שהשינויים בין הממוצעים לפני ואחרי האלגוריתם הם משמעותיים ולא נובעים רק מהשתנות אקראית.	ה-p-value קטן מ-0.05, מה שמעיד על הבדל מובהק סטטיסטית בין הקבוצות שנבדקו. תוצאות אלו מצביעות על כך שיש הבדל משמעותי בין ממוצע זמן הטיפול לפני האלגוריתם לבין זמן הטיפול אחרי האלגוריתם. ניתן להסיק שהאלגוריתם השפיע באופן משמעותי על זמן הטיפול, והשיפור שנמצא הוא מובהק סטטיסטית ולא מקרי.	מסקנה-



שני המבחנים (ANOVA ו-T-test) מראים הבדל מובהק סטטיסטית בין הקבוצות שנבדקו. ANOVA מראה הבדל משמעותי בין הקבוצות השונות באופן כללי, בעוד שה-T-test מציע הבדל מובהק בין זמן הטיפול לפני ואחרי השימוש באלגוריתם. תוצאות אלו תומכות בהשערה שהאלגוריתם שיפר את זמני הטיפול באופן מובהק.

להלן גרף עמודות אשר מוכיח שהאלגוריתם ההונגרי אכן הפחית משמעותית את ממוצע זמני הטיפול-



לסיכום:

במסגרת ההערכה, נבחנו:

זמני התגובה של המתנדבים לפני ואחרי הטמעת האלגוריתם באמצעות כלים סטטיסטיים מדויקים. ראשית, המרת נתוני זמני הטיפול לדקות אפשרה השוואה ישירה בין שתי התקופות. לאחר ההמרה, בוצעו בדיקות נורמליות (Shapiro-Wilk ו-Kolmogorov-Smirnov), שאישור שהנתונים מתפלגים בצורה נורמלית, מה שאפשר את המשך השימוש בשיטות פרמטריות. בשלב הבא, בוצע מבחן T לתלויים (Paired T-Test) שבדק את ההבדלים בזמני התגובה לפני ואחרי הטמעת האלגוריתם. תוצאות המבחן הראו הבדל מובהק סטטיסטית ($p\text{-value} < 0.05$), דבר המצביע על כך שהאלגוריתם אכן שיפר את זמני התגובה של המתנדבים. בנוסף לכך, בוצע מבחן ANOVA שנועד לבחון את השפעת האלגוריתם על קבוצות שונות של מתנדבים, כגון אזורים גיאוגרפיים או קטגוריות של אירועים. תוצאות ה-ANOVA הצביעו גם הן על שיפור מובהק בזמני התגובה בכל הקבוצות שנבדקו. כדי לחזק את הממצאים, בוצע גם מבחן Bootstrap, שבחן את השיפורים במונחים של שינוי ממוצעים באופן לא פרמטרי, והוכיח את המובהקות הסטטיסטית של השיפורים שנמדדו. האיורים המצורפים ממחישים בצורה ברורה את התוצאות המתקבלות, ומדגישים את ההשפעה החיובית והמשמעותית של האלגוריתם על טיב הביצועים. הפחתת עומס ומקסום אופטימלי של המתנדבים: נוסף על שיפור זמני הטיפול, המודל מתמקד בבחירת המתנדב הקרוב ביותר גיאוגרפית ובעל הניסיון המתאים ביותר למשימה, מה שמסייע במניעת עיכובים בטיפול באירועים ומבטיח שהמתנדב המיומן ביותר יוקצה לכל קריאה. בהרצת סימולציה מדגמית, המודל הצליח להתמודד בצורה יעילה יותר עם עומסי קריאות, במיוחד בשעות השיא ובזמני עומס גבוהים, באמצעות שילוב הסינון הגרידי וגישה היברידית. סינון ראשוני זה הקטין את העומס החישובי על האלגוריתם, מה שהוביל ליעול הקצאת המתנדבים ולמניעת עיכובים נוספים. מבחני ANOVA שנעשו על קבוצות שונות של מתנדבים הראו שהשיפורים הללו חלים באופן שיטתי על כל הקבוצות שנבדקו, ללא תלות במיקום גיאוגרפי או קטגוריית האירוע, דבר המצביע על פוטנציאל לשיפור משמעותי באיכות השירות ובשביעות הרצון של מבקשי הסיוע. *המודל המשולב מוזכר בסקר הספרות ובנספחים.



12. דיון ומסקנות

12.1 ניתוח הממצאים

הפרויקט התמקד בשיפור תהליך הקצאת המתנדבים לקריאות שירות בארגון "ידידים - סיוע בדרכים" באמצעות מערכת אופטימיזציה מבוססת אלגוריתם הונגרי. לפני ההטמעה, תהליך השיבוץ היה ידני, מה שהוביל לעיכובים בזמני התגובה ולניצול לא אופטימלי של משאבי המתנדבים. במהלך הניסוי והסימולציות, גילינו שהמערכת החדשה מצליחה לשפר באופן משמעותי את זמני התגובה של המתנדבים. מבחנים סטטיסטיים כמו T-Test ו-ANOVA הראו שהשיפורים בזמני הטיפול מובהקים סטטיסטית, במיוחד בשעות עומס. בנוסף, האלגוריתם ההונגרי בשילוב עם סינון גרידי הצליח להפחית את עומס החישוב, למנוע עיכובים ולהבטיח הקצאת מתנדבים מהירה ומדויקת יותר. הממצאים מצביעים על כך שהמערכת מספקת שיפורים עקביים בכל קבוצות המתנדבים, באזורים גיאוגרפיים שונים ובקטגוריות אירועים מגוונות.

12.2 המלצות לארגון

שלב ההטמעה של האלגוריתם במערכות העמותה לא נכלל במסגרת פרויקט זה. אי לכך, ריכוזו בעבור העמותה מספר דגשים ופעולות להמשך ע"מ להבטיח הטמעה נכונה של האלגוריתם למימוש פתרון הבעיה.

הטמעה באפליקציית הכוננים של פיצ'ר 'סטטוס טיפול' - בסיום טיפול תקלה, הכונן יידרש לציין את סטטוס גמר טיפול בתקלה שיכלול את הערכים 'נסגר בהצלחה', 'נסגר ללא הצלחה', 'בוטל' ו'הועבר' במטרה להבטיח את אמינות של הפרמטר 'אחוזי הצלחה' שהמוה חלק מהמשוואה.

הכנסה של פיצ'ר 'דירוג מתנדב' באפליקציה של מבקשי העזרה - פיצ'ר חדש שיוטמע באפליקציה שיוצג למבקש הסיוע בסיום התקלה. המערכת תבקש לדרג את איכות השירות של אותו כונן שסייע בתקלה ובכך אנחנו מקנות לעמותה את היכולת לדעת את טיב השירות של המתנדבים שלה. גם דירוג זה נכלל בציון הסופי אותו מקבל המתנדב.

הטמעה של הטבלאות - כדי להבטיח שיטת עבודה נכונה, ריכוז נתונים והטמעה נכונה של האלגוריתם יש חשיבות רבה לחלוקה נכונה של הטבלה המקורית לטבלאות אותן הצגנו בדו"ח זה.

עדכון הכונן וביצוע חבירה - בהתקבל הכונן האופטימלי בעזרת האלגוריתם שבנינו העמותה אחראית למצוא את הדרך להפצת הבשורה הן לכונן והן למבקש הסיוע ולחבר בין הגורמים הנ"ל.

כמו כן, לאור זה שהפרויקט היה ללא עלות ולכן לא הושקעו בו תקציבים, נבצר מאיתנו להשתמש בתוכנת API שמטרתה לנטר מיקום מדויק הן בעבור מיקום התקלה והן בעבור מיקום הכונן ובאמצעותה לחשב את המרחק המדויק בין השניים. בפרויקט זה לא יכולנו להשתמש בתכונה מסוג זה מאחר והשימוש בה גבוה מאוד.

12.3 תובנות ולקחים

במהלך הפרויקט, הוסקו מספר תובנות ולקחים חשובים: חשיבות שיתוף פעולה עם הארגון: שיתוף פעולה הדוק עם צוות הארגון הוא קריטי להצלחת הפרויקט. זה מאפשר להבין את הצרכים והאתגרים הספציפיים של הארגון ולבצע התאמות נדרשות בזמן אמת. חשיבות ההתאמה לצרכים ספציפיים של הארגון: המערכת שפיתחנו לא רק נועדה לשפר את הקצאת המתנדבים אלא גם להתאים לצרכים ולמגבלות הייחודיות של ארגון "ידידים". זה כולל התחשבות במגבלות תקציביות, אי יכולת להשתמש בטכנולוגיות מתקדמות לזיהוי מיקום בזמן אמת, ושמירה על פרטיות המתנדבים והמבקשים. תובנה זו מדגישה את החשיבות של התאמה אישית של פתרונות טכנולוגיים לאילוצים ולצרכים של ארגונים קטנים ובינוניים, במיוחד כאלה שפועלים במסגרת תקציב מוגבל.



ניהול נתונים רגישים : טיפול זהיר בנתונים רגישים הוא חיוני. שימוש במנגנוני התממה לשמירה על פרטיות המשתמשים התגלה כפתרון יעיל ואמין, ששיפר את ביצועי המערכת תוך שמירה על דרישות הגנת הפרטיות.

חשיבות עיצוב ממוקד משתמש : הפרויקט הדגיש את הצורך בהתמקדות בחוויית המשתמש בעת פיתוח מערכת, במיוחד כאשר המשתמשים הם מתנדבים עם רמות שונות של ידע טכנולוגי. על ידי עיצוב ממשק אינטואיטיבי וקל לשימוש, הבטחנו שהמתנדבים יוכלו להשתמש במערכת במהירות וביעילות, מה שמוביל לשיעורי אימוץ גבוהים ולביצועים טובים יותר של המערכת. איכות ושלמות הנתונים : איכות ושלמות הנתונים בהם נעשה שימוש במערכת הם קריטיים להשגת תוצאות מדויקות ואמינות. במהלך הפרויקט, הושם דגש רב על ניקוי ואימות הנתונים כדי להבטיח שהנתונים המוזנים לאלגוריתם האופטימיזציה יהיו מדויקים. המיקוד הזה באיכות הנתונים הוא חיוני לשמירה על אמינות המערכת ויעילותה. שיפור מתמיד באמצעות לולאות משוב : שילוב לולאות משוב קבועות בתפעול המערכת הוא חיוני לשיפור מתמשך. על ידי איסוף וניתוח משוב מהמתנדבים ומהארגון, ניתן לשפר ולעדכן את המערכת על פי הצרכים המשתנים ולהבטיח שהיא תמשיך להעניק ערך לאורך זמן. שימוש באנליטיקה מתקדמת לקבלת החלטות : הפרויקט הראה את הכוח של שימוש בכלים מתקדמים לאנליטיקה ואופטימיזציה לשיפור תהליכי קבלת ההחלטות בארגון. יישום האלגוריתם ההונגרי ושימוש בכלים סטטיסטיים תרמו לקבלת החלטות מבוססות נתונים, מה שמוביל לשיפור ניכר ביעילות ובאפקטיביות התפעולית.

12.4 השלכות ותחזיות לעתיד

המערכת החדשה מספקת כלי יעיל לניהול אופטימלי של המתנדבים ומציעה פתרון לבעיות הקיימות בתהליך הידני. היכולת למקסם את ניצול המשאבים האנושיים תורמת להפחתת זמני התגובה, לשיפור איכות השירות ולשביעות רצון גבוהה יותר של המשתמשים. עם זאת, יש לקחת בחשבון שהשיפורים שהמערכת מציעה מבוססים על תוצאות סימולציה בלבד. לכן, מומלץ לערוך מעקב מתמשך אחר ביצועי המערכת בשטח ולהתאים אותה לפי הצורך על מנת להבטיח את הצלחתה לטווח הארוך. שיפור עתידי בטכנולוגיה : לצד ההמלצות הקיימות, כדאי לשקול לשלב בעתיד טכנולוגיות נוספות, כמו אינטגרציה עם מערכות GPS בזמן אמת לשיפור הדיוק במיקום המתנדבים. זה יאפשר התאמה מהירה ומדויקת יותר לצורכי השטח ויגביר את יעילות המערכת. מעקב אחר התקדמות המתנדבים : מומלץ לשלב במערכת כלים למעקב אחר התקדמות המתנדבים, כמו קבלת פידבק על ביצועים באופן שוטף ודיווחי יעילות.

12.5 תרומה של כל חבר צוות-

בתחילת הדרך, כשהתבקשנו לבחור את פרויקט הגמר שלנו, היה לנו ברור שנרצה לעסוק בפרויקט שיעזור ויתרום לעמותה שמקדמת ערכים של תרומה לחברה והיה חשוב לנו שהפרויקט יהיה בעל משמעות אמיתית. החלטנו לבחור בתחום הידע והנתונים, שכן זהו תחום שמרתק אותנו מאוד, ובחרנו באתגר של בניית מודל ייחודי ומועיל מפני שהאמנו שכך נוכל ללמוד ולאתגר את עצמנו הכי שאפשר.

בפרויקט זה שיתפנו פעולה באופן הדוק לפיתוח המודל, המאמצים הקולקטיביים שלנו השתרעו על גבי השלבים השונים של הפרויקט, כולל איסוף נתונים, תכנון מערכת, עיצוב ויצירת אב טיפוס אשר התעדכן לאורך הפרויקט במטרה להתאימו בצורה הטובה ביותר לעמותת ידידים שישתמשו בו.

המאמץ המשותף שלנו הביא ליצירת מודל אופטימיזציה מוצלח שעתיד לעזור בקבלת ההחלטות ולהביא לידי שיפור בביצועים של עמותת ידידים.

ליאור בירנדורף :

כחברת צוות בפרויקט, אני גאה ושמחה להיות חלק מפרויקט זה. מתוך מחויבות משותפת למצוינות והצלחה, לקחנו על עצמנו את המשימות השונות כצוות, בהבנה עמוקה של חשיבות הפרויקט לעמותת ידידים וכתרומה לחברה.

לאורך כל הפרויקט, נשאתי באחריות למגוון תחומים, מה שאיפשר לי לתרום משמעותית להצלחתו. בתחילת הדרך, הייתי אחראית על ניסוח והגשת הצעת הפרויקט, תהליך שבו למדתי את אחד השיעורים החשובים ביותר – פרויקט הוא תהליך דינמי, מלווה בשינויים ואי-וודאות.



במהלך הפרויקט, לקחתי חלק פעיל ומשמעותי, החל מאיחוד הנתונים הרלוונטיים ועד להפיכת המידע הגולמי לידע מעשי. כל שלב דרש ממני שיקול דעת מדוקדק ותשומת לב לפרטים. בשיתוף פעולה עם המנחה התעשייתי, הבטחנו שהמודל שפיתחנו עונה על הצרכים הספציפיים של העמותה.

היו רגעים שבהם נדרשתי לחקר מעמיק וניתוח יסודי של הנתונים, לצד ארגון ועיבוד המידע במטרה להגיע לתוצאה מיטבית. היכולת להפוך מידע גולמי לתובנות משמעותיות העניקה לי תחושת הישג והצלחה, והעמיקה את הבנתי בפרויקט ואת מחויבותי למצוינות. כעת, כשהפרויקט הגיע לסיומו, אני מרגישה תחושה של הגשמה והצלחה, הידיעה שהעבודה הקשה והמסירות שלנו הגיעו לשיאה. אני אסירת תודה על ההזדמנות להיות חלק מפרויקט הגמר לצד חברות הצוות המוכשרות, ביחד התגברנו על אתגרים, שמחנו כאשר היו ניצחונות קטנים והערות חיוביות ולמדנו מההערות והארות של המנחים. אני מאמינה שביצוע פרויקט הגמר יסייע לי בעבודה בתעשייה ובפתרון בעיות, ובפרט בפרויקטים עתידיים שאצטרך לבצע.

סלין מירילשוילי :

פרויקט הגמר מהווה את אבן הדרך האחרונה והמשמעותית ביותר בתואר, שבו אנו, הסטודנטים, נדרשים ליישם בצורה מעשית ועצמאית את הכלים שרכשנו במהלך שנות הלימודים. עבודת הצוות הייתה יוצאת דופן, וכל אחד מחברי הצוות תרם את הגוון הייחודי שלו למען המטרה המשותפת.

בתחילת הפרויקט, הייתי אחראית על שלב סקירת הספרות, שבו גיבשנו את הבסיס לשיטות והמתודות ששימשו אותנו בהמשך העבודה. מעורבותי הייתה משמעותית בכל שלב, ולמדתי להכיר את עוצמתה של עבודת צוות. שיתוף הפעולה מצד חברות הצוות ושימוש מושכל בחוזקות של כל אחת הובילו לעבודה יעילה, חיסכון בזמן, ומקסום התוצאות. כמו כן, גיליתי שחשוב להגדיר בצורה מדוקדקת ומקיפה את המשימות הקטנות לאורך הדרך, ולתעדף אותן נכון, כדי לשפר משמעותית את יעילות העבודה ולהשיג את היעדים שהצבנו. ראיתי לנכון לשלב חקר מעמיק מסקירת הספרות עם חקר המצב הקיים בארגון והשתקפותו מחקר הנתונים, המגמות והפערים שהארגון לא ידע להציף, על מנת שנוכל לשרת את מטרתו בצורה יעילה ואפקטיבית ולהמליץ על שיפורים שיוכלו ליישם למראה מגמת העלייה. לסיכום, פרויקט הגמר אפשר לי לראות כיצד ניתן ליישם קורסים, פרויקטים, התייעצויות ושיחות ידע עם מרצים ומתרגלים בצורה מוחשית, עד לשימוש נעלה למען עמותה שרצונה הוא לטובת הכלל. הבנת תהליכי הארגון, התנהלות מול גופים גדולים, ידע בתחומי ניתוח נתונים, ועוד כל אלה באו לידי ביטוי בפרויקט. אני בטוחה שהתנסות זו תסייע לי בהמשך דרכי ובשאיפה להמשיך לעזור לעמותות וארגוני חסד.

שרון סייג :

בתחילת הדרך, כאשר הוצע לנו לבחור את פרויקט הגמר שלנו, היה לי ברור שארצה לעסוק בפרויקט בעל משמעות, מעבר להרחבת הידע והניסיון האקדמי. הבחירה נפלה על פרויקט שמסייע לעמותה ללא מטרות רווח, שעוסקת בסיוע לאנשים שנתקעו עם רכבם או נזקקים לעזרה דחופה אחרת. הערכים שבהם דוגלת העמותה, כמו נתינה, התנדבות, סולידריות ואחריות חברתית, מהווים עבורי עקרונות מפתח, והם ליוו את כל תהליך פרויקט הגמר שלי ושל חברות הצוות שלי.

בתחילת הפרויקט לקחתי על עצמי את האחריות לתקשורת רציפה מול המנחה התעשייתי שלנו, ובכך שימשתי כאיש הקשר שמתווך בין העמותה לצוות שלנו ומדווח על תהליך העבודה. לאורך כל הפרויקט, אני וחברות הצוות שלי עבדנו בשיתוף פעולה מלא, הכולל מפגשים פיזיים קבועים, מה שתרם רבות להתקדמות ולגיבוש שלנו כצוות.

במהלך השנה האחרונה, במסגרת הפרויקט, חוויתי תהליך למידה והתפתחות משמעותי. לראשונה, יכולתי ליישם את הכלים התיאורטיים והמעשיים שרכשתי במהלך לימודיי באופן מעשי. תהליך הכניסה לארגון חדש וההבנה של תהליכי העבודה היו עבורי מלמדים במיוחד. למדתי גם שעין חיצונית, במיוחד מנקודת מבט הנדסית, יכולה לתרום רבות לשיפור תהליכי עבודה קיימים ולפיתוח תהליכים חדשים.



הפרויקט עבורי מהווה אבן דרך חשובה ונקודת פריצה לשוק ולתעשייה. בזכותו, רכשתי ידע וכלים שישמשו אותי גם בעתיד. אני מרגישה כעת הרבה יותר מוכנה להשתלב בתעשייה ולהתמודד עם אתגרים בתחום שלי, ואני שמחה על כך שהייתה לי ההזדמנות לקחת חלק בפרויקט הזה יחד עם חברות הצוות שלי.



13. נספחים :

13.1 מימוש מתודולוגיית סקר הספרות ושיפור המודלים

יישום בפועל (A bipartite graph matching approach for relevance scoring and combinatorial optimization)

שלב 1 : חישוב TF-IDF עבור קטגוריות המשימות

בשלב זה, השתמשנו בטכניקת TF-IDF (Term Frequency-Inverse Document Frequency) כדי לחשב את הרלוונטיות של כל קטגוריה (סוג משימה) במסגרת המערכת. מכיוון שהקטגוריות במערכת מייצגות סוגי משימות שונים, חישוב ה-TF-IDF מאפשר לנו לקבוע את מידת הרלוונטיות של כל קטגוריה בהשוואה לאחרות, על סמך השכיחות שלהן. הרלוונטיות של קטגוריה מסוימת נלקחת בחשבון כחלק מחישוב הציון הסופי של המתנדבים, כך שניתן להבטיח התאמה טובה יותר של המתנדב למשימה הנדרשת.

```
from sklearn.feature_extraction.text import TfidfVectorizer
# על הקטגוריות TF-IDF שלב 1: חישוב
categories = user_data_df['קטגוריה'].values

vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(categories)

# לנתונים TF-IDF הוספת ציון
tfidf_scores = tfidf_matrix.sum(axis=1) # סכום הציונים עבור כל קטגוריה
user_data_df['tfidf_score'] = np.array(tfidf_scores).flatten()
```

שלב 2 : חישוב ממוצעים עבור פרמטרים שונים

לאחר מכן, חישבנו את הממוצעים עבור כל פרמטר מרכזי (כגון "ממוצע זמן טיפול בתקלה", "ממוצע דירוג לקוח", ו"אחוזי הצלחה") לפי קטגוריות המשימות. הממוצעים הללו שימשו אותנו לחישוב היחס בין כל ערך לבין הממוצע של הקטגוריה המתאימה, ובכך סיפקו לנו מדד להשוואת ביצועי המתנדבים ביחס לקבוצת השווים שלהם.

```
# חישוב הממוצעים של כל עמודה לפי קטגוריות עבור העמודות הרלוונטיות בלבד
columns_to_average = ['ממוצע דירוג לקוח', 'ממוצע זמן טיפול בתקלה', 'אחוזי הצלחה']
averages = user_data_df.groupby('קטגוריה')[columns_to_average].transform('mean')

# חישוב היחסים בין כל ערך לממוצע של העמודה המתאימה בקטגוריה
# הערה: בשביל ממוצע זמן טיפול בתקלה, נשתמש ביחס הפוך
user_data_df['t_ratio'] = averages['ממוצע זמן טיפול בתקלה'] / user_data_df['ממוצע זמן טיפול בתקלה']
user_data_df['r_ratio'] = user_data_df['ממוצע דירוג לקוח'] / averages['ממוצע דירוג לקוח']
user_data_df['w_ratio'] = user_data_df['אחוזי הצלחה'] / averages['אחוזי הצלחה']
```

שלב 3 : חישוב יחסים ונרמול ציונים

היחסים בין ביצועי המתנדבים לבין הממוצעים של קבוצת השווים חושבו עבור כל פרמטר רלוונטי. לדוגמה, עבור "ממוצע זמן טיפול בתקלה", נעשה שימוש ביחס הפוך, כך שמספרים קטנים יותר הצביעו על ביצועים טובים יותר. לאחר מכן, נרמלו היחסים הללו לציונים על סולם קבוע, כדי להקל על חישוב הציון הסופי.



```
# מתן ציונים לפי הסולם שניתן
def score(value):
    if 0 <= value <= 0.4:
        return 1
    elif 0.4 < value <= 0.7:
        return 2
    elif 0.7 < value <= 1:
        return 3
    elif 1 < value <= 1.3:
        return 4
    else:
        return 5

# לשיפור היעילות vectorization-שימוש ב
score_vectorized = np.vectorize(score)
```

שלב 4: חישוב הציון הכולל

הציון הכולל של כל מתנדב חושב על פי נוסחה משוקללת, שהתבססה על היחסים המחושבים בפרמטרים השונים. בנוסחה זו שולבו משקלים שונים עבור פרמטרים כמו זמן טיפול (t_ratio), דירוג לקוח (r_ratio), ואחוזי הצלחה (w_ratio). בנוסף, הוסף לציון זה ציון ה- $TF-IDF$ של הקטגוריה, מה שהבטיח שהתחשבות ברלוונטיות הקטגוריה תשפיע על הציון הסופי של המתנדב.

```
# חישוב הציון הכולל של כל מתנדב לפי המשוואה
user_data_df['rank'] = (user_data_df['t_score'] * 0.5 + user_data_df['r_score'] * 0.1
                        + user_data_df['w_score'] * 0.4 + user_data_df['tfidf_score']) * np.log1p(user_data_df['אירועים'])
```

שלב 5: נרמול הציונים לפי קטגוריות

בשלב הסופי, נורמלו הציונים הסופיים לכל קטגוריה, כדי להבטיח שהשוואת הביצועים תתבצע בצורה הוגנת גם בין קטגוריות שונות. נרמול זה התבצע באופן שמבטיח כי כל ציון סופי נע בין 1 ל-100, בהתאם לפערים בתוך הקטגוריה.

```
# שלב 6: נרמול הציונים בין 1 ל-100 עבור כל קטגוריה בנפרד
def normalize_category(group):
    min_rank = group['rank'].min()
    max_rank = group['rank'].max()
    group['final score'] = 1 + ((group['rank'] - min_rank) * (100 - 1) / (max_rank - min_rank))
    return group

user_data_df = user_data_df.groupby('קטגוריה').apply(normalize_category)

user_data_df
```

יישום בפועל (Distributed Matching-By-Clone Hungarian-Based Algorithm for Task Allocation of Multiagent Systems)

שלב חישוב מרחקים והוספת משימות דמה למטריצת העלויות

במסגרת הפרויקט, נדרשנו להתמודד עם מצב שבו מספר המשימות אינו שווה למספר המתנדבים. כדי להבטיח פעולה תקינה של האלגוריתם ההונגרי, ביצענו חישוב מרחקים והוספנו משימות דמה (Dummy Tasks) למטריצת העלויות, מה שהפך אותה לריבועית. תהליך זה מאפשר לכל משימה אמיתית לקבל את ההתאמה האופטימלית ביותר, תוך שמירה על מבנה ריבועי שמבטיח פעולה יעילה של האלגוריתם. לאחר הוספת המשימות הדמה, האלגוריתם ההונגרי נורמל והקצה משימות למתנדבים, בצורה שמזערה את העלות הכוללת או מקסמה את הרלוונטיות. יישום זה מדגים את



ההתאמה למצבי חוסר התאמה במספרים, בעיה נפוצה במערכות שיבוץ מתנדבים. יישמנו את הגישה המוצעת לדיוק ומיטוב ההתאמה מהמחקר, הקוד שופר בהתאם כדי להתמודד עם מצבים אלה בצורה אופטימלית.

יישום בפועל (Online Algorithms for Matching Platforms with Multi-Channel Traffic)

שלב 1: יישום סינון גרידי (Greedy Filtering) עבור מתנדבים ומשימות

בשלב זה, השתמשנו בטכניקה של סינון גרידי כדי לצמצם את כמות ההתאמות הפוטנציאליות על ידי סינון מתנדבים שנמצאים במרחק רב מדי מהמשימה. המטרה הייתה להפחית את עומס החישובים על האלגוריתם ההונגרי, כך שיפעל בצורה יעילה יותר.

```
# המצב הקיים: במצב הנוכחי, אין וידוא שהמטריצה היא ריבועית.
# השיפור בזכות המאמר: במקרה שמספר המשימות קטן ממספר המתנדבים, נוודא שהמטריצה תהיה ריבועית.
# על ידי הוספת שורות/עמודות עם ערכים גבוהים מאוד (משימות דמה) למטריצה.
n, m = cost_matrix.shape
max_dim = max(n, m)
if n < max_dim:
    cost_matrix = np.vstack([cost_matrix, np.full((max_dim - n, m), 1e6)])
if m < max_dim:
    cost_matrix = np.hstack([cost_matrix, np.full((max_dim, max_dim - m), 1e6)])

# ביצוע האלגוריתם ההונגרי
row_ind, col_ind = linear_sum_assignment(cost_matrix)
```

האלגוריתם בודק את כל ההתאמות האפשריות ומסנן את אלה שבהן המרחק בין המתנדב למשימה עולה על 60 ק"מ. מדובר בסינון ראשוני שמקטין את מטריצת העלויות ($cost\ matrix$) ומשאיר רק את ההתאמות עם פוטנציאל גבוה.

הגישה מפחיתה משמעותית את מספר ההתאמות שצריך לבדוק באלגוריתם המרכזי, מה שמאפשר חישוב יעיל יותר ומקטין את זמן הריצה.

```
# פונקציה לביצוע סינון גרידי עם סף מרחק של 60 ק"מ
def greedy_filter(vol_df, call_df):
    filtered_vol_df = vol_df.copy()
    filtered_vol_df['min_distance'] = filtered_vol_df.apply(
        lambda vol_row: min(geodesic(vol_row['כונן'], call_row['תקלה']).kilometers for _, call_row in call_df.iterrows()),
        axis=1
    )
    # סינון מתנדבים שהם במרחק של מעל 60 ק"מ כיוון שהוכח שמעל מרחק זה הסיוע אינו רלוונטי
    return filtered_vol_df[filtered_vol_df['min_distance'] <= 60]

# שימוש בסינון גרידי לפני הפעלת האלגוריתם ההונגרי
vol_df = greedy_filter(vol_df, call_df)
```

שלב 2: יישום הגישה ההיברידית

כדי לשפר עוד יותר את ביצועי האלגוריתם ההונגרי, יישמנו גישה היברידית שבה האלגוריתם ההונגרי מופעל רק על הקבוצה המצומצמת של מתנדבים שעברו את הסינון הגרידי. כך,



האלגוריתם מתמקד רק בהתאמות הפוטנציאליות הרלוונטיות ביותר, מה שמפחית את העומס על המערכת ומשפר את הדיוק.

```
# חישוב מרחקים בין המיקומים והפעלת האלגוריתם ההונגרי על הקבוצה המצומצמת
results = []
# חישוב המרחקים והפעלת האלגוריתם ההונגרי עם יחס תחרותי
total_hungarian_cost = 0
total_greedy_cost = 0

for category, call_df in calling_by_category:
    if category not in volunteer_by_category.groups:
        continue

    vol_df = volunteer_by_category.get_group(category)

    # הפעלת סינון מקדים בגישה ההיברידית
    vol_df = greedy_filter(vol_df, call_df)

    call_df.loc[:, 'מיקום תקלה'] = call_df['מיקום תקלה'].apply(eval)
    vol_df.loc[:, 'מיקום כונן'] = vol_df['מיקום כונן'].apply(eval)
```

לאחר הסינון הגרידי, האלגוריתם ההונגרי מופעל על קבוצת ההתאמות המצומצמת. האלגוריתם ההונגרי מתבסס על מטריצת עלויות ריבועית, שנבנית ממספר פרמטרים כמו מרחק גיאוגרפי וציוני מתנדבים (*normalized volunteer scores*). מטריצה זו מנורמלת על בסיס המרחק והציון שניתן למתנדב בהתחשב בפרמטרים של זמן טיפול, דירוג לקוח ואחוזי הצלחה.

שלב 3: חישוב יחס תחרותי (Competitive Ratio)

כדי להעריך את יעילות האלגוריתם ההונגרי בהשוואה לאלגוריתם גרידי, הוספנו שלב שבו מחושב יחס תחרותי בין העלויות של שני האלגוריתמים. המדד הזה מאפשר לבדוק את הביצועים של האלגוריתם ההונגרי במונחים של עלות כוללת לעומת פתרון פשוט יותר.

```
# הפעלת האלגוריתם הגרידי וספירת העלות הכוללת
greedy_assignments, greedy_cost = greedy_assignment(vol_df, call_df)
total_greedy_cost += greedy_cost

# חישוב יחס תחרותי
if total_greedy_cost > 0:
    competitive_ratio = total_hungarian_cost / total_greedy_cost
else:
    competitive_ratio = float('inf') # יחס אינסופי במידה ואין עלות גרידית

# הצגת יחס תחרותי
print(f"יחס תחרותי: {competitive_ratio}")

# עם העמודות בסדר הרצוי DataFrame יצירת
results_df = pd.DataFrame(results, columns=['מספר קריאה', 'מזהה פונה', 'מזהה כונן', 'קטגוריה', 'מרחק בק"מ', 'עלות'])

# הדפסת התוצאות
results_df
```

יחס תחרותי: 35.8407434032733

עלות	מרחק בק"מ	קטגוריה	מזהה כונן	מזהה פונה	מספר קריאה
36.474327	17.886288	שמן-מים-דלק	משה 6664	מתי 7340	36259
0					0

היחס התחרותי (Competitive Ratio) שהופק על ידי האלגוריתם משווה בין עלות השיבוץ הכוללת (*total_hungarian_cost*) באמצעות האלגוריתם ההונגרי לבין עלות השיבוץ באמצעות אלגוריתם גרידי (*total_greedy_cost*).



הפלט מראה יחס תחרותי גבוה מאוד 35.84. נוכל להסיק שהאלגוריתם ההונגרי היה הרבה יותר טוב במציאת התאמות יעילות, בהתבסס על המשקלים והציונים המוגדרים במטריצה.

ככל שהיחס התחרותי גבוה יותר, כך האלגוריתם ההונגרי הצליח להוריד את העלויות בצורה יותר משמעותית בהשוואה לאלגוריתם הגרדי, מה שמצביע על כך שהגישה שיישמו הונגרי עם גרידי כסינון מקדים היא אופטימלית עבור המערכת שלנו.

בזכות היישום הוכח השיפור בפועל :

שיפור בזמני התגובה : אחד מהאתגרים היה לשפר את זמני התגובה של המערכת. באמצעות הטמעת הסינון הגרידי והגישה ההיברידית, הצלחנו לצמצם את כמות ההתאמות הפוטנציאליות על ידי סינון ראשוני של מתנדבים בעלי מרחק רב מדי. הדבר הוביל להקטנת העומס החישובי על האלגוריתם ההונגרי ושיפר את זמני התגובה של המערכת.

מדדת היעילות באמצעות יחס תחרותי : לאחר הטמעת השיפורים, החלטנו להשתמש במדד יחס תחרותי כדי להעריך את ביצועי האלגוריתם ההונגרי ביחס לשיטות בסיסיות יותר. המדד הראה על שיפור משמעותי באיכות ההתאמות שביצענו, מה שהוביל לתוצאות מדויקות יותר ולשיפור באחוזי ההצלחה.

An Optimization Model for Volunteer Assignments in Humanitarian (Organizations)

במסגרת הפרויקט, התמודדנו עם האתגרים הקשורים למגבלות זמינות המתנדבים כדי להבטיח שהמודל יפעל בצורה תקינה ויספק הקצאות מדויקות ורלוונטיות. לשם כך, הוספנו למערכת עמודה ייחודית המצביעה על זמינות המתנדבים. כאשר מתנדב מסומן כזמין, המערכת לא רק מזהה את זמינותו אלא גם שולחת את המיקום הנוכחי שלו. תהליך זה מדגים כיצד המערכת של ידידים פועלת : כאשר מתנדב מעוניין לקחת את הקריאה, המערכת מאמתת את זמינותו ומספקת את מיקומו הנוכחי, מה שמאפשר הקצאה מהירה ומדויקת בהתאם לצרכים האמיתיים של האירוע. כך, אנו מבטיחים שהמתנדבים המוקצים יהיו זמינים ויכולים להשתתף במצבי חירום באופן מיידי.

יישמו את הגישה המוצעת לדיוק ומיטוב ההתאמה מהמחקר, הקוד שופר בהתאם כדי להתמודד עם מצבים אלה בצורה אופטימלית.

מיקום כינון	זמינות	final score	קטגוריה	מזהה כינון	
32.78982551746709,34.38239472604465	זמין	7.285183	שמן-מים-דלק	דוד 9930	21784
30.013245176821847,35.86810066835339	זמין	17.855717	שמן-מים-דלק	עדי 9930	21785
None	לא זמין	10.142084	שמן-מים-דלק	יהודה 9933	21786

כדי להבטיח צמצום אי-ההתאמה בין המתנדב למשימה ולהשיג ביצוע מיטבי של כל משימה, ביצענו חישוב מתמטי של ציון המתנדב. הציון חושב על בסיס מספר האירועים שהמתנדב ביצע בהצלחה לפי קטגוריה ספציפית. באמצעות מדד זה, אנו יכולים להבטיח שהמתנדב הנבחר בעל ניסיון ומיומנות בפתרון הבעיות הספציפיות הנדרשות, מה שמוביל לתוצאות מדויקות ויעילות יותר.



```
# לשיפור היעילות - שימוש ב- vectorization
score_vectorized = np.vectorize(score)

user_data_df['t_score'] = score_vectorized(user_data_df['t_ratio'])
user_data_df['r_score'] = score_vectorized(user_data_df['r_ratio'])
user_data_df['w_score'] = score_vectorized(user_data_df['w_ratio'])

# שלב 5: חישוב הציון הכולל של כל מתנדב לפי המשוואה
# 'num_events' נניח שהעמודה שמכילה את מספר האירועים נקראה
user_data_df['rank'] = (user_data_df['t_score'] * 0.5 + user_data_df['r_score'] * 0.1 + user_data_df['w_score'] * 0.4) * np.log1p(user_data_df[['כמות אירועים']])
```

כפי שניתן לראות, הציון הסופי של המתנדב עולה בהתאם לאחוזי ההצלחה שלו, כשהגידול בציון מתבטא ככל שמספר האירועים המבוצעים בהצלחה גדל. הדבר מעיד על רמת ניסיון גבוהה יותר של המתנדב בתחום הספציפי. ציון זה נחשב בהתחשבות בקטגוריה הרלוונטית, מה שמבטיח שהמתנדב הנבחר הוא בעל ניסיון מוכח ויכולת גבוהה לפתרון הבעיות הנדרשות.

```
[ ] user_data_df[user_data_df['קטגוריה'] == 'שמן-מים-דלק'].sort_values(by='final score')
```

```
/usr/local/lib/python3.10/dist-packages/pandas/core/nanops.py:1010: RuntimeWarning: invalid value encountered in subtract
  sqr = _ensure_numeric((avg - values) ** 2)
```

		מזהה כיון	קטגוריה	ממוצע זמן טיפול בתקלה	כמות אירועים	ממוצע דירוג לקוח	אחוזי הצלחה	אירועים מוצלחים	t_ratio	r_ratio	w_ratio	t_score	r_score	w_score	rank	final score
קטגוריה																
שמן-מים-דלק	13257	5842	שמן-מים-דלק	94.000000	1	0.0	0.000000	2.000000	0.245178	0.663776	0.000000	1	2	1	0.762462	1.000000
	6948	2860	שמן-מים-דלק	65.000000	1	0.0	0.000000	2.200000	0.354565	0.730154	0.000000	1	3	1	0.831777	1.285690
	6615	2702	שמן-מים-דלק	104.000000	1	0.0	0.000000	2.946809	0.221603	0.978011	0.000000	1	3	1	0.831777	1.285690

13.2 קוד לבניית ממשק

בניית ממשק

```
[ ] # import streamlit as st
import pandas as pd
import numpy as np
from geopy.distance import geodesic
from scipy.optimize import linear_sum_assignment
import requests

# הורדת CSV מ-GitHub
url = 'https://raw.githubusercontent.com/Celinmi/Final-Project/main/volunteer_pool.csv'
Volunteer_pool = pd.read_csv(url)
logo_url = 'https://raw.githubusercontent.com/Celinmi/Final-Project/main/ידידים.jpg'
back_ground_url = 'https://raw.githubusercontent.com/Celinmi/Final-Project/main/ריקע.jpg'
categories = ['קורונה - קניית אוכל', 'קורונה - שינוע מזון', 'שינוע', 'שמן-מים-דלק', 'אחר', 'דלת', 'קורונה - קניית תרופות', 'רכב נעול', 'פנצ'ר', 'הנעה']

st.markdown(
    f"""
    <style>
    .main {{
        background-image: url("{back_ground_url}");
        background-size: 90vw 100vh; # This sets the size to cover 100% of the viewport width and height
        background-position: center;
        background-repeat: no-repeat;
    }}
    .stButton button {{
        background-color: rgba(255, 255, 255, 0.8);
        color: black;
        border-radius: 12px;
    }}
    """
)
```



```
1    }}
    .stTextInput, .stNumberInput input {{
        color: black;
    }}
    .title {{
        background-color: rgba(255, 255, 255, 0.8); /* רקע בהיר */
        color: black !important;
        padding: 10px;
        border-radius: 10px;
    }}
    .header {{
        background-color: rgba(255, 255, 255, 0.8); /* רקע בהיר */
        color: black !important;
        padding: 5px;
        border-radius: 5px;
    }}
    .css-10trblm, .css-1v3fvcr p {{
        color: white !important;
    }}
    .stTitle, .stHeader, .stSubheader, .stMarkdown, .stText, .stNumberInput label, .stDateInput label {{
        color: white !important;
    }}
</style>
"""
unsafe_allow_html=True
)

st.image(logo_url, width=200)

st.markdown('<h1 class="title">בחירת מתנדב אופטימלי</h1>', unsafe_allow_html=True)

st.markdown('<h2 class="header">בחירת פרטים לבחירת כונו</h2>', unsafe_allow_html=True)
```

```
# קליטים מהמשתמש
location = st.text_input("מיקום תקלה - קו אורד, קו רוחב לדוגמה 32.085 , 34.784")
caller_id = st.text_input("הכנס מזהה ייחודי של הפונה - שם פרטי ו 4 ספרות אחרונות של ת.ז.")
category = st.selectbox("בחר קטגוריה של התקלה", categories)

if st.button("בחר מתנדב"):
    if location and caller_id:
        Volunteer_pool = Volunteer_pool[Volunteer_pool['זמין'] == 'זמין']

        # הכנת פול של פניות
        calling_pool = pd.DataFrame({
            'מזהה פונה': [caller_id],
            'קטגוריה': [category],
            'מיקום תקלה': [location]
        })

        # הכנת מטריצת העלויות
        cost_matrix = []
        for call_index, call_row in calling_pool.iterrows():
            row = []
            call_location = tuple(map(float, call_row['מיקום תקלה'].split(',')))
            for vol_index, vol_row in Volunteer_pool.iterrows():
                vol_location = tuple(map(float, vol_row['מיקום כונו'].split(',')))
                distance = geodesic(call_location, vol_location).kilometers
                row.append(distance)
            cost_matrix.append(row)

        cost_matrix = np.array(cost_matrix)

        # נרמול ציוני המתנדבים
        def normalize_scores(scores):
            return (scores - scores.min()) / (scores.max() - scores.min())
```



```

if m < max_dim:
    cost_matrix = np.hstack([cost_matrix, np.full((max_dim, max_dim - m), 1e6)])

# ביצוע האלגוריתם ההונגרי
row_ind, col_ind = linear_sum_assignment(cost_matrix)

# קבלת התוצאות
results = []
for row, col in zip(row_ind, col_ind):
    if row < len(calling_pool) and col < len(volunteer_pool):
        distance = geodesic(calling_pool.iloc[row]['מיקום תקלה'], volunteer_pool.iloc[col]['מיקום כונן']).kilometers
        assignment = (
            calling_pool.iloc[row]['מזהה פונה'],
            volunteer_pool.iloc[col]['מזהה כונן'],
            category,
            distance
        )
        results.append(assignment)

results_df = pd.DataFrame(results, columns=['מזהה פונה', 'מזהה כונן', 'קטגוריה', 'מרחק בק"מ'])

# הצגת התוצאות
st.write("תוצאות הבחירה:")
st.dataframe(results_df)

else:
    st.write("יש להזין את כל השדות.")

```

תוצאות הבחירה:

מרחק בק"מ	קטגוריה	מזהה כונן	מזהה פונה
1.294065	הנעה	יוסף 0220	12345

קישור לממשק

שמנו לב שעקב מניפולציות על הנתונים, ישנם ערכים חריגים/חסרים.

על מנת לקבל תוצאות אמיתיות, נבדוק אותם שנית ונבין את המשמעויות החדשות שלהם

נספח פירוט חקר נתונים חסרים
ולא מנומלים:

```

missing_values_clean_data = clean_data_df.isna().sum()
zero_values_clean_data = (clean_data_df == 0).sum()
empty_values_clean_data = (clean_data_df == ' ').sum()
# הצגת התוצאות
print(missing_values_clean_data)

print(zero_values_clean_data)

print(empty_values_clean_data)

```

0	זמן	0	זמן
0	כתובת	0	כתובת
1733	רכב	0	רכב
0	קטגוריה	0	קטגוריה
0	תת קטגוריה	0	תת קטגוריה
0	מזהה פונה	0	מזהה פונה
1	מזהה כונן	5160	מזהה כונן
1	זמן לקיחה	0	זמן לקיחה
1	זמן סגירה	0	זמן סגירה
1	סטטוס אירוע	0	סטטוס אירוע
1	זמן טיפול	0	זמן טיפול
1	זמן טיפול ממוצע	0	זמן טיפול ממוצע
1	חריג זמן טיפול	0	חריג זמן טיפול
1	קו רוחב תקלה	0	קו רוחב תקלה
2	קו אורך תקלה	0	קו אורך תקלה
2	קו רוחב מתנדב	0	קו רוחב מתנדב
2	קו אורך מתנדב	0	קו אורך מתנדב
2	מרחק בק"מ	0	מרחק בק"מ
2	דירוג לקוח	0	דירוג לקוח



ניצור העתק לנתונים על מנת לא לפגוע בתקינותם, לאחר סיום ניקוי הנתונים נחזור להשתמש בקובץ המעודכן.

```
[88] clean_data=clean_data_df.copy()
```

```
[ ] # זיהוי הערכים החסרים בעמודות רכב
missing_cars = clean_data['רכב'].isna().sum()
print(f"Missing 'רכב' values before handling: {missing_cars}")
# זיהוי ערכים חסרים אחרי הטיפול
missing_values_clean_data_after = clean_data.isna().sum()

missing_values_clean_data_after
```

Missing 'רכב' values before handling: 1733

0	0
זמן	0
כתובת	0
רכב	0
קטגוריה	0
תת קטגוריה	0
מזהה פונה	0

בעמודות בהן מדובר בערך 1, נבחר להסיר את החסר

מתגמד לעומת כמות השורות והנתונים שיש לנו

עכשיו נשארו עם שתי בעיות עיקריות ונצטרך לנתח את הפער:

- יש 8138 ערכים שהם 0 בעמודות זמן טיפול (zero_values_clean_data) ערכים שהם 0.
- יש 5160 ערכים שהם מחוזות ריקות בעמודות מזהה כונן. ננסה להבין את ההקשרים וממה הבעיה: (empty_values_clean_data) ערכים שהם מחוזות ריקות. נוצרת והאם ישנה השפעה על נתונים אחרים או שניתן להסיק מגמתיים מסויימת, נוודא שאין מקרה או מסקנה חשובה שפספסנו

ניתוח הערכים שהם 0 בעמודות זמן טיפול:

בעזרת תצוגת ויזואלית ניתן להבין האם יש קטגוריות מסוימות או כוננים מסוימים שבהם זמן הטיפול הוא אפס לעיתים קרובות יותר. ניתן לזהות האם יש התפלגויות חריגות או קשרים מעניינים בין קטגוריות/כוננים לבין זמן הטיפול.

Correlation Heatmap: נראה את הקורלציה בין הערכים שהם 0 בעמודות זמן טיפול לבין יתר העמודות.

```
[60] # זיהוי הערכים שהם 0 בעמודות 'זמן טיפול'
zero_treatment_time = clean_data[clean_data['זמן טיפול'] == 0]
zero_treatment_time
```

זמן טיפול מטופע	זמן טיפול	סטטוס אירוע	זמן סגירת	זמן לקיחה	מזהה סניף	מזהה פונה	קטגוריה	קטגוריה	רכב	כתובת	זמן
25.305340	0.0	נסגר בהצלחה	2022-01-01 18:46:00.000000000	2022-01-01 18:46:00.000000000	2147	2147	אברהם	אברהם	פנלר	הריטב"א 26, מונדאו עילית	2022-01-01 18:21:00
18.064785	0.0	נסגר בהצלחה	2022-01-01 18:44:00.000000000	2022-01-01 18:44:00.000000000	7041	1952	אהרון	דוד	הנעה	עמיחי 45, קריית אפלנד עקרון, ישראל	2022-01-01 18:42:00
18.064785	0.0	נסגר בהצלחה	2022-01-01 19:17:00.000000000	2022-01-01 19:17:00.000000000	0498	9997	אלעד	אלעד	הנעה	דב גרונר 55, בני ברק,	2022-01-01 19:14:00



התוצאה מראה כי שורות שבהן זמן הטיפול הוא 0 והפרש הזמן בין זמן לקיחה לזמן סגירה הוא קטן מ-1 דקה. מדובר כנראה במקרים שבהם האירוע נסגר מהר מאוד, אולי בגלל ביטול האירוע או שהבעיה נפתרה מיד. נראה כי יש שורות בהן "מזהה כונן" זהה ל"מזהה פונה"-הנחת יסוד היא כי מדובר בטעות אנוש, אך נחקור האם קיימת השפעה על הנתונים.

על מנת להבין טוב יותר את המקרים הללו, ניתן לבדוק את סטטוס האירוע ואולי גם משתנים נוספים כמו תת קטגוריה או מזהה כונן.

```
[64] # שורות שבהן מזהה הכונן ומזהה הפונה זהים
same_id_rows = close_times[close_times['מזהה כונן'] == close_times['מזהה פונה']]
same_id_rows
```

זמן טיפול	זמן סגירה	זמן לקיחה	מזהה כונן	מזהה פונה	תת קטגוריה	קטגוריה	רכב	כתובת	זמן
0.0	נסגר בהצלחה	2022-01-01 18:46:00	2022-01-01 18:46:00	2147	2147	אברהם	אברהם	2147	2022-01-01 18:21:00
0.0	נסגר בהצלחה	2022-01-05 10:51:00	2022-01-05 10:51:00	5529	5529	צבי	צבי	5529	2022-01-05 10:41:00
0.0	נסגר בהצלחה	2022-01-10 16:03:00	2022-01-10 16:03:00	2221	2221	שימי	שימי	2221	2022-01-10 15:58:00

יש תופעה שבה זמן הלקיחה וזמן הסגירה הם קרובים או זהים כאשר זמן הטיפול הוא 0. נבדוק וננסה להבין את המצב יותר לעומק.

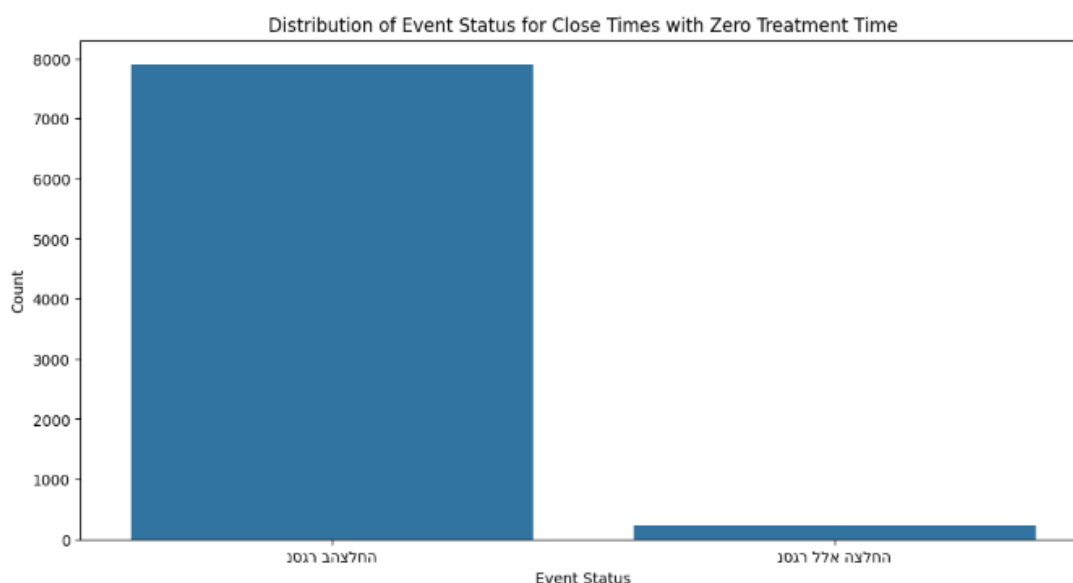
בדיקה אם זמן הסגירה אכן נרשם נכון או אם יש שגיאה בנתונים. ניתוח הנתונים לפי קטגוריה ומזהה כונן כדי לראות אם ישנם דפוסים חוזרים. בדיקת השפעת הזמן של כל שורה והשוואה למשתנים אחרים.

```
# הצגת כל השורות בהן זמן הטיפול הוא 0 וקטן מ-1 דקה
# Creating a Timedelta object for 1 minute
close_times = clean_data[(clean_data['זמן טיפול'] == 0) & (clean_data['זמן טיפול'] < 1)]
close_times
```

זמן טיפול	זמן סגירה	זמן לקיחה	מזהה כונן	מזהה פונה	תת קטגוריה	קטגוריה	רכב	כתובת	זמן
25.305340	0.0	נסגר בהצלחה	2022-01-01 18:46:00	2022-01-01 18:46:00	2147	2147	אברהם	אברהם	2022-01-01 18:21:00
18.064785	0.0	נסגר בהצלחה	2022-01-01 18:44:00	2022-01-01 18:44:00	7041	1952	אהרון	דוד	2022-01-01 18:42:00
18.064785	0.0	נסגר בהצלחה	2022-01-01 19:17:00	2022-01-01 19:17:00	0498	9997	אלעד	אלעד	2022-01-01 19:14:00
18.064785	0.0	נסגר בהצלחה	2022-01-01 19:46:00	2022-01-01 19:46:00	3829	1942	שמאל	אשר	2022-01-01 19:26:00



```
# הצגת התפלגות סטטוס אירוע שבהם זמן הטיפול הוא 0 והפרש הזמן קטן מ-1 דקה
plt.figure(figsize=(12, 6))
sns.countplot(data=close_times, x='סטטוס אירוע')
plt.title('Distribution of Event Status for Close Times with Zero Treatment Time')
plt.xlabel('Event Status')
plt.ylabel('Count')
plt.show()
```



```
[65] # חישוב האחוז של שורות כאלה מתוך כלל השורות
percentage_same_id = (len(same_id_rows) / len(clean_data)) * 100
percentage_same_id
```

0.019510608893585888

נראה כי מדובר סה"כ ב-60 שורות ואחוז מזערי מהנתונים, מאשש את ההנחה שלנו כי מדובר בטעות אנוש, השורות שבהן מזהה הכונן זהה למזהה הפונה אינן מביאות ערך מוסף והן מהוות אחוז קטן מהנתונים (כ-0.02%), אפשר להוריד אותן.

על מנת להבין טוב יותר את המקרים הללו, ניתן לבדוק את סטטוס האירוע ואולי גם משתנים נוספים כמו תת קטגוריה או מזהה כונן.

```
# הסרת השורות שבהן מזהה הכונן זהה למזהה הפונה
clean_data = clean_data[clean_data['מזהה כונן'] != clean_data['מזהה פונה']]

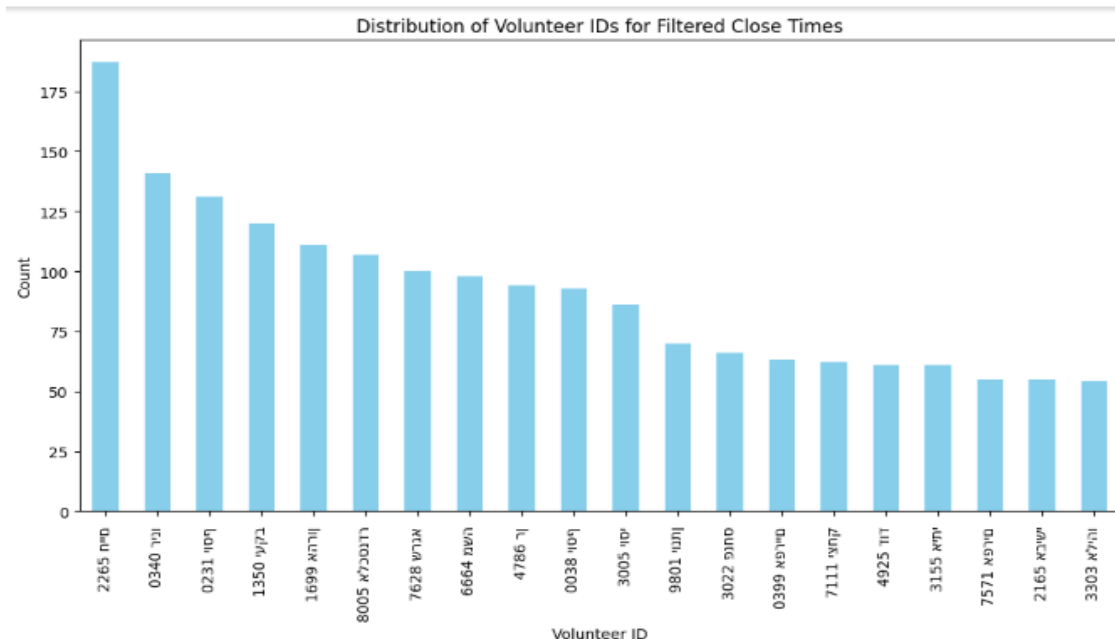
# הצגת נתונים לאחר ההסרה
print(f"מספר השורות לאחר ההסרה: {len(clean_data)}")
```

מספר השורות לאחר ההסרה: 307340



Distribution of Volunteer IDs for Filtered Close Times:

מזהה כונן	
2265 187	חיים
0340 141	דניאל
0231 131	יוסף
1350 120	יעקב
1699 111	אהרון
8005 107	אלכסנדר
7628 100	שרה
6664 98	משה
4786 94	דן
0038 93	יוסף
3005 86	יוסי
9801 70	יונתן
3022 66	פנחס
0399 63	אפרים
7111 62	יצחק
4925 61	דוד
3155 61	איתי
7571 55	אפרים
2165 55	אבישי
3303 54	אליהו

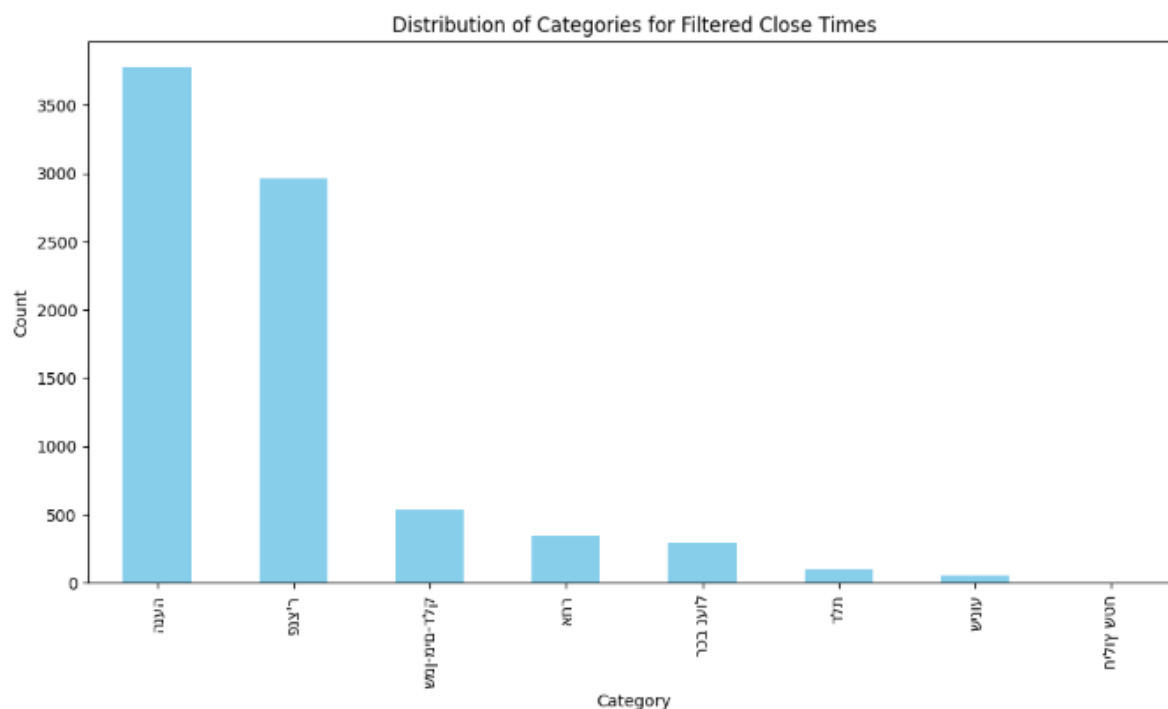


ניתוח לפי קטגוריה ומזהה כונן

הקוד מציג את התפלגות הקטגוריות בהן זמן הטיפול הוא 0 וקטן מדקה ומציג את התפלגות מזהה הכוננים והקטגוריות עבור אותם מקרים. מוצגים גרפים להמחשת ההתפלגות.

Distribution of Categories

קטגוריה	
3776	הנעה
2963	פנצ'ר
531	שמן-מים-דלק
346	אחר
293	רכב נעול
91	דלת
51	שיטות
2	חילוקי שטח





נשים לב שמבחינת קטגוריות: בעייתיות ביותר מבחינת זמני הטיפול הן מסוג "הנעה" ומסוג "פנצ'ר", ייתכן שזה קשור לכך שמקרים אלו הם קלים יחסית לטיפול או שיש יותר קריאות במקרים אלו.

מבחינת הכוננים עצמם: יש בולטים מבחינת חריגה: מזהים: "חיים 2265", "רינו 0340", "יוסף 0231", "יעקב 1350", יכול להעיד על כך שכוננים אלו עשויים להיות זריזים במיוחד או שבחלק מהמקרים הם פשוט לא התערבו (ייתכן שסגרו קריאה מבלי לטפל בפועל). ננסה להשליך על הקטגוריות שנחשבות חריגות מבחינת זמני טיפול יחד עם הכוננים הלרוונטיים

```
[70] filtered_data = close_times_filtered[
    (close_times_filtered['סטטוס אירוע'] == 'נסגר בהצלחה') &
    (close_times_filtered['קטגוריה'].isin(['הנעה', 'פנצ'ר']))
]

# יצירת טבלה מסוננת לפי מזהה כונן
grouped_data = filtered_data.groupby(['מזהה כונן', 'קטגוריה']).size().unstack(fill_value=0)
# מציאת הכוננים עם הכי הרבה אירועים שנסגרו בהצלחה
top_volunteers = grouped_data.sum(axis=1).sort_values(ascending=False).head(20).index

# הצגת הטבלה עבור הכוננים המובילים
top_grouped_data = grouped_data.loc[top_volunteers]
# הצגת הטבלה
top_grouped_data
```



מזהה כונן	קטגוריה	הנעה	פנצ'ר
2265	חיים	91	72
0340	רינו	81	23
0231	יוסף	52	50
1350	יעקב	51	50
8005	אלכסנדר	45	45





נבדוק האם הבעיה תואמת בטבלה הראשית הנוגעת לנתוני המתנדבים והפונים

```
[71] # יצירת טבלה מסוננת לפי מזהה כונו וקטגוריה מהשלב הקודמים
grouped_data = filtered_data.groupby(['מזהה כונו', 'קטגוריה']).size().unstack(fill_value=0)

# מציאת הכוונים עם הכי הרבה אירועים שנסגרו בהצלחה
top_volunteers = grouped_data.sum(axis=1).sort_values(ascending=False).head(20).index

# הצגת הטבלה עבור הכוונים המובילים
top_grouped_data = grouped_data.loc[top_volunteers]

# והוספת עמודת "אחוזי הצלחה" user_data_table מיזוג הנתונים עם
merged_data = user_data_table[user_data_table['מזהה כונו'].isin(top_volunteers)]
merged_data = merged_data[merged_data['קטגוריה'].isin(['הנעה', 'ר', 'פנצ\ר'])]
# הצגת הטבלה
merged_data
```



קוח אחוזי הצלחה אירועים מוצלחים כמות אירועים ממוצע זמן טיפול בתקלה קטגוריה מזהה כונו

קטגוריה

הנעה	169	0038	הנעה	11.838861	188	178.0	94.680851
	775	0231	הנעה	10.820512	285	285.0	100.000000
	1094	0340	הנעה	14.157462	913	911.0	99.780942
	1240	0399	הנעה	8.658248	122	121.0	99.180328
	3592	1350	הנעה	12.921450	229	179.0	90.404040

נראה שאחוזי הצלחה גבוהים יחסית בכוונים שנחשבים בעייתיים בקטגוריות הרלוונטיות, ולכן נשאיר את הנתונים הללו. בשיחה עם אחד המתנדבים דרך העמותה, הובן כי הבעיה העיקרית שהיא הנעה לרוב נפתרת מעצמה והפונה מבטל את הפנייה במהרה, לכן נסיק כי מדובר בשורות שאינן בעייתיות.

בעיה 2: ערכים ריקים במזהי הכוונים

```
] empty_clean_data = (clean_data == ' ').sum()
empty_clean_data
```

	0
זמן	0
כתובת	0
רכב	0
קטגוריה	0
תת קטגוריה	0
מזהה פונה	0
מזהה כונו	5160



```
# בדיקה האם יש ערכים ריקים בצורות שונות
empty_clean_data = (clean_data['מזהה כונן'].isin(['', ' ', None])).sum()
print(f"Number of empty values in 'מזהה כונן': {empty_clean_data}")

# החלפת כל הצורות של ערכים ריקים בערך מציין "לא ידוע"
clean_data['מזהה כונן'].replace(['', ' ', None], 'לא ידוע', inplace=True)

# בדיקה חוזרת של ערכים ריקים
empty_clean_data = (clean_data['מזהה כונן'].isin(['', ' ', None])).sum()
print(f"Number of empty values in 'מזהה כונן' after replacement: {empty_clean_data}")

# הצגת שורות עם ערכים ריקים במזהה כונן לאחר ההחלפה
empty_connan_id = clean_data[clean_data['מזהה כונן'] == 'לא ידוע']
print(empty_connan_id)

# הסרת מרווחים ריקים בעמודה "מזהה כונן"
clean_data['מזהה כונן'] = clean_data['מזהה כונן'].str.strip()

# בדיקה אם יש ערכים ריקים לאחר הסרת מרווחים ריקים
empty_clean_data = (clean_data['מזהה כונן'].isin(['', ' ', None])).sum()
print(f"Number of empty values in 'מזהה כונן' after stripping: {empty_clean_data}")
```

מזהה כונן: 5160
Number of empty values in 'מזהה כונן' after replacement: 0
Number of empty values in 'מזהה כונן' after stripping: 0

```
[80] # הסרת השורות שבהן מזהה הכונן הוא "לא ידוע"
clean_data = clean_data[clean_data['מזהה כונן'] != 'לא ידוע']

# בדיקה חוזרת לוודא שהשורות הוסרו
empty_clean_data_filtered = (clean_data['מזהה כונן'] == 'לא ידוע').sum()
print(f"Number of rows with 'לא ידוע' in 'מזהה כונן' after filtering: {empty_clean_data_filtered}")
```

Number of rows with 'לא ידוע' in 'מזהה כונן' after filtering: 0

שתי הבעיות העיקריות טופלו, ניתן להמשיך לבניית המודל וקבלת תוצאות אמיתיות

נשים לב, שאחרי חקר מעמיק מתגלה כי רוב השורות הבעייתיות נחתמות בסטטוס תקלה שהוחזרה. בשיחה עם הארגון וקביעת המדדים סטטוס אירוע שהוחזר אינו נראה כרלוונטי לצורך יעול והשיפור של הקצאת המתנדב, נוסף לעניין כי אין מזהה כונן שניתן לאפיין כמתנדב אופטימלי ולכן נראה בשורות האלו נתונים שלא ניתן ללמוד/להסיק מהם דבר. בהסכמה עם הארגון נוריד את הנתונים האלו ונמשיך לחקור עם נתונים רלוונטיים להמשך התהליך.

```
[79] # סינון הנתונים לפי סטטוס אירוע "הוחזר"
returned_events = unknown_volunteer_data[unknown_volunteer_data['סטטוס אירוע'] == 'הוחזר']

# הצגת הנתונים
returned_events
```



זמן טיפול	סטטוס אירוע	זמן סגירה	זמן לקיחה	מזהה כונן	מזהה פונה	תת קטגוריה	קטגוריה	רמז	כתובת	זמן
18.046653	הוחזר	2022-01-01 19:22:02.799164210	2022-01-01 19:04:00	לא ידוע	2025 אברהם	כבלים	הנעה	טיוטה קורולה	גורליץ 10 נתניה, הרב ברוך...הלברשטם מגורליץ, נתנ	2022-01-01 19:04:00
25.291697	הוחזר	2022-01-01 22:22:17.501820218	2022-01-01 21:57:00	לא ידוע	3338 חורי	פנצ'ר	פנצ'ר	מאדדה 3	ביטוח לאומי, אזור התעשייה, המחצבות 3, נצרת, ישראל	2022-01-01 20:37:00
24.568448	הוחזר	2022-01-01 21:32:34.106910039	2022-01-01 21:08:00	לא ידוע	1389 אריה	דלת טרוקה	דלת	סיאט לאון	נחום סוקולוב 35, תל אביב-יפו, ישראל	2022-01-01 20:50:00
18.046653	הוחזר	2022-01-01 22:46:02.799164210	2022-01-01 22:28:00	לא ידוע	7653 יהודה	כבלים חזקים	הנעה	פורד גלאקסי	הרב פניצל 10, ירושלים	2022-01-01 21:03:00
25.291697	הוחזר	2022-01-01 22:39:17.501820218	2022-01-01 22:14:00	לא ידוע	1057 שיפה	פנצ'ר	פנצ'ר	יונדאי i10	8G3QV3MQ+82	2022-01-01 22:14:00

המחקר על הנתונים אפשר לנו להבין מעט יותר טוב על אופי המתנדבים, אופן הטיפול בתקלות ויצוא הנתונים למאגר של הארגון.