**דירוג טלפונים חכמים**

**באמצעות אלגוריתמים חכמים**

****

**פרויקט סיום קורס**

**אלגוריתמים היוריסטיים ומקורבים ויישומים**

סמסטר ב', תשע"ט

**מגישים:**

לוי מלינדה

בארשבסקי צחי

לוקשין קוסטיה

**תוכן העניינים**

רקע 2

חלק א' – הפרמטרים הנבדקים3-4

מאפייני המכשיר3

הצרכים שלנו4

חלק ב' – אלגוריתמי הבדיקה5-10

שיטת BORDA5

אלגוריתם TOPSIS7

שיטת שולצה9-10

בחירת האלגוריתם המתאים10

חלק ג' – יישום11

**רקע**

שוק הסמארטפונים כיום מספק לנו מגוון עצום של מכשירים מתקדמים שבאמצעותם נוכל לנהל את חיינו. הקדמה הטכנולוגית והחידושים הדיגיטליים מבטיחים שהסמארטפון שנרצה יהיה מהיר, איכותי ונוח.

מבדיקה כללית שביצענו במסגרת הפרויקט על מנת להבין את הבעיה המשתקפת מולנו לעומקה, מצאנו כי מרבית האנשים מכלל קבוצות האוכלוסייה מעדיפים לקנות מכשיר טלפון על סמך המלצות של קרוביהם, ואלו שבכל זאת מגיעים "מוכנים" לחנות הסלולר, לרוב מתמקדים בבדיקה שלכאורה נראית להם מקיפה, אך לא די בכדי להשיג, בסופו של דבר, את המכשיר האופטימלי עבורם.

במרבית המקרים, חיפוש הסמארטפון האידאלי מסתכם בבדיקת מספר יכולות מרכזיות, אך מאוד מצומצמות, של המכשיר:

* מחיר - ככל הנראה הפרמטר הכי חשוב למרבית האנשים. ניתן למצוא לא מעט מכשירים מצוינים במחיר שווה לכל כיס, כאלה שמספקים תמורה טובה מאוד מבחינת נוחות, אבזור, טכנולוגיה ועוצמה.
* איכות השמע – היכולת של המכשיר לנהל שיחה בנוחות, בעוצמה מספקת ונטולה קטיעות באמצעות הספיקר הפנימי, הרמקולים של הסמארטפון והספיקר החיצוני.
* מסך - הדגש המרכזי הוא בעיקר על הרזולוציה, טווח הצבעים, נוחות הקריאה וחוויה בזמן משחק.
* נוחות השימוש – עד כמה נוח וידידותי הסמארטפון למשתמש, החל ממידות המכשיר ועד לתפעול מערכת ההפעלה, ממהירות חיישן טביעת האצבע ועד לאיכות החומרים וההרכבה.
* תוכנה וביצועים – פרמטר המתייחס ליעילות המעבד ומהירות מערכת ההפעלה, ובפשטות: כולנו מעוניינים במכשיר שלא יגמגם ברגע האמת, לא יתבע מסכי טעינה רבים וארוכים מדי ולא יתסכל אותנו בתפעול השוטף כתוצאה מ"שיהוקי מערכת".
* איכות המצלמה – נמדדת על סמך איכות הצילומים באור יום, תחת תאורה מלאכותית וצילומי מאקרו בשעות הערב. כמו כן, איכות הווידאו נמדדת ברמת החדות והפירוט, וכמות הרעידות שתועדו בכל סרטון.
* סוללה – אחת התכונות החשובות ביותר לרוב משתמשי הסמארטפון. נקדים ונספר שנורא קשה למצוא סמארטפון שיידע לעבוד יותר מיום וחצי ולענות בצורה טובה על כל הקריטריונים שהוצגו קודם-לכן.

כידוע, Life time של סמארטפון נע סביב שלוש שנים. כל קונה במעמד הקנייה מוכן להשקיע סכום לא מבוטל של כסף, בהנחה שההשקעה "תסגור לו את הפינה" לפחות לשנתיים וחצי הקרובות, ושהוא יוכל להפיק ממנה את התועלת הרצויה.

**התוצאה בפועל:** רבים מכלל האוכלוסיה מחליפים את מכשיר הטלפון שברשותם לאחר שימוש של כשנה וחצי עד שנתיים בעקבות חוסר שביעות רצון.

**חלק א' – הפרמטרים הנבדקים**

על מנת לקבל את התמורה המיטבית להשקעה, החלטנו להקל על רוכשי הסלולר ולהציע דרך למציאת מכשיר הסמארטפון הנכון ביותר עבורם. הפתרון המוצע מתמקד להתמקד בשני פרמטרים מרכזיים: **מאפייני המכשיר והצרכים שלנו**.

**מאפייני המכשיר**

מאפייני המכשיר בהם בחרנו להתמקד במסגרת הפרויקט כוללים את הקריטריונים שהוצגו קודם-לכן וכן קריטריונים נוספים:

****

**הצרכים שלנו**

המילה "שלנו" מקבלת משמעות שונה כאשר מתבוננים בקבוצות אוכלוסייה שונות. במסגרת הפרויקט בחרנו להתמקד ב-3 קבוצות אוכלוסייה ולהבין את הצרכים שלהן לעומקם:

|  |  |  |
| --- | --- | --- |
| **קבוצת אוכלוסייה** | **עיקר הפעילות** | **הדרישה מהמכשיר** |
| **ילדים** | * הפעלת סרטוני יוטיוב ברצף * שיטוט באתרי תוכן כבדים * הפעלת משחקים כבדים | * חיי סוללה ארוכים * גודל מסך מירבי * טעינת סרטונים ומשחקים כבדים במהירות * יכולות וידאו גבוהות * תפעול אינטואיטיבי וקליל בעבור הילד הממוצע במינימום תסכול ובמקסימום הנאה |
| **אנשי הייטק** | * העברה שוטפת של מידע * הודעות טקסט * שמירה על בטיחות המידע * שיחות וידאו | * חיי סוללה ארוכים * מעבד חזק * ביצועי מערכת חזקים * נפח זיכרון גדול * מצלמות איכותיות (ראשית ומשנית) |
| **פנסיונרים** | * שיחות טלפון * הודעות טקסט * צילום תמונות | * עוצמת סאונד חזקה * איכות צליל * ביצועי מערכת חזקים * תפעול פשוט, מהיר וחלק * איכות אור התצוגה הכחול |

**חלק ב' – אלגוריתמי הבדיקה**

במהלך התמודדות עם הבעיה, בחנו מספר אלגוריתמים ושיטות שיכולים לסייע בפתרונה.

בחלק זה, נציג את האלגוריתמים והשיטות שבאמצעותם יהיה ניתן להציג פתרון יעיל לבעיה, בהתאם לפרמטרים השונים שהצגנו בחלק הקודם.

האלגוריתמים והשיטות הם:

* שיטת BORDA.
* אלגוריתם TOPSIS.
* שיטת שולצה.

כמו כן, נסקור את האלגוריתמים והשיטות הנ"ל על ידי הצגת סקירה כללית, אופן הפעולה, כיצד הם מסייעים בפתרון הבעיה וכן יתרונות וחסרונות.

**שיטת BORDA**

סקירה כללית

* שיטת BORDA היא שיטה לדירוג פרויקטים ומערכות, המבוצע על ידי הקצאת נקודות אשר מתקבלות על פי דעה של אנשי מקצוע. BORDA היא משפחה של שיטות בחירה של מנצח יחיד, שבהן הבוחרים מדרגים אופציות או מועמדים לפי סדר עדיפויות.
* שיטת BORDA קובעת את הזוכה בבחירות על ידי מתן פתק לכל מועמד, מספר נקודות המתאימות למספר המועמדים המדורגים נמוך יותר. לאחר שכל הקולות נספרים, האופציה או המועמד שקיבל את מרבית הנקודות – נבחר כמנצח.
* שיטה זו נועדה לבחור באופציות או במועמדים מקובלים, ולאו דווקא באלה העדיפים על פי הרוב, ולכן היא מתוארת לעיתים קרובות כמערכת הצבעה המבוססת על קונצנזוס ולא על רודנות.

אופן פעולה

* **נקודות BORDA:**
* נתונים פרויקטים וקיימים מומחים
* כל מומחה מדרג את הפרויקטים ונותן נקודות לכל פרויקט
* הפרויקט הטוב ביותר מקבל נקודות
* הפרויקט הגרוע ביותר מקבל 0 נקודות
* הפרויקט הזוכה הוא הפרויקט שמקבל את כמות הנקודות המקסימלית.
* **ספירת BORDA:**
* ספירת BORDA היא ספירה של סכום הנקודות שניתנות לכל פרויקט.
* החישוב מתבצע על יד הנוסחה הבאה:  
  , כאשר:

– מספר המומחים

– מספר הפרויקטים

– נקודות BORDA הניתנות לפרויקט שניתנו על ידי מומחה

הערות

* דירוג BORDA לפרויקט נתון הוא מספר הפרויקטים האחרים שהם קריטיים יותר.
* דירוג BORDA לפרויקט מראה על מספר הפרויקטים שהם טובים יותר מפרויקט זה.

שימוש באלגוריתם

על מנת לתת מענה לבעיה שהצגנו, נשתמש בשיטת BORDA באופן הבא:

* כל אחד מאיתנו ידרג את כל הסמארטפונים במסד הנתונים ויעניק לכל אחד נקודות לפי ראות עיניו בהתאם ל-28 הקריטריונים שמתארים כל סמארטפון.
* לאחר הדירוג נוכל לבצע ספירת BORDA לפי הנוסחה הנ"ל.
* הסמארטפון הנבחר הוא זה שיקבל את כמות הנקודות המקסימלית לאחר ביצוע ספירת BORDA.
* פעולה זו תבטיח את מציאת הסמארטפון הטוב ביותר בהינתן הקריטריונים שבחרנו ובהינתן הדירוג שביצענו.

יתרונות

* שיטה פשוטה להשוואה בין מערכות רבות ובין קריטריונים רבים.
* החישובים בשיטה פשוטים לביצוע.
* הבוחרים מסוגלים להביע מערכת העדפות מפורטת.

חסרונות

* אין התחשבות ביעילות ובאמינות (דיוק) החישוב.
* המערכת דורשת לפחות רמה מסוימת של נומריות כדי לעבוד. ייתכן שלבוחרים יהיה קשה להבין.
* רמת המידתיות ומספר הקולות המבוזבזים תלויים במידה רבה בגודל המחוזות, כלומר בכמות הבוחרים.

**אלגוריתם TOPSIS**

סקירה כללית

* אלגוריתם TOPSIS הינו שיטה של צירוף פיצוי אשר משווה סדרה של חלופות על ידי זיהוי משקל עבור כל קריטריון, נרמול התוצאות/הציונים עבור כל קריטריון וחישוב המרחק הגיאומטרי (אוקלידי) בין כל חלופה לבין האלטרנטיבה האידיאלית, שהיא התוצאה הטובה ביותר עבור כל קריטריון.
* ההנחה של TOPSIS היא שהקריטריונים עולים או יורדים באופן מונוטוני. הנורמליזציה נדרשת בדרך כלל כאשר הפרמטרים או הקריטריונים לעיתים קרובות במידות אשר לא מתאימות לבעיות של קריטריונים מרובים.
* שיטות פיצוי כמו TOPSIS מאפשרות תמורה בין קריטריונים, כאשר תוצאה גרועה בקריטריון אחד יכולה להישלל על ידי תוצאה טובה בקריטריון אחר.
* אופן פעולה זה מספק צורה מציאותית יותר של מודלים, לעומת שיטות שאינן מפצות, אשר כוללים / לא כוללים פתרונות חלופיים המבוססים על חתכים קשים.

אופן פעולה

1. יצירת מטריצה הערכות המכילה מ- אלטרנטיבות ו-קריטריונים,  
   עם חיתוך של כל אלטרנטיבה עם קריטריון המוגדר כ-*.*

*לפיכך, תתקבל מטריצה* .

1. נרמול המטריצה וקבלת מטריצת ההחלטה בעזרת הנוסחה .
2. חישוב את מטריצת ההחלטה המשוקללת ע"י הכפלה של כל תא במטריצה במשקל של כל קריטריון.
3. דירוג הערך הטוב ביותר והערך הגרוע ביותר עבור כל עמודה:

* - הערך **המקסימלי** בכל עמודה
* - הערך **המינימלי** בכל עמודה

1. חישוב המרחק הגיאומטרי (אוקלידי) בין כל קריטריון לבין כל חלופה על ידי הנוסחאות ; .
2. חישוב הדימיון למצב הגרוע ביותר על ידי הנוסחה .
3. דירוג סופי לפי הערך שהתקבל ב- מהגבוה לנמוך.

שימוש באלגוריתם

על מנת לתת מענה לבעיה שהצגנו, נשתמש בשיטת TOPSIS באופן הבא:

* ניצור מטריצת הערכות עבור כל הסמארטפונים במסד הנתונים לפי 28 הקריטריונים המתארים כל אחד.
* ננרמל את המטריצה ונקבל מטריצת החלטה, אותה נשקלל ונקבל מטריצת החלטה משוקללת.
* נמצא את הערך המינימלי ואת הערך המקסימלי בכל עמודה, ונחשב את המרחק האוקלידי בין כל קריטריון לבין כל סמארטפון וכן את הדימיון למצב הגרוע ביותר.
* לבסוף, נדרג באופן סופי את הערכים מהגבוה לנמוך ונבחר בערך הגבוה ביותר.
* פעולה זו תבטיח את מציאת הטלפון הכי טוב בהינתן הקריטריונים שבחרנו.

תרשים מסכם:

יתרונות

* פשוט, רציונלי ובעל קונספט קל להבנה.
* אינטואיטיבי ובעל לוגיקה ברורה שמייצגת את הרציונל מאחורי הבחירה האנושית.
* קל לחישוב ויעיל.

חסרונות

* כאמור, השיטה מחשבת מרחק על פי הנוסחה של מרחק אוקלידי. משמעות הדבר היא שלא לוקחים בחשבון את הקורלציה בין הקריטריונים.
* באופן יחסי, חישוב המשקל קשה.
* קשה לשמור על עקביות בשיפוט.

**שיטת שולצה**

סקירה כללית

* שיטת שולצה היא שיטת הצבעה אשר משקללת עמדות של מצביעים לגבי כמה מועמדים לכדי סדר עדיפויות משותף.
* בבסיס השיטה מונחת השוואה בין כל שני מועמדים (לצורך העניין נקרא להם מועמד א' ומועמד ב') ובחינה של מספר המעדיפים את מתמודד א' על פניו של מתמודד ב', ולהיפך. שיטה זו מקיימת את תנאי קונדורוסה, לפיו אם יש מועמד המנצח כל מועמד אחר בהתמודדות ראש בראש מולו, אז הוא המועמד הזוכה.
* אם לגבי מועמד א' לכל מועמד אחר ב' רוב הקבוצה מעדיפה את א' על ב', אז א' צריך לזכות בבחירות. אם אכן מכתירים את המועמד בעל תוצאה עקבית שכזאת, שיטת שולצה מאמצת את התוצאה הזאת ללא שום שינוי.
* עם זאת, התחכום בשיטה מתגלה כשהשוואת רוב פשוטה מביאה לתוצאות מעגליות, כגון העדפה של אפשרות א' על ב', ב' על ג' ו-ג' על א'. במקרים כאלה, הידועים כפרדוקס קונדורסה, שיטת שולץ פותרת את הקונפליקט באמצעות השוואת עצומת ההעדפה לאורך מסלולים שונים בגרף המועמדים.
* כמו בשיטות אחרות המשקללות סדרי עדיפויות, גם בשיטת שולצה רשאי כל מצביע לדרג את המועמדים, לפי סדר העדפותיו. המצביע אינו חייב לדרג את כל המועמדים. כל שנדרש ממנו הוא הצבעה עקבית (בלשון מתמטית, קביעת יחס סדר חלקי בין המועמדים).

אופן פעולה

1. מסמנים ב- את מספר הבוחרים שהעדיפו את מועמד על פני מועמד .
2. מסמנים
3. בונים גרף מכוון משוקלל, שקודקודיו הם המועמדים ועל הקשת ממועמד למועמד כתוב את הערך .
4. העוצמה של מסלול בגרף היא הערך הקטן ביותר המופיע על הקשתות במסלול. אם מסמנים ב-את העוצמה הגדולה ביותר על המסלולים מ-A ל-B, נקבל שהיחס הוא הוא יחס סדר חזק. בנוסף, מכיוון שמספר האפשרויות הוא סופי, יש בו איבר מקסימלי. איבר (מועמד) כזה הוא הזוכה בהצבעה לפי שיטת שולצה.

הערה: כדי למצוא את המסלולים הכבדים ביותר, ניתן להשתמש באלגוריתם פלויד-ווארשל למציאת המסלולים הקצרים ביותר בין כל שני זוגות צמתים.

שימוש בשיטה

על מנת לתת מענה לבעיה שהצגנו, נשתמש בשיטת שולצה באופן הבא:

* נדרג את הסמארטפונים על מנת לסווג אותם מהגרוע אל הטוב ביותר.
* נבנה גרף מכוון משוקלל בהתאם לדירוגים הנ"ל.
* נפעיל את אלגוריתם פלויד-ווארשל על הגרף שהתקבל כדי למצוא את המסלול הכבד ביותר.

**בחירת האלגוריתם המתאים**

על מנת לבחור את האלגוריתם המתאים, נבחן מספר היבטים:

* סיבוכיות זמן ריצה.
* יעילות האלגוריתם בהתאם לבעיה.

סיבוכיות זמן ריצה

מבחינת סיבוכיות זמן ריצה, שיטת BORDA נותנת פתרון בצורה מהירה יותר כמעט פי שניים יותר מאלגוריתם TOPSIS.

יעילות האלגוריתם

במקרה זה, בהתאם לבעיה הספציפית בה אנחנו דנים, שיטת BORDA תיתן תוצאה פחות טובה מאלגוריתם TOPSIS, מהסיבה שבשיטה זו אין משקל לקריטריונים, כלומר הם שווי משקל. למשל, לבחירת ראש ממשלה, שיטת BORDA תתאים יותר מכיוון שכל קול של אזרח שווה ואין חשיבות למשקל.

לכן האלגוריתם שנבחר הוא...

בהתאם לצרכים שלנו, נבחר להשתמש באלגוריתם TOPSIS, שכן אנחנו מחפשים פתרון יעיל ומהיר.

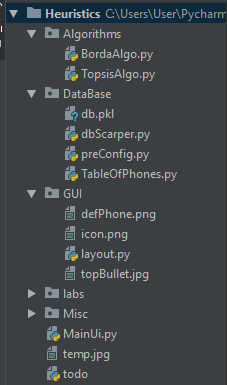
**חלק ג' – מימוש**

סקירה כללית

* המימוש בוצע בשפת Python 3.7 בסביבת העבודה PyCharm 2019.
* לטובת הממשק הגרפי, עשינו שימוש בספריות wxPython.
* הפרויקט מחולק לשלושה חלקים, והם:

1. בסיס הנתונים.
2. אלגוריתמים לדירוג.
3. ממשק משתמש.

מבנה של כל החלקים בפרויקט.



1. בסיס הנתונים:
   1. **dbScraper** - סקריפט אשר ניגש לאתר gsmarena.com ומבצע איסוף שיטתי של הספסיפיקציות של כל הטלפונים אשר קיימת באתר, בחלוקה ליצרנים ודגמים.
   2. **TableOfPhones** - מחלקה אשר מקבלת את בסיס הנתונים וההגדרות של הקריטריונים לפי קבוצת היעד ובונה טבלה, בה כל טלפון מקבל ציון לכל קריטריון שלו על פי הכללים שהוגדרו לקבוצת היעד הרלוונטית.
   3. **preConfig** - מילון אשר מכיל את ההגדרות ברירת מחדל של קבוצות היעד שאיתן עבדנו (ילדים, עובדי הייטק, פנסיונרים).
   4. **db** - בסיס הנתונים עשר נוצר ע"י הרצת הסקריפט dbScraper איתו אנו עובדים באפליקציה.
2. אלגוריתמים לדירוג:
   1. **BordaAlgo** - מחלקה אשר מקבלת טבלה, בה כל טלפון מקבל ציון לכל קריטריון שלו על פי הכללים שהוגדרו לקבוצת היעד הרלוונטית ומפעילה עליה את שיטת BORDA. המחלקה מחזירה מערך של טלפונים כאשר לכל טלפון ציון BORDA.
   2. **TopsisAlgo** - מחלקה אשר מקבלת טבלה, בה כל טלפון מקבל ציון לכל קריטריון שלו על פי הכללים ומערך משקלים לכל קריטריון שהוגדרו לקבוצת היעד הרלוונטית ומפעילה את אלגוריתם TOPSIS. המחלקה מחזירה מערך של טלפונים כאשר לכל טלפון ציון על פי האלגוריתם.
3. ממשק משתמש:
   1. **layout** - מחלקה אשר מגדירה ויזואלית את המסך הראשי של האפליקציה.
   2. **MainUi** - הקוד הראשי של האפליקציה, מוגדרות בו EventHanler לממשק משתמש וההפעלה הראשית של האפליקציה.

עקרון הפעולה של האפליקציה

1. כאשר האפליקציה מופעלת לראשונה נטען בסיס הנתונים אשר הוכן מראש (יפורט בהמשך),
2. על המשתמש לבחור את האלגוריתם אותו ירצה לממש, אם יבחר בTOPSIS תתווסף עמודה של משקלים למסך.
3. המשתמש יבחר את אוכלוסיית היעד אליה הוא רוצה להתאים טלפון, בהתאם לאוכלוסייה יטענו ההגדרות והכללים למתן ציון לכל קריטריון.
4. ניתן להגדיר את הכללים למתן ציון לכל קריטריון בצורה ידנית ומותאמת אישית.
5. כעת ניתן להריץ את האלגוריתם.
6. בסיום הריצה יופיע במסך התוצאה הזמן שלקח לאלגוריתם לרוץ (זמן זה לא כולל את זמן המידע מהבסיס נתונים והזמן להצגת התוצאה).
7. התוצאה היא חמשת הטלפונים שדורגו במקומות הראשונים, בתוספת תמונה להמחשה של כל טלפון וקישור לעמוד של כל טלפון באתר gsmarena.com.