

הקדמה

בפרויקט זה אנו מתעסקים בנושא דחיסת נתונים ללא הפסד, כאשר המטרה המרכזית היא להפחית את גודל המידע ללא אובדן כלל.

חקרנו כיצד הדחיסה עובדת תוך שימוש בשני האלגוריתמים הנפוצים: למפל-זיו והאפמן קוד וראינו את ההבדלים בין שתי השיטות.

האלגוריתמים המרכזיים בפרויקט

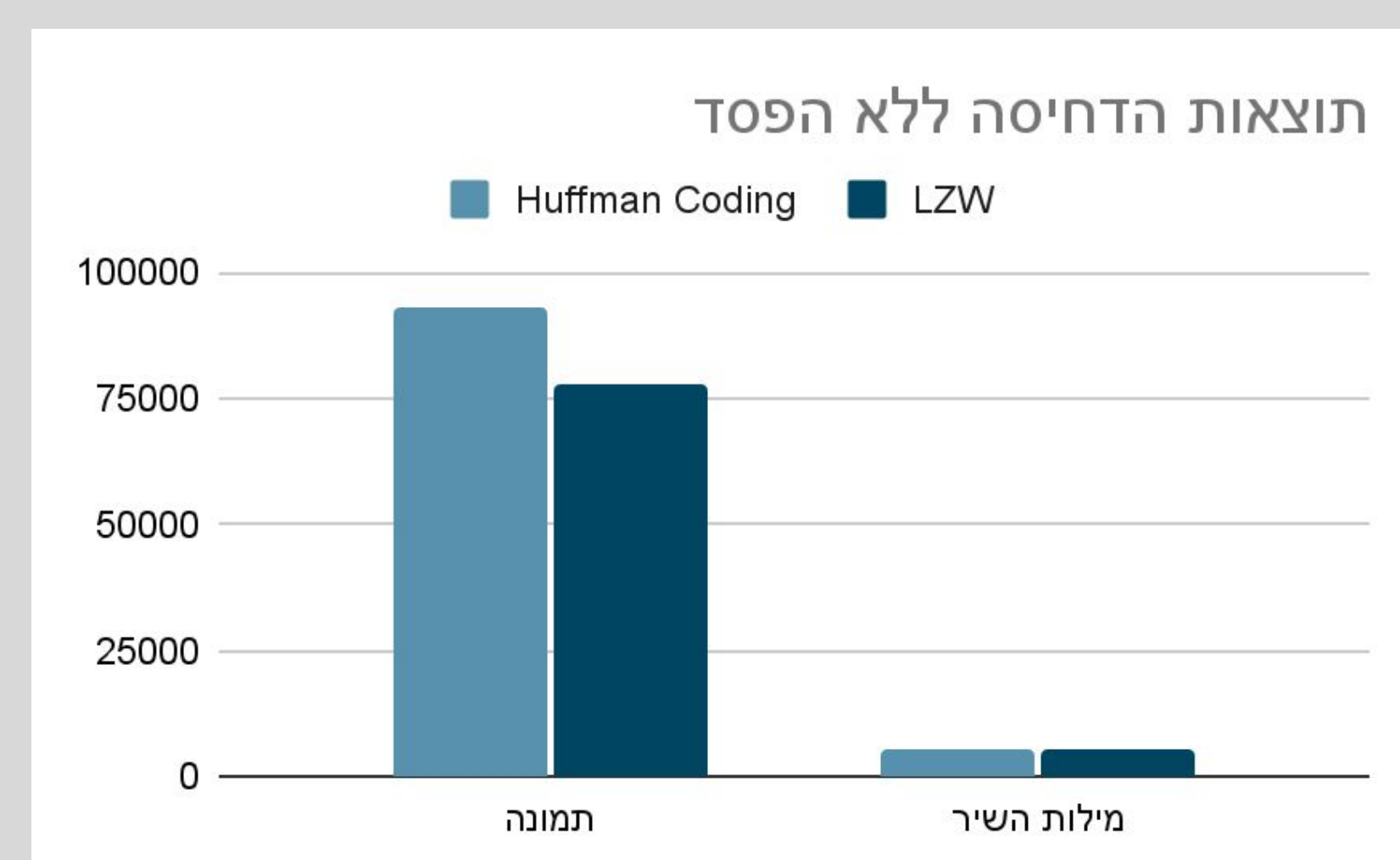
האפמן קוד:

שיטה לקידוד סימנים, כגון תווי טקסט, ללא אובדן נתונים. השיטה מבוססת על הקצאת אורך משתנה סימנים על פי שכיחותם, כך שסימן נפוץ יוצג באמצעות מספר קטן של סיביות. לרוב ניתן לחסוך באמצעות שיטה זו בין 20% ל-90% משטח האחסון.

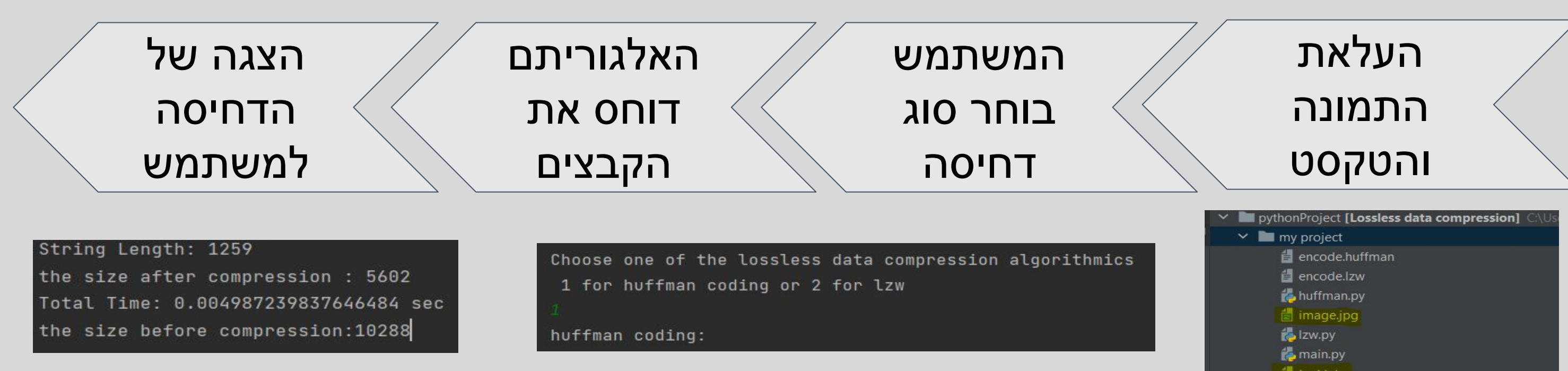
למפל-זיו:

אלגוריתם למפל-זיו הוא אלגוריתם לדחיסת נתונים מסוג דחיסה משמרת מידע, המאפשרת שחזור המידע הדחוס במלואו (ללא עיוות). האלגוריתם מבוסס על חלוקת המחרוזת המקודדת לתת-מחרוזות הנקראות פסקאות בתהליך המכונה פיסוק. כל פסקה מותאמת למחרוזת מעל א"ב סופי ונבנה מילון בתהליך דינמי. האלגוריתם הוא אוניברסלי, הדחיסה היא אסימפטוטית אופטימלית ולא נדרש ידע קודם של התוכן הנדחס.

על פי תוצאות המחקר נראה כי דחיסת LZW יותר טובה מהאפמן קוד (ללא התייחסות לזמן הדחיסה).



בפרויקט המשתמש יכול לבחור בין שתי שיטות הדחיסה LZW או האפמן קוד, לאחר שהמשתמש בוחר את סוג הדחיסה, האלגוריתם הנבחר מעבד את שני הקבצים (האחד תמונה והשני קובץ טקסט) ומראה לנו כיצד השיטה דוחסת את הקבצים, ובזכות זה אנו יכולים לראות את ההבדלים ביניהם.



מסקנות ודיונים

- בפרויקט אנו מצליחים להראות את היכולת של שני סוגי הדחיסות וכך ניתן להבין באיזו שיטה עדיף להשתמש, כאשר אנו מעוניינים לדחוס תמונה או קובץ טקסט עם מילות שיר. בעזרת הפרויקט הצלחנו להסיק שעדיף להשתמש ב-LZW בדחיסת התמונה וקובץ הטקסט, אך ככל שיש פחות תווים אנחנו רואים עדיפות בשימוש האפמן קוד לעומת LZW בזכות השיפור ביעילות הדחיסה.
- העלנו לדיון את השיפורים הבאים;
- פתיחת תיקייה אוטומטית בקוד אשר אליה יעלה המשתמש את הקבצים שהוא רוצה לדחוס. ביצוע שיפור זה הוא די פשוט, הוספה לקוד את הפונקציה של פתיחת תיקייה ולהמתין עד להעלאת הקובץ לתיקייה, לאחר שהמשתמש הכניס את הקובץ ממשיכים בקוד לדחיסה הקובץ.
- שיפור שני שאפשר לעשות בפרויקט הוא פתיחת תיקייה עם הקבצים הדחוסים, ביצוע שיפור זה אפשרי על ידי שהקוד ישמור כל דחיסה שהוא מבצע בתיקייה אשר תיפתח למשתמש בסוף התהליך.
- שיפור נוסף, בחירה אוטומטית איזה אלגוריתם הכי עדיף לו בלי שייאלץ לבחור לבד, ניתן לבצע את זה ע"י בדיקה של דחיסת כל סוגי קבצים הקיימים (כמו pdf/mp3..) בגדלים שונים תוך שימוש בשני האלגוריתמים, הגעה למסקנות והוספה בקוד את התנאים שאם המשתמש מעלה קובץ מסויים - סיומת וגודל אנו נגדיר עבורו את האלגוריתם היעיל ביותר.



סרטון הפרויקט: