

מיקומים גאוגרפיים בספרים וסיפורים – עדן עבדי ועמית טורנר

בעיה ופיתרון:



כשחשבנו על נושא לפרויקט עלתה לנו הבעיה הבאה: אם אני רוצה לדעת על ספר היכן הסיפור מתקיים בעולם, האם יש כלי שיכול לעשות זאת? אנחנו רצינו לראות על מפה היכן הסיפורים האהובים עלינו קורים גיאוגרפית! לכן החלטנו ליצור כלי שלוקח ספרים, מחלץ מהטקסט שלהם את המקומות הגיאוגרפיים ומציג את זה בצורה נוחה ונעימה לעין על מפה. נושא זה נכנס לשדה מדעי הרוח הדיגיטליים בכך שהוא לוקח כמות מידע גדולה שקשורה ישירות למקצועות הומניים מסורתיים (ספרות, היסטוריה, מדע, שפה וכו..), מעבד את מידע באמצעים טכנולוגיים של

שיטות מחקר מתחום המחשבים (במקרה שלנו עיבוד שפה, ניתוח טקסט למיקום גאוגרפי וחילוץ מידע בעזרת מילות מפתח) ולאחר מכן מציגים את המידע לאחר עיבוד בצורה גרפית שיכולה לתת רבדים חדשים למידע ולמחקר, ובנוסף בחרנו לתת דגש על הצגה גרפית נעימה לעין ומושכת שגורמת למשתמש לרצות "לשחק" עם הכלי ובכך להגיע למסקנות בעצמו. בעזרת המפה שיצרנו והפילטרים ניתן להגיע למסקנות מעניינות שלא היה ניתן להגיע אליהם עד היום - כמו למשל על איזה מקומות כתבו מתחילת הספרות הכתובה והאם בתקופות מסוימות בהיסטוריה היו מקומות יותר "פופולריים" שכתבו עליהם, ולאיזה ז'אנרים הם שייכים.

פרויקטים דומים:

לאחר שיטוט באינטרנט מצאנו כמה בלוגים ישנים שניסו לעשות זאת וכבר לא פעילים (למשל <http://www.placingliterature.com>), ועוד שירות שנותן אפשרות לחיפוש ספרים לפי מיקום אך מתייחס למיקום כללי של ספר, כלומר נותן את שם העיר או המדינה שעיקר הסיפור מתקיים בו, ולא לוקח אותך למסע סובב עולם אם הגיבור בסיפור מחליט באמצע לבקר במדינה אחרת. לאחר שראינו שפתרונות אחרים לא עושים את מה שאנחנו רוצים, ואם כן אז עושים זאת בצורה מאוד צולעת, התחלנו לעבוד.

מקורות מידע, אופן העבודה ועיבוד המידע בעזרת כלים (נכתב באנגלית כי משולב באתר ההצגה הסופית):

כמקור מידע לספרים עצמם, אנחנו משתמשים בProject Gutenberg שנותן גישה קלה לקבצי הטקסט המלאים של הספר. אנחנו רצים על הספרים בסדר עולה של המזהה שלהם, ומכל טקסט אנחנו שולפים את השם של הספר והכותב. בעזרת המידע הזה אנחנו משתמשים בAPI של OpenLibrary ומבצעים שאילתה של: `Book_name + Author_name` ומקבלים רשימת

תוצאות, כאשר הראשונה היא המידע ב-JSON של הספר שאנחנו מחפשים (לפעמים יש יותר מהוצאה אחת לספר, אך רוב המידע החשוב לנו זהה). עכשיו שיש לנו את קובץ ה-JSON המתאים ב-OpenLibrary אנחנו יכולים לחלץ את שנת ההוצאה, תמונת הספר, תמונת המחבר וקטגוריות הספר. בחרנו לחלץ מתוך הקטגוריות 10 קטגוריות מרכזיות הכי נפוצות לטובת פילטרים נוחים למשתמש הקצה. לאחר שוידאנו שקיים לספר את כל המידע שאנחנו צריכים לטובת המערכת, אנחנו מתחילים בעיבוד הטקסט לחיפוש מיקומים גאוגרפיים. בעזרת אלגוריתם **spacy NLP** אנחנו מחלצים את כל ה-**Named Entities** בספר, ולאחר מכן בעזרת **black list** אנחנו מדללים את הרשימה ושולחים לאלגוריתם **geopy** שלוקח מילה ומחזיר קורדינטות גאוגרפיות ומידת ביטחון בצורת מספר ($0 < x < 1$). לפי מידת הביטחון אנחנו מחליטים אם להשאיר את המיקום לספר. לאחר שיש לנו מאגר מידע של ספרים, המידע הרלוונטי שלהם ורשימת קורדינטות לכל ספר – אנחנו משתמשים בספריית ה-**MapBox JavaScript** וממירים את המידע לנקודות על מפה וירטואלית שהכנו ויושבת בענן. למפה כתבנו פונקציות שמשתמשות בפילטרים לפי הצרכים של הפרויקט ובנוסף פונקציות נוספות שמבצעות שליפה של התמונות הרלוונטיות של הכריכה והסופר בזמן אמת כאשר לוחצים על כל ספר במפה.