

מסמך דרישות לפרויקט בקורס כריית טקסטים

תקציר הפרויקט

מטרת הפרויקט היא ליישם תהליך כריית טקסטים לפתרון בעיה מוגדרת מראש ומבוססת מחקר. הפרויקט כולל שלבים של איסוף נתונים, עיבוד מקדים, מימוש אלגוריתמים מתקדמים, ניתוח ביצועים והצעת שיפורים. הפרויקט יכלול חמישה שלבים עיקריים, כאשר כל שלב תורם למימוש כולל ואיכותי של המערכת.

שלבי הפרויקט

שלב א': איסוף ועיבוד מקדים של הנתונים (Data Collection & Preprocessing)

מטרות:

1. איסוף נתונים:

- השגת מערך נתונים רלוונטי ממקורות אמינים (API, קבצי CSV, scraping מאתרים).

- וידוא רישוי נתונים ושמירה על פרטיות.

2. עיבוד מקדים (Preprocessing):

- ניקוי נתונים: הסרת תווים מיוחדים, טיפול בטקסטים כפולים, הסרת שורות ריקות.

- עיבוד טקסטים: הסרת stopwords, המרה לאותיות קטנות, סטמינג ולמנטיזציה.

- יצירת תכונות חדשות: זיהוי ישויות (Named Entity Recognition), ניתוח רגשות (Sentiment Analysis), או זיהוי נושאים (Topic Modeling).

3. ייצוג טקסטים (Feature Engineering):

- בחירת שיטת ייצוג: Bag of Words, TF-IDF, Word2Vec, FastText, BERT.

- ניתוח השפעת תכונות (Exploratory Data Analysis) להדגשת תבניות מעניינות בנתונים.

תוצרים נדרשים:

- מערך נתונים נקי ומוכן לעבוד.

- מחברת Jupyter עם ניתוחים ויזואליים (גרפים, מילים נפוצות, word clouds).

- פסקה קצרה המתארת את ה-Dataset: מה מקורו? מתי נאסף? מהן מגבלותיו? אילו קבוצות באוכלוסייה הוא מייצג?

שלב ב': מימוש אלגוריתמים מתקדמים לפתרון הבעיה (Modeling)

מטרות:

1. הגדרת בעיית למידה:

- סיווג טקסטים (Classification), ניתוח נושאים (Topic Modeling), זיהוי ישויות (NER), או ניבוי טקסט (Text Prediction).

2. בחירת אלגוריתמים:

- עבור בעיות סיווג: Logistic Regression, SVM, Naive Bayes, Decision Trees, או מודלים מתקדמים (LSTM, BERT, GPT).

- עבור בעיות ניתוח נושאים: Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA).

3. אימון מודלים:

- חלוקת הנתונים ל-Train/Validation/Test (ביחסים 70/15/15).

- שימוש בטכניקות לערבול נתונים (Cross-Validation).

- כונון היפר-פרמטרים (Hyperparameter Tuning) באמצעות Grid Search או Bayesian Optimization.

תוצרים נדרשים:

- קוד מאורגן התומך בהרצה מקצה לקצה משורת הפקודה.

- מחברת Jupyter המתארת את האלגוריתמים והבחירות שנעשו.

- שמירת המודלים המאומנים (Model Serialization) לשימוש עתידי.

שלב ג': אבולוציה וניתוח התוצאות (Evaluation & Analysis)

מטרות:

1. מדידת ביצועים:

- שימוש במדדים מתקדמים: Accuracy, Precision, Recall, F1-Score, Confusion Matrix, AUC-ROC.

- מדדים נוספים בהתאם לאופי הבעיה (למשל, Perplexity בניתוח נושאים, BLEU בניבוי טקסט).

2. ניתוח שגיאות (Error Analysis):

- בחינת מקרי כשל (False Positives/Negatives) וזיהוי דפוסים אפשריים לשיפור.

3. ויזואליזציה של תוצאות:

- שימוש בגרפים להמחשת ביצועי המודלים (Precision-Recall Curve, Confusion Matrix Heatmap).

- ניתוח תובנות מרכזיות מתוך התוצאות.

תוצרים נדרשים:

- מחברת Jupyter עם גרפים וניתוח תוצאות.

- ניתוח שגיאות (Error Analysis): להציג דוגמאות קונקרטיות של טעויות (False Positives/Negatives), ולנתח מדוע המודל טעה. האם הטעות נובעת מטקסט דו-משמעי? מחסור בנתונים מסוג מסוים? מבעיה בעיבוד המקדים? זה לב ליבו של שיפור המודל.

- סיכום מסקנות והמלצות לשיפור ביצועים.

שלב ד': הצעת אלגוריתם משופר וניתוח מחודש (Algorithm Enhancement & Re-evaluation)

מטרות:

1. שיפור המודל:

- שימוש בטכניקות מתקדמות כגון Ensemble Learning (Random Forest, Gradient Boosting), Transfer Learning (BERT, GPT), או Fine-Tuning למודלים קיימים.
- כוונת נוסף של היפר-פרמטרים על בסיס ניתוח שגיאות מהשלב הקודם.

2. אבולוציה מחודשת:

- בחינת הביצועים לאחר השיפורים בהשוואה לתוצאות המקוריות.
- הערכת השפעת השיפורים באמצעות מדדים זהים לשלבים הקודמים.

תוצרים נדרשים:

- קוד מעודכן עם האלגוריתם המשופר.
- השוואה בין תוצאות המודלים לפני ואחרי השיפורים (גרפים, טבלאות).
- שלב זה צריך להתבסס ישירות על המסקנות מניתוח השגיאות. לדוגמה:
- **טענה:** "המודל טועה במקרים של סרקזם. אולי הוספת תכונות רגש תעזור." -> ניסוי: הוספת Sentiment Score כתכונה ובדיקת השיפור.
- **טענה:** "נראה שהמודל מתקשה עם מילים נדירות." -> ניסוי: שימוש בטכניקות Data Augmentation כמו החלפת מילים נרדפות או תרגום-חזרה (Back-translation) כדי להעשיר את הנתונים.

שלב ה': כתיבת דו"ח מסכם והגשת הפרויקט (Final Report & Submission)

מטרות:

1. הכנת דו"ח מפורט:

- מבוא: רקע, הגדרת הבעיה, מטרות הפרויקט.
- סקירת ספרות: סיכום מחקרים ופתרונות קיימים בתחום.
- מתודולוגיה: פירוט שלבי העבודה, בחירת אלגוריתמים וייצוג הנתונים.
- תוצאות וניתוח: סיכום הביצועים, ניתוח שגיאות, ותובנות.
- סיכום והמלצות: תובנות מרכזיות, מגבלות הפרויקט והמלצות להמשך.

2. הגשת הפרויקט:

- העלאת הקוד, הדו"ח והמחברות למאגר ה-GitHub של המעבדה.
- וידוא שהקוד תומך בהרצה מקצה לקצה בעזרת שורת הפקודה.

דרישות נוספות:

- הדו"ח יוגש בפורמט PDF. מומלץ לעבוד עם Latex היות והוא הסטנדרט המקצועי המקובל במדעי המחשב בג'ורנלים וכנסים מתקדמים.
- שמירה על סטנדרטים לכתיבת קוד ופרויקט מתועד כראוי.

דרישות טכניות כלליות:

- שפת תכנות: Python.
- כלי פיתוח: Jupyter Notebook, Google Colab, PyCharm/VSCode.
- ספריות מומלצות: NLTK, SpaCy, Scikit-Learn, TensorFlow, PyTorch, Hugging Face Transformers, Pandas, Matplotlib, Seaborn.
- ניהול גרסאות: שימוש ב-Git לניהול גרסאות, כאשר הפרויקט כולו חייב להיות מאוחסן במאגר GitHub ייעודי של המעבדה.

לוח זמנים (אופציונלי)

חלוקת שלבי הפרויקט לפי לוח זמנים

שלב	תיאור	שבועות	זמן משוער להשלמה
שלב א'	איסוף ועיבוד מקדים של הנתונים	שבוע 1-4	4 שבועות
שלב ב'	מימוש אלגוריתמים מתקדמים	שבוע 5-8	4 שבועות
שלב ג'	אבלואציה וניתוח תוצאות	שבוע 9-10	2 שבועות
שלב ד'	הצעת אלגוריתם משופר וניתוח מחודש	שבוע 11-12	2 שבועות
שלב ה'	כתיבת דו"ח והגשת הפרויקט	שבוע 13	1 שבוע

קריטריונים להערכת הפרויקט

חובה לציין על שלבי הביצוע וחלקי הפרויקט באחריות מי זה נעשה על מנת שאוכל לתת ציון אישי

1. איכות ועומק העיבוד המקדים (20%).
2. בחירת אלגוריתמים והצדקות טכניות (25%).
3. מדדי ביצועים וניתוח שגיאות (20%).
4. חדשנות ושיפור ביצועים (20%).
5. כתיבת דו"ח ועמידה בדרישות טכניות (15%).

אתרים רלוונטיים -

- אתר huggingface - קהילת opensourcen העולמית של בינה מלאכותית. אפשר למצוא שם מגון [datasets](#) ובעיקר [מודלים](#) מכל הסוגים.
- אתר קאגל - אתר שמציע תחרויות שונות בתחומי data science. אפשר לבחור משם [datasets](#) או ממש להירשם ל**תחרות** ועל בסיסה לבצע את הפרויקט.
- [מאגר הNLP הלאומי](#) של ישראל שמכיל נתונים, מודלים וכל מה שקיים כדי לבצע כריית טקסטים בעברית וערבית.

תחרויות -

- [SemEval2026](#)
- [Clef2026](#)
- [Evalita2026](#)

הרשת מכילה הרבה מאוד מאגרים ותחרויות שלא מפורטים פה, מוזמנים לחפש.

בהצלחה בפרויקט!