הצעה למודל ML:

1. מספר הפיצ'רים יהיה כמספר המילונים.
2. עבור כל שורה ב-Data Set (שמייצגת הודעה שנכתבה בצ'אט) – ננקה את ההודעה לשלושה חלקים תחביריים: נושא, נשוא ומושא.
3. נשערך את ערכי הפיצ'רים ע"פ שיוך של כל חלק תחבירי למילון שמייצג את הפיצ'ר המתאים.  
   לדוגמה: עבור הפיצ'ר שמייצג את המילון עבור מילים עם הקשר מתמטי – אם הנושא או הנשוא או המושא שייכים למילון הנ"ל, נסמן ב-1 את הפיצ'ר. יש עוד לוגיקות שאפשר לממש בהקשר הזה.

המצב כרגע: יש שלושה מילונים. בהינתן משפט אנו מבצעים חלוקה תחבירית של המשפט. שולחים כל אחד מהחלקים (נושא, נשוא ומושא) לפונקציה שבודקת לאיזה רגע קריטי המילה נקשרת ע"י בדיקה באיזה מילון המילה קיימת.

בהינתן תיוג של כל המילים במשפט, נחליט על תיוג המשפט באופן הבא:

1. אם לפחות אחת מהמילים תויגה כ-DS, נחזיר DS.
2. אם אף אחת מהמילים לא תויגה כ-MAT וגם TC, נחזיר NMD.
3. נחזיר את המקסימום מבין MAT ו-TC.

נשאר לבצע:

1. לנסות שוב להוציא את מילת הבסיס (יחיד, עבר). הצלחנו לנקות מסימני פיסוק ותחיליות (ה' הידיעה, ו' החיבור וכד').
2. לחשוב איך לשלב למידת מכונה בנתונים הקיימים שהוצאנו.

בעיות / סוגיות:

1. לא לתייג את משפטי המורה. (שלא ישלחו לנו אותם)
2. לסדר את המילונים לפי מילים ולא לפי משפטים.
3. לשנות את ה-DS שיתאים עבור הודעה ספציפית במקום שימוש במילון.
4. לעשות Deploy לענן.

משפטים לניסוי:

טכני:

1. העכבר נתקע לי
2. תלחץ על הכפתור
3. איך לוקחים כאן שליטה

אופציונלי:

1. לסנכרן מבני נתונים של ה-IdlenessAlerter כדי להימנע ממצב בו הת'רד שבודק Idleness והת'רד שמטפל בבקשה שניהם ניגשים לאותו מבנה נתונים (לדוגמה: last\_alert\_in\_rooms).

מצב נוכחי:

1. האלגוריתם של ה-ML מוכן. כרגע בוחרים את התיוג לפי שיטת Majority Vote בין כל ה-Classifiers השונים. התוצאות בזיהוי הודעות של NMD ושל TEC הן לא מוצלחות במיוחד. צריך לתת Train data לגביהם.
2. לעשות בדיקות על כל הפרויקט ולראות שהכל עובד אחרי השינויים.
3. לבדוק אם DS עובד אחרי שמקבלים הודעה מהצד של יקיר ואחיעד.

באגים:

1. אין אנשים בחדר ועדיין מתקבלות התראות של Idleness.
2. מתקבלת התראת Idleness ואז שולחים הודעה ועדיין יש Idleness.