**0**

**AI-mong Us**

**פרויקט סיום - קורס בינה מלאכותית (67842)**

מתן גולדשטיין

בן צאיג

איתמר סבן

יאיר איילון

**Find The**

**AI-mpostor**

**תוכן העניינים**

שער.............................................................................................1

תוכן העניינים.................................................................................2

מטרת הפרויקט + הקדמה................................................................3

הצגת המשחק................................................................................4

תיאור הבעיה..................................................................................5

דרכי הפתרון...................................................................................8

דרכי מימוש....................................................................................13

תוצאות..........................................................................................14

מסקנות ודיון...................................................................................20

השלכות ויעדים לעתיד......................................................................21

סיכום חוויית העבודה על הפרויקט......................................................22

הרצת התוכנית................................................................................23

ביבליוגרפיה וחומרי עזר....................................................................25

נספחים..........................................................................................26

**מטרת הפרויקט + הקדמה**

* **מטרת הפרויקט הינה לגלות האם ניתן לפתור ביעילות משחקים בעלי אלמנטים פסיכולוגיים/סוציולוגיים באמצעות כלי AI.**
* **בפרויקט נתמקד בדרכי פתרון שלimperfect information games /incomplete information games** [**(1)**](https://web.stanford.edu/~jdlevin/Econ%20203/Bayesian.pdf) **(מפה והלאה נשלב היפר-קישורים לשם הרחבת המידע, והם יסומנו בקו תחתון), זהו מצב משחק שבו לא לכל השחקנים יש אותו ידע על המשחק, ולכן הם פועלים בתנאי אי וודאות, ולכן שיטות קלאסיות עבור משחקים כנגד יריב כמו alpha beta agent או nash equilibrium שלמדנו בקורס פחות רלוונטיות, מכיוון שהמידע שאנחנו מחזיקים עבור המשחק הוא שונה מהמידע שהיריב שלנו מחזיק עבור המשחק, ואלו בעיות שיותר קרובות לפתרון בעיות מהחיים האמיתיים.**
* **דוגמה מאוד מוכרת למשחק כזה הוא פוקר, שהייתה בעיה קשה לפתרון במשך הרבה שנים, ורק בשנים האחרונות הצליחו למצוא לה פתרון** [**(2)**](https://www.nature.com/articles/d41586-019-02156-9)**, שמנצח באחוזים גבוהים מאוד שחקנים מן הטופ העולמי, אך גם הוא אינו מושלם ולא מנצח ב-100%.**
* **המשחק שאותו נבחר לשחק יהיה מסוגSocial deduction game** [**(3)**](https://en.wikipedia.org/wiki/Social_deduction_game)**, זהו משחק שבו השחקנים מנסים לגלות את ה"תפקיד החבוי" של השחקנים האחרים, ולשם כך הם משתמשים בהסקה לוגית ובנתונים שצברו לאורך המשחק.**
* **מצאנו ניסיון למימוש סוג משחקים כזה של חוקרים מMIT-** [**(4)**](https://news.mit.edu/2019/deeprole-ai-beat-humans-role-games-1120)**, שפתרו את המשחק "** **“The Resistance: Avalon”, ונעזרנו בעקרונותיו כדי לפתור את המשחק שלנו.**
* **במהלך הפרויקט ננסה לפתור את המשחק Among Us ובאמצעותו נסיק את התובנות הכלליות שלנו לשאלה – האם ניתן לפתור משחקים בעלי אלמנטים פסיכולוגיים/סוציולוגיים באמצעות כלי AI?**
* **הבהרה: במהלך תיק העבודה השתמשנו בשמות השונים של crew/crewmate/חבר צוות – הם כולם שקולים. כנ"ל לגבי "מתחזה"/"שקרן"/"רוצח"/impostor שגם מייצגים אותו דבר. סה"כ יש ייצוג ל-2 התפקידים האפשריים במשחק.**

**הצגת המשחק**

**Among Us הוא Social deduction game מקוון בו יכולים לשחק במקביל בין 5 ל-12 שחקנים, המשמשים כחברי צוות (CREW), ומתוכם בין 1 ל-3 שחקנים מסומנים כ"רוצחים"/"מתחזים" (IMPOSTORS). שחקנים אלו נבחרים באופן אקראי בתחילת כל משחק. על חברי הצוות להשלים משימות ברחבי המפה בצורה של "משחקונים" פשוטים הנמשכים מס' שניות. המתחזים יודעים גם הם את רשימת המשימות, והם רוצים לזייף את ביצוען. אם שחקן נהרג או מודח מהמשחק, הוא אינו יכול לזוז או להצביע יותר. ל"רוצחים" בניגוד ל"חברי הצוות" יש**[**שדה ראייה**](https://he.wikipedia.org/wiki/%D7%A9%D7%93%D7%94_%D7%A8%D7%90%D7%99%D7%99%D7%94)**משופר שמאפשר להם לראות יותר טוב מחברי הצוות את מצב המשחק.**

**חברי הצוות יכולים לנצח אם הם מסיימים את כל המשימות לפני שנרצחו מספיק חברי צוות, או על ידי איתור והוצאת המתחזים מהמפה דרך הצבעות.**

**מטרתם של הרוצחים היא לרצוח את חברי הצוות. כשהרוצחים נמצאים בקרבה מסוימת לחבר צוות ("טווח רציחה") הם יכולים להרוג אותו.**

**כדי שהמתחזים יזכו, עליהם להרוג מספיק חברי צוות כך שמספר המתחזים ישתווה למספר חברי הצוות, או להגיע למצב בו מסתיימת הספירה לאחור של משימת החבלה.**

**אם שחקן מוצא גופה של אדם שנרצח, הוא יכול לדווח על כך, דבר שיוביל למפגש קבוצתי שבו כל שאר המשחקים נעצרים והשחקנים מנסים לזהות את המתחזים, בהתבסס על ראיות ועל ה**[**אליבי**](https://he.wikipedia.org/wiki/%D7%90%D7%9C%D7%99%D7%91%D7%99)**של כל שחקן, ולעיתים גם על ידי בחינת זירת הרצח. אם מגיעים להצבעה בה מתגבש רוב נגד אחד השחקנים, השחקן הנבחר מודח מהמפה ושאר השחקנים רשאים גם ליזום "פגישת חירום" על ידי לחיצה על**[**כפתור**](https://he.wikipedia.org/wiki/%D7%9B%D7%A4%D7%AA%D7%95%D7%A8)**בחדר שנמצא במקום מסוים במפת המשחק כל העת.**

**תיאור הבעיה**

* **הבעיה אותה אנו מנסים לפתור במהלך הפרויקט היא ניצחון במשחק Among Us, תוך כדי בחירת הפעולות הנכונות והסקת המסקנות הנכונות על אף החוסר הוודאות ולמרות שלא בכל משחק תמיד קיימת ה"החלטה הנכונה", כיוון שהרבה מידע לא תמיד ידוע לאותו שחקן מסוים והוא נע לרוב בחוסר וודאות.**

**קלט הבעיה הגדולה**

* **כמות crew ו-impostor בתחילת המשחק.**
* **מפת המשחק עם מיקומי החומות ומיקומי המשימות של המפה (תתקבל כקובץ JSON להרצת התוכנית).**

**-----**

* **במהלך המשחק נלמד כל צד – גם את ה-IMPOSTOR-ים ("המתחזים") וגם את ה-CREW-ים ("חברי הצוות"), לנצח את הצד השני.**
* **למעשה יש לנו בעיה גדולה של ניצחון במשחק, שאותה נפרק לתתי-בעיות כך שמטרת האלגוריתם שנבנה תהיה לבחור את הצעד הנכון בכל צעד שיידרש במהלך המשחק (בין אם זה ביצוע הצעד הפיזי, בחירת המהלך הנכון – לעשות משימות, לרצוח וכד'), ולבסוף לבצע את הסקת המסקנות המתאימה ולהצביע בהתאם.**
* **המטרה היא לבצע את ההחלטות הנכונות כל פעם, בראייה קדימה לבחירה במהלכים שיובילו אותנו לניצחון במשחק.**

**קלט תתי-הבעיות**

**הסוכן שלנו יבחר את המהלך הנכון בעזרת ה-observations (התצפיות והמידע שהוא צובר לאורך המשחק – דבר שיישמנו כחלק ממימוש המשחק), וישתמש באמצעי בינה מלאכותית ולמידה ממקרי עבר כדי לבחור את הצעד הנכון בהתאם לידע הנתון לו באותו רגע (הסבר על כיצד הוא יעשה זאת יפורט בהמשך תיק העבודה).**

**להלן הפרמטרים אשר יחזיק הסוכן ובעזרתם יקבל את ההחלטה:**

**עבור CREW:**

* **מיקום השחקן (קואורדינטה)**
* **צבע השחקן**
* **כמה זמן עבר מתחילת המשחק**
* **כמה משימות נשאר לבצע**
* **אילו משימות נשארו**
* **מס' הסיבוב הנוכחי**
* **מס' הסוכנים שאני רואה כרגע**
* **הזמן שנשאר עד שניתן "לקרוא" לשולחן (ניתן לבצע call meeting מתוך השולחן רק כעבור פרק זמן מסוים)**
* **מס' הקריאות לשולחן שנותר לי לבצע (מס' הקריאות ל-call meeting באמצעות קרבה לשולחן הוא מוגבל)**
* **כמה זמן עבר מתחילת הסיבוב**
* **עבור כל סוכן – המיקום האחרון שראיתי אותו (קואורדינטה)**
* **עבור כל סוכן – הזמן האחרון שראיתי אותו**
* **עבור כל סוכן – כמה משימות שונות ראיתי אותו עושה**
* **עבור כל סוכן – אילו משימות ראיתי אותו עושה**
* **עבור כל סוכן – האם ראיתי אותו רוצח**
* **עבור כל סוכן – האם הוא מת**
* **עבור כל סיבוב – מי מת בסיבוב הזה**
* **עבור כל סיבוב – מי הודח בסיבוב הזה (מי שקיבל הכי הרבה הצבעות)**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב**
* **עבור כל סיבוב – מי השתמש בשולחן באותו סיבוב**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב**
* **עבור כל סיבוב – מיקום הגופה שדווחה באותו סיבוב**
* **עבור כל סיבוב – מי רצח (לפי המדווח) באותו סיבוב**

סה"כ 22 פרמטרים.

**עבור IMPOSTOR:**

* **מה ה-kill\_cool\_down שלי (עוד כמה זמן אני יכול לרצוח שוב)**
* **המשימות שזייפתי ביצוע שלהן**
* **זמן שזייפתי ביצוע משימות**
* **מיקום השחקן (קואורדינטה)**
* **צבע השחקן**
* **כמה זמן עבר מתחילת המשחק**
* **כמה משימות נשאר לבצע**
* **אילו משימות נשארו**
* **מס' הסיבוב הנוכחי**
* **מס' הסוכנים שאני רואה כרגע**
* **הזמן שנשאר עד שניתן "לקרוא" לשולחן (ניתן לבצע call meeting מתוך השולחן רק כעבור פרק זמן מסוים)**
* **מס' הקריאות לשולחן שנותר לי לבצע (מס' הקריאות ל-call meeting באמצעות קרבה לשולחן הוא מוגבל)**
* **כמה זמן עבר מתחילת הסיבוב**
* **עבור כל סוכן – על ידי מי הוא נראה נרצח**
* **עבור כל סוכן – האם הוא impostor**
* **עבור כל סוכן – המיקום האחרון שראיתי אותו (קואורדינטה)**
* **עבור כל סוכן – הזמן האחרון שראיתי אותו**
* **עבור כל סוכן – כמה משימות שונות ראיתי אותו עושה**
* **עבור כל סוכן – אילו משימות ראיתי אותו עושה**
* **עבור כל סוכן – האם ראיתי אותו רוצח**
* **עבור כל סוכן – האם הוא מת**
* **עבור כל סיבוב – מי מת בסיבוב הזה**
* **עבור כל סיבוב – מי הודח בסיבוב הזה (מי שקיבל הכי הרבה הצבעות)**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב**
* **עבור כל סיבוב – מי השתמש בשולחן באותו סיבוב**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב**
* **עבור כל סיבוב – מיקום הגופה שדווחה באותו סיבוב**
* **עבור כל סיבוב – מי רצח (לפי המדווח) באותו סיבוב**

סה"כ 28 פרמטרים.

**בנספח א' (**[**היפר-קישור**](#כמותמצביםאפשרים)**) מפורטת הדרך לחישוב כמות הקומבינציות האפשריות, זהו מס' עצום של בערך 10 בחזקת 274, וזאת כמות מצבים אפשרית שלא ניתן לתת לבן אדם להתייחס לכל אחת מהן, ולצורך כך נצטרך להחזיק רשת שלמה שתוכל לתת מענה לכל הצבת פרמטרים אפשרית.**

**דרכי הפתרון**

**הקדמה**

* **Among Us הוא אחד מן המשחקים הכי פופולריים בשנה האחרונה, עם זאת לאחר חיפוש מעמיק לא מצאנו לו פתרון באמצעות AI שמתיימר לנסות לפתור את המשחק, ככל הנראה בשל המורכבות שבכך. לכן המשימה שלנו הייתה ראשונית ולא היה ברור אם בכלל היה ניתן לפתור אותה.**
* **מה שהיה ידוע לנו הוא בעיקר השראה מפרויקטים דומים שנעשו בנושא של social deduction games (הסבר בהקדמה). ידענו שעלינו להשתמש באלגוריתם למידה כלשהו שילמד את הסוכן תוך כדי שימוש במבנה מידע מוקדם, כך שלסוכן יהיה גורם התחלתי ללמוד ממנו.**
* **עם זאת לא ידענו מהי דרך הלמידה הנכונה מתוך דרכי הלמידה הרבות שלמדנו בקורס, כמו supervised learning,Q-learning ועוד. לכן השאלה עבורנו בפרויקט הייתה – האם בכלל ניתן לפתור את הבעיה, ואם כן אז כיצד.**
* **בחרנו להשתמש ב-probabilistic graphical model מכיוון ש-PGM חוזה על סמך חישוב של הסתברות מותנית (נוסחת "בייס"), ומפני ש-Among Us הוא משחק דדוקציה, אז הדרך הנכונה ביותר לכתיבת ערך לפעולה הוא על ידי חיזוי השלכותיה בעזרת חישוב ההסתברות המותנית בהינתן הידע הנוכחי של הסוכן. כל זאת תוך השראה ממחקרי עבר אחרים בנושאי דומים (שהרחבנו עליהם בהקדמה), שראינו שגם בהם הלוגיקה של המשחק נבנתה ע"י עץ הסתברויות. בחרנו לממש את ה-PGM באמצעות Bayesian network.**
* **זוהי גישה שברוב המקרים אמורה לתת חישוב מדויק ואמין יותר, עם זאת היא גישה שלא משתמשים בה לרוב לפתרון משחקים עם הרבה מידע, מה שהופך את השימוש בה, או ליתר דיוק היכולת להשתמש בה לפתרון, למעניין יותר.**
* **ישנם מס' חסרונות לרשת הבייסיאנית, בעיקר בשל העובדה שלא קיימות ספריות רבות למימושה, ובנוסף מדובר ברשת שמבצעת חישובים הרבה יותר מהרגיל, והיא כבדה יותר להרצה (הרחבה על רשתות בייסיאניות נמצאת בהמשך תיק העבודה כחלק מ"חומרי עזר מומלצים לקריאה וצפייה").**
* **בנוסף, על מנת לייעל את תהליך הלמידה של הסוכנים, וגם פה בעקבות השראה ממחקרי עבר בנושאים דומים (ובנוסף הדרכה מיוני המתרגל להשתמש ב-fuzzy logic), היינו צריכים להשתמש בעץ החלטה ראשוני המבוסס על ידע מוקדם (באינטרנט מצאנו מס' שמות לדברים דומים לכך כגון expert knowledge או deduction reasoning). בפרויקט נשתמש בסגנון עץ משלנו שאנו פיתחנו, המבוסס על חלוקה למקרים ונתינת הסתברויות לבחירת פעולה בכל מצב.**
* **העצים יהוו סוכני "למידה" עבור ה-PGM, מכיוון שהם יהיו שחקנים "חכמים" במידה מסוימת, בעלי הבנה בסיסית (לכל הפחות) של המשחק, שבעזרתם ה-PGM ילמד לשחק מול שחקנים ברמה גבוהה יותר, כאשר הוא עצמו אמור להתעלות עליהם ולהוות שחקן "חכם" יותר.**
* **חשוב להדגיש כי כלים אלו ברובם הם לא כלים שלמדנו בקורס. הם אכן כלים שמבוססים על הנושאים בחיפוש, ייצוג ידע ,למידה ותיאוריית המשחקים שלמדנו בקורס, אך מכיוון שהכלים שלמדנו בקורס בלבד אינם מספיקים כדי לפתור את הבעיה שלנו, אז למדנו בעצמנו את השיטות המצוינות מעלה בעזרת חקר באינטרנט.**
* **בנוסף, מימשנו בדרך יותר סטנדרטית של למידה – רשת נוירונים (כפי שכבר למדנו בקורס), על מנת להשוות את יעילותה וביצועיה מול הרשת הבייסיאנית (ה-PGM), תוך הערכה שרשת נוירונים תהיה פשוטה יותר למימוש אך פחות תתאים לבעיה הנ"ל ולכן תביא תוצאות פחות מספקות.**
* **תהליך הלמידה העיקרי של הסוכנים אמור להתבצע כאשר לאחר למידה מול עצי ההחלטה, הם ישחקו אחד כנגד השני , וכך ישתפרו ביצועיו גם של סוכן ה-crew וגם של סוכן ה-impostor. לפי מה שקראנו באינטרנט, זהו למעשה תהליך של multi agent system (שבו לא דנו בקורס), והוא למעשה מתאר כיצד סוכנים יכולים ללמוד אחד באמצעות השני. לדוגמה במשחק הנ"ל ה-impostor-ים יכולים ללמוד לתת דיווחים מסוימים בעקבות הבנה של מה גורם ל-crew להצביע בצורה מסוימת, ובעקבות כך ה-crew-ים יכולים ללמוד לדעת להתייחס בצורה שונה לדיווחים שמתבצעים במהלך ההצבעה.**
* **החזון המרכזי לכך וההכרה ביתרון המשמעותי שהדבר יכול לתרום ללמידה של הסוכנים שלנו, מבוסס על הסרטון** [**הנ"ל**](https://www.youtube.com/watch?v=kopoLzvh5jY)**, שגם מפורט ב"חומרי עזר מומלצים לקריאה וצפייה" ומתאר שילוב בסביבה מרובת סוכנים במשחק אחר. הצפייה בסרטון מומלצת, לדעתנו הוא נותן אינטואיציה מאוד חזקה למה שניסינו לעשות.**
* **Among Us הוא משחק שמכיל המון פרטים, יש המון פרמטרים לשקלל ולהתייחס אליהם במשחק על מנת לקבל תוצאות מקסימליות, בפתרון שלנו התעמקנו מאוד בפרטים הקטנים של התרגיל, מה שדרש זמן, התעסקות והחזקת כמות מידע גדולה מאוד על המשחק (כפי שכבר פירטנו מקודם). לדעתנו זו הדרך הכי טובה לקבל תוצאות מיטביות שידמו חשיבה של אדם אמיתי במשחק, וייתנו מקום לכל האספקטים במשחק בצורה מרבית – לדוגמה התעמקות בהקשר להתייחסות לא רק ל"האם מצאנו גופה", אלא ל"איפה מצאנו", "מי הגופה", ו"מה הקרבה של כל שחקן אליה", או לדוגמה התייחסות ל"היכן ומה עושה כל שחקן במהלך הסיבוב", "היכן הוא נמצא", "מתי הוא נמצא" וכד'.**
* **לסיכום, נשתמש בעצי החלטה על מנת לתת אימון ראשוני לסוכנים הלומדים שניצור, אשר אמורים ללמוד באמצעות חישוב הסתברויות למאורעות בהתאם למצבים לבחור במצב הנכון, ובהמשך ילמדו זאת גם באמצעות משחק אחד נגד השני.**

**הסוכן הרנדומלי והתאמתו לבעיה**

* **נעשה שימוש בסוכן רנדומלי לצורך בדיקה ראשונית של המשחק, ובעיקר בשביל לקבל פרספקטיבה חזקה לכמה הסוכנים שמבוססים על עצי החלטה והסוכנים הלומדים הם אכן מספיק "חכמים" ומבצעים החלטות נכונות שיותר טובות מההחלטות הרנדומליות.**
* **הסוכן הוא רנדומלי לחלוטין ומבצע בחירה מתוך רשימת המהלכים שלו באופן אקראי לחלוטין.**

**עץ ההחלטה עם הידע המקדים והתאמתו לבעיה**

* **נעשה שימוש בעץ ההחלטה עם הידע המקדים לצורך ייעול ושיפור תהליך הלמידה, כדי שהסוכנים ילמדו מול שחקנים מורכבים יותר ולא רנדומליים לחלוטין, וגם על מנת שיהיה לנו גורם שיהווה שחקן ברמה מתקדמת יותר, שנוכל לבדוק מולו את רמתו ויכולותיו של הסוכן הלומד שלנו. יצרנו מראש עצי החלטה מובנים שמתבססים על ידע מוקדם של המשחק. השחקנים מבוססי-העצים יודעים מה לעשות בכל מקרה, ולמעשה הם מהווים באופן די מרשים התנהגות מלאכותית שמייצגת את הידע האנושי במשחק, ובכך תיתן בהמשך יתרון לסוכנים הלומדים שאמורים להתעלות ביכולותיהם על יכולותיהם של שחקנים אמיתיים.**
* **בחירת הפעולה בכל מצב מיוצגת ע"י התפלגות מסוימת, שנותנת לנו הסתברות כלשהי לביצוע כל משימה אפשרית (ומומלצת) במצב זה, ובאופן רנדומלי השחקן בוחר מה אכן לעשות במצב זה. כך אנו מייצגים את אפשרויות הבחירה למצבים, על מנת קודם כל ליצור שחקן שהוא אינו צפוי בתגובותיו (בדיוק כמו שחקן אמיתי), ובעיקר על מנת לפעול כמו stochastic hill climbing שלמדנו עליו בקורס, שלעיתים בוחר בפעולות שלא נראות הכי "יעילות" או "חכמות", במטרה לפתוח לעצמו דרכי פתרון נוספות ולהכיר מצבים נוספים על הלוח. עשינו זאת במטרה שהסוכנים שלנו לא יהיו מקובעים ויפעלו כל פעם באותו דרך, אלא ילמדו מצבים חדשים ואולי יבינו שאלו דווקא הדרכים היותר טובות לבחור בהן (לדוגמה נתנו הסתברות של 94% "בלבד" לבחירה של מי שהואשם ברצח עבור כאלו שלא ראו אותו רצח, וזאת על מנת להימנע ממצב מוחלט שבו שאר ה-crew מצביעים "על עיוור" יחד עבור כל דיווח, אלא בוחנים את האופציה גם לא להאמין לזה).**
* **ההחלטה תתקבל על בסיס כך שבכל מצב, כלומר בכל סיבוב של המשחק שבו הסוכן נמצא במקום כלשהו ועליו לבצע החלטה כיצד לזוז/מה לעשות כעת, הסוכן מחזיק את כל המידע הדרוש (פרמטרי המשחק – ה-observations), ולפיו "ינווט" בהחלטתו להסתברות להחלטה המתאימה על בסיס התנאים המדויקים שבה היא עומדת (למשל לפי כמות ה"המתחזים" ו"חברי הצוות" שבטווח ראייה באותו זמן). הוא יפעל בהתאם לכך, והפעולה תיוצג באמצעות בחירה תזוזה (ימינה, שמאלה, למעלה, למטה) או באמצעות בחירה מעשה (רצח, קריאה לשחקנים אל הגופה, קריאה לפגישה, ביצוע משימה).**
* **המידע שהכנסנו לתוך עץ ההחלטה מבוסס על ההיכרות והניסיון שלנו מהמשחק, שיחקנו בעבר כמות רבה של משחקי Among Us, ושיחקנו עוד כמה משחקים במיוחד לצורך הפרויקט כדי בסופו של דבר להשיג "מוסכמות" ודרכי פעולה למצבי המשחק השונים – דרכים שברובן הן די טריוויאליות, אבל חלקן אינן טריוויאליות (למשל לא תמיד להאמין לדיווחים בעת הצבעה או לדעת כיצד להצביע בתור impostor). לדעתנו השחקנים מבוססי-העצים הם מספיק טובים ומנוסים, ועצם המשחק נגדם והניסיון לנצח אותם הם קשים ודורשים הבנה גבוהה של המשחק. ניתן לומר שמכיוון שהעצים האלו מייצגים את דרכי ההחלטה שלנו כשחקנים, שמבוססות על הניסיון שלנו ומוסכמות משחק וטיפים שקראנו באינטרנט, ניצחון כנגדם שקול לניצחון שחקנים אמיתיים ומנוסים.**
* **העצים שיצרנו יודעים לתת מענה לכל מצב במשחק, ויידעו לתת החלטה (שברוב המקרים תהיה הרציונלית והמיטבית לאותו מקרה) לכל מצב במשחק, וכמו שתיארנו כבר קודם בתיק העבודה, מדובר בכמות גדולה מאוד של מצבים, שהתמודדנו איתה בעזרת איחוד מקרים ובנייה מתוכננת בקפידה של העץ.**
* **כיוון שהדבר דרש מאיתנו חלוקה לתתי-עצים רבים, ומתוך רצון שלא לעשות חלוקה "גסה" מדי למקרים אלה, אלא לתת משמעות לכל הפרש קטן בפרמטרים (לדוגמה שיהיה הבדל בין "האם ראיתי מישהו לפני 2 ש' או לפני 3 ש' / 4 ש'" – מה שבמשחק יכול להוות הבדל משמעותי בהחלטה האם לחשוד במישהו או לא), השתמשנו רבות ב"פונקציות שקלול" שמתייחסות למס' פרמטרים רבים, משקללות אותם על בסיס נוסחאות שחישבנו בהסתמך על הידע שלנו כשחקנים, ואמורות להתאים ולהביא התפלגויות מתאימות למקרים רבים הרבה יותר, תוך צמצום משמעותי של העץ עצמו. לדוגמה השתמשנו בפונקציה: 5 \* כמות המשימות שעשיתי + כמות הזמן שעבר מתחילת הסיבוב, על מנת לתאר מה המהלך שלחבר צוות בזמן משחק רגיל עדיף לבחור – לעשות משימות או להסתובב בחדרים, כך שלא משנה מה יהיה הערך עבור כל פרמטר, תמיד תתבצע הבחירה ה"נכונה". נוסחאות כאלה מייצגות בצורה יותר טובה את המחשבה האנושית, מכיוון שלרוב בראש השחקן עושה שקלול בין פרמטרים, ופחות פועל על בסיס החלטות עם בחירה מוגבלת.**
* **לחברי הצוות יש עץ משלהם ולמתחזים יש עץ משלהם. עצים אלו מחולקים לעצים עיקריים של מצב הצבעה (דיווח מידע ובחירת הצבעה) או מצב משחק רגיל (תזוזה וביצוע פעולות). בחרנו שלא להרחיב כאן לגבי התוכן המדויק של העצים, על מנת לא להלאות בפרטים שקשורים ללוגיקה של המשחק ולמחשבות מאחורי פרטים משחקיים אלה. בנספח ב' (**[היפר קישור](#עץלוגיקה)**) מתוארים העצים הנ"ל במפורט וגם הסבר לוגי שעומד מאחוריהם.**

**PGM – הרשת הבייסיאנית והתאמתה לבעיה**

* **PGM הוא מודל הסתברותי שמבטא יחס של תלות בין משתנים רנדומליים, ונעשה בו שימוש בעיקר בתיאוריות סטטיסטיות והסתברותיות, ובמיוחד בהסתברויות בייסיאניות ובמערכות לומדות.**
* **הסיבות לשימוש במודל זה מפורטות בהקדמה (**[**היפר קישור**](#הסברלPGM)**).**
* **PGM היא רשת בייסיאנית שמייצגת אוסף של משתנים – במקרה שלנו משתנים מקריים בדידים (ולא רציפים) ואת יחסי התלות ביניהם באמצעות גרף לא מעגלי מכוון. מה שהרשת עושה זה לקחת כל מאורע ולחשב את ההסתברות שלו בהינתן גורמים שונים.**
* **המימוש שלנו עבור הרשת התבצע ע"י חבילת הפייתון pgmpy (מה שהתברר כבחירה מוטעית בדיעבד).**
* **סקיצה של הרשת הבייסיאנית שיצרנו מצורפת בנספח ג' (**[**היפר קישור**](#סקיצתרשת)**).**
* **השימוש ברשת מתבצע ע"י הזנת קודקודים של מידע שהשחקן מחזיק במהלך המשחק (ה-observations), והם מפורטים כ-22-28 הפרמטרים ב"תיאור הבעיה". עבור שחקני ה-crew, הרשת תחשב באמצעות הפרמטרים את ההסתברויות החדשות עבור כל מתחזה, ותשקלל את ההסתברויות כדי לקבל הסתברויות למקרי החיזוי הרצויים שהם: ניצחון, לשרוד סיבוב – עבור כל סיבוב, להצביע למתחזה – עבור כל משימה, לסיים את המשימות (אלו פרמטרים חשובים במשחק, הם סוג של "מטרות ביניים" של השחקן).**
* **את ההסתברות לכל "פרמטר חיזוי" נמשקל (כמו שעשינו במהלך הקורס להיוריסטיקות למיניהן) תוך כדי מחשבה כמה כל פרמטר חשוב ועוזר במטרת העל, שהיא ניצחון במשחק. בעזרתם נקבל "ציון" לכל מהלך שיתאר כמה עדיף לנו לעשות אותו.**
* **עבור כל מהלך אפשרי של השחקן בכל צעד נחשב את ה"ציון" הנ"ל ונבחר במהלך עם ה"ציון" המקסימלי.**
* **את בחירת המשקולות ביצענו על בסיס בחירה של מס' אנשים בעלי הבנה המשחק. אנחנו נריץ כל אחד מהם כמות מספקת של סיבובים, ולבסוף נבחר את השקלול שמביא את התוצאות הכי טובות, ונשתמש בו על מנת להמשיך ללמד את הסוכן.**
* **על אף יתרונות הרשת הבייסיאנית שציינו מקודם, היא מאוד כבדה חישובית, ואנחנו ננסה להבין במהלך ההרצה אם בכלל אפשר להשתמש בה בהתחשב בכמות המידע שאנו מתעסקים איתה.**
* **בנוסף, יתרון נוסף וחשוב של הרשת הבייסיאנית זה שהיא אינטרפטבילית ומניחה אי וודאות ומקלה על החישובים.**

**רשת הנוירונים והתאמתה לבעיה**

* **רשת נוירונים היא מודל מתמטי חישובי, שמשמש במסגרת ML. רשת מסוג זה מכילה בדרך כלל מס' רב של יחידות מידע – קלט ופלט, המקושרות זו לזו בקשרים שעוברים דרך "שכבות חבויות", כאשר הרשת מכילה מידע על החוזק והקשר בין השכבות.**
* **בקורס למדנו על רשתות נוירונים ככלי חשוב לאלגוריתמי למידה.**
* **למימוש רשת הנוירונים נשתמש במודלsklearn.neural\_network.MLPRegressor .**
* **רשת הנוירונים תיראה דומה לסקיצת הרשת בייסיאנית, עם התייחסות לכל הפרמטרים שאנחנו מכניסים, אך תכלול פחות קשרים ולדוגמה לא תחשב את פרמטרי החיזוי עבור "מתחזים".**
* **השימוש ברשת מתבצע ע"י הזנת קודקודים של מידע שהשחקן מחזיק במהלך המשחק (ה-observations), והם מפורטים כ-22-28 הפרמטרים ב"תיאור הבעיה". עבור שחקני ה-crew, הרשת תחשב באמצעות הפרמטרים את ההסתברויות החדשות עבור כל מתחזה, ותשקלל את ההסתברויות כדי לקבל הסתברויות למקרי החיזוי הרצויים שהם: ניצחון, לשרוד סיבוב – עבור כל סיבוב, להצביע למתחזה – עבור כל משימה, לסיים את המשימות (אלו פרמטרים חשובים במשחק, הם סוג של "מטרות ביניים" של השחקן).**
* **את ההסתברות לכל "פרמטר חיזוי" נמשקל (כמו שעשינו במהלך הקורס להיוריסטיקות למיניהן) תוך כדי מחשבה כמה כל פרמטר חשוב ועוזר במטרת העל, שהיא ניצחון במשחק. בעזרתם נקבל "ציון" לכל מהלך שיתאר כמה עדיף לנו לעשות אותו.**
* **עבור כל מהלך אפשרי של השחקן בכל צעד נחשב את ה"ציון" הנ"ל ונבחר ללכת במהלך עם ה"ציון" המקסימלי.**
* **את בחירת המשקולות ביצענו על בסיס בחירה של מס' אנשים בעלי הבנה המשחק. אנחנו נריץ כל אחד מהם כמות מספקת של סיבובים, ולבסוף נבחר את השקלול שמביא את התוצאות הכי טובות, ונשתמש בו על מנת להמשיך ללמד את הסוכן.**
* **בנוסף, על מנת להתאים את הערכים שלנו לערכים שרשת הנוירונים יכולה לעבוד איתם, נפעיל על הערכים את הפונקציה SoftMax, שתנרמל לנו ערכים כנדרש.**
* **כפי שמפורט בהקדמה, היתרון של הרשת הוא שהיא הרבה יותר מהירה ונגישה חישובית מאשר הרשת הבייסיאנית, אבל החיסרון הוא שאנו מצפים שיהיו לה תוצאות פחות טובות בשל אופי החישוב שלה, שלא מתבסס על הסתברות מותנית כפי שהיינו רוצים.**

**דרכי המימוש**

* **כיוון ש-Among Us הוא לא משחק עם קוד open source, לא הייתה לנו אפשרות להתממשק עם המשחק בעצמנו, ולכן יצרנו בעצמנו גרסה של המשחק ב-Python, ובנוסף יצרנו לו GUI חזותי באמצעות ספריית ה-tkinter ב-Python.**
* **מימשנו את הפרויקט באמצעות יצירת מחלקות עבור הלוח, הסוכנים, הסוגים השונים של סוכני ה-AI שלנו, ויצירת מחלקה ראשית שמריצה את המשחק.**
* **לצורך הרצת ה-PGM השתמשנו בחבילת הפייתון pgmpy, ובאמצעותה מימשנו רשת בייסיאנית של המידע, שאותה הרצנו על מנת לקבל החלטות. בדיעבד החלטה זו התבררה כטעות.**
* **אחת הבעיות הגדולות שעלו במימוש ה-Decision Tree הייתה: כיצד נלמד את הסוכנים (גם ה-crew וגם ה-impostor) לנוע על המפה בצורה הגיונית וחכמה בין משימות וחדרים?**

לשם כך השתמשנו בכלי בינה מלאכותית שלמדנו בקורס – A-star search algorithm. **יצרנו את המחלקה Search.py, שמממשת את אלגוריתם החיפוש A-star על בסיס היוריסטיקה שמתבססת על מרחק ה"מנהטן" של הנקודה מנקודת היעד, מה שעזר למציאת המסלול היעיל היותר מכל נקודה לכל נקודה במפה.**

**מהמסלול הכי קצר שלפנו בכל פעם (ע"י פונקציית עזר) את הצעד הראשון הנדרש על מנת להגיע לנקודה הראשונה במסלול זה, ושמרנו את המידע הנ"ל במילוני עזר.**

**מטרת הנקודות הראשונות במסלולים הכי קצרים היא לסייע בלימוד הסוכנים איך ללכת בין משימות או בין חדרים בצורה יעילה והגיונית, שמהווה דמיון לאופן שבו שחקנים אמיתיים זזים במשחק (הבחירה לאן ללכת היא עדיין תלוית-החלטה של הסוכן, והכלי שאנחנו הוספנו מיועד להראות לו כיצד להגיע לנקודה זו).**

**השימוש בחיפוש A-star הוא אינו מטרת הפרויקט, כלומר התשובה לשאלה "איזה צעד הסוכן בוחר לקחת?" בכל פעם לא משנה את התוצאות הרלוונטיות. עם זאת, מדובר בכלי נוסף שלמדנו בקורס AI, שעוזר לנו להגיע לפתרון בצורה יעילה וחכמה.**

* **את רשת הנוירונים מימשנו באמצעות המודל sklearn.neural\_network.MLPRegressor.**
* **מימשנו את הקוד בצורה מודולרית כך שיהיה ניתן להריץ אותו עם כמות מקסימלית של 15 שחקנים, ובנוסף מימשנו אותו כך שניתן יהיה לעבוד מול סוגים שונים של מפות שמתקבלות בקובץ JSON.**
* **את מפת המשחק מימשנו במיוחד בקובץ JSON והיא מבוססת על מפת המשחק הקלאסית.**

**תוצאות**

**הרצת הרשת הבייסיאנית ואי-ההצלחה החישובית**

* **לאורך כתיבת הפרויקט מימשנו את ה-PGM באמצעות חבילת הפייתון pgmpy, אך במהלך ניסיונות ההרצה גילינו שחבילה זו מוגבלת לעד 32 nodes בשל אופן המימוש שלה, מה שהפך את השימוש בה ללא רלוונטי לאור כמות ה-nodes הגדולה הרבה יותר שתיכננו להשתמש בה. בכל זאת צירפנו את המימוש של ה-PGM להגשה כדי שיהיה ניתן לראות מה ניסינו להריץ בעזרתו.**
* **ניסינו להשתמש בחבילת פייתון אחרת בשם pomegranate עבור הרשת הבייסיאנית. ניסיון ההרצה היה עם data של 3 nodes וכמה מאות דגימות, מה שלקח מעל לחצי שעה עבור כמות nodes שאפילו לא קרובה לכמות הרצויה. גם כשניסינו להריץ עם כמות data שיותר קרובה למה שהיינו רוצים בבעיה שלנו, קיבלנו בעיות out of memory וכדומה.**
* **ציפינו כבר מראש שהרצת רשת בייסיאנית תהיה כבדה מאוד חישובית. עם זאת, העובדה שלא היה ניתן בכלל להריץ אותה במלואה, או שזה לקח המון זמן (ברמה שהרצה מינימלית של אלפי דגימות, כלומר משחקים, הייתה לוקחת הרבה מעבר לזמן הנתון לכל הפרויקט), הייתה אכן מפתיעה, ודרשה מאיתנו היערכות מחדש מבחינת המשך הפרויקט.**
* **לאחר מכן גילינו ששימוש ברשת בייסיאנית יכול להיות אפשרי אם עושים התאמה מיוחדת רק בין nodes מסוימים בחישוב הרשת, ולא "מחברים" בין כל ה-nodes האפשריים, מה שהביא לנו כמות מאוד גבוהה של חיבורים אפשריים בין nodes, כלומר nodes-משנה.**

**תוצאות לפני הלמידה**

* **הסוכן הרנדומלי והסוכן-מבוסס-עץ-ההחלטות-הלוגי עם המידע הידוע מראש והם קבועים.**
* **לכן קודם נבצע הרצה של משחקים שלהם אחד נגד השני, 100 משחקים לכל סוג משחק (סוג משחק – crew או impostor), ובכך נקבל פרספקטיבה של כמה טוב כל סוכן וגם נוכל לקבל סטטיסטיקה כללית של המשחק, של אחוזי הניצחונות הכלליים של המתחזים מול אלו של חברי הצוות, מה שיעזור לנו בהמשך בניתוח התוצאות.**
* **להלן התוצאות:**
* **כפי שניתן לראות, הסוכן-מבוסס-עץ-ההחלטות-הלוגי (logic) בעל ביצועים משמעותית יותר טובים מהסוכן הרנדומלי (random), ויש נטייה טבעית למתחזים לנצח.**

**תוצאות לאחר הלמידה**

* **כדי לאמן את רשת הנוירונים ביצענו fit של הרשת ל-2 סוגי הסוכנים של רשת הנוירונים (חבר הצוות והמתחזה) בעזרת הכנסת נתונים של משחקי עבר (כמשחקי דוגמה נגד ה-logic). תחילה הכנסנו נתוני 500 משחקים, שמהווים בפועל 500,000 דגימות עבור הסוכן שהוא crewmate ו-120,000 דגימות עבור ה-impostor. עבור כל משחק הרשת קיבלה את כל הפרמטרים ואת ה-observations עבור אותו משחק, ולבסוף את הקבוצה שניצחה במהלכו. למידה זו מהווה את השלב הראשון של המשחק.**
* **לאחר מכן, על מנת לקבל פרספקטיבה טובה למידת ההצלחה של הסוכנים, הרצנו 100 משחקי ניסיון כדי לבדוק את אחוזי ההצלחה של כל אחד.**
* **בשלב השני ביצעו הכנסה של נתוני עוד 500 משחקים, כך שנקבל סך הכל כ-1,000,000 דגימות עבור הסוכן שהוא crewmate ו-240,000 דגימות עבור ה-impostor. זה כבר יהווה לנו כמות די גבוהה של ערכים, ובנוסף ייתן לנו אינדיקציה טובה לגבי תהליך הלמידה של הרשת, כדי לראות כיצד והאם היא משתפרת.**
* **לבסוף בשלב השלישי הרצנו עוד 500 משחקים – סך הכל 1500 משחקים, 1,500,000 דגימות עבור crewmate ו-360,000 דגימות עבור ה-impostor, וכמו בשלב השני נקבל אינדיקציה.**
* **להלן התוצאות עבור ריצות של רשת הנוירונים בלבד, והצגת אחוזי הניצחונות בכל שלב:**
* **בנוסף, על מנת לקבל השוואה לגבי מידת השפעת כמות הסוכנים בתוך ה-crew על התוצאות, אחד מאיתנו שיחק משחק של 2 מתחזים רנדומליים נגד 1-7 סוכני crew של רשת הנוירונים, כאשר הסוכנים היו crew רנדומליים (גם פה היו 100 משחקים לכל מקרה, החישוב יהיה לאחר השלב השני של הלמידה):**
* **ניתן לראות כי סוכן ה-Neural Network הוא מעט טוב יותר מהסוכן הרנדומלי לאחר למידה בכמות של השלב השני, ויש השפעה ישירה לכך שככל שיש יותר סוכני NN ב-crew ככה הם מנצחים בסבירות גבוהה יותר. עם זאת אין הבדל משמעותי במקרים.**

**סיכום התוצאות**

* **כדי לקבל פרספקטיבה יחסית טובה למידת ההצלחה של הסוכנים, עלינו להתייחס לעובדה כי המשחק עצמו לא שקול ויש יתרון יחסי לאחד הצדדים במשחק. לשם כך ביצענו בדיקה בה אותו סוכן שיחק נגד עצמו בתפקידים השונים (פעם crew ופעם impostor). הכללנו גם את הסטטיסטיקה הכללית של המשחק, שאותה מצאנו באינטרנט** [**(7)**](https://gamingonphone.com/news/among-us-stats-show-imposters-having-higher-win-rate-than-crewmates/)**.**

**יש הבדל קל בין המשחק שלנו למשחק הכללי באינטרנט, אך כנקודת ייחוס להצלחה של סוכן מול סוכן, יש שאיפה ליחס טוב יותר מ-30%-70% לטובת ה-impostor.**

* **ניתן לראות ששחקן העץ (logic) מצליח מעבר למצופה, והוא משמעותית עוקף את הרף של 70-30 נגד הסוכן הרנדומלי, מה שהופך אותו באמת לסוכן טוב (ניצח ב-88%) בהשוואה לרשת הנוירונים.**
* **גילינו שרשת הנוירונים צריכה כמות גבוהה מעל למצופה של דגימות, ושגם הכמות של מאות אלפי ואף מיליוני הדגימות (פרמטרים אפשריים) שהייתה נראית לנו גבוהה, היא לא באמת כזו גבוהה והיה צריך עוד יותר דגימות מכך.**
* **אחד הדברים הכי מעניינים ולא צפויים במהלך התוצאות היה הלמידה של רשת הנוירונים במשחקי הסוכן שמבוסס עליה מול הסוכן שמבוסס על העץ – תחילה דווקא יש ירידה מסוימת (8% ניצחונות בלבד בתור crew בשלב השני אחרי שבשלב הראשון היו 20% ניצחונות, אבל אחרי שלב הלמידה השלישי אחוז הניצחונות עומד על 37%), דבר שלדעתנו נובע מאופי הלמידה של הרשת, שעם הזמן לומדת לשחק באופן פחות רנדומלי, מה שהופך אותה ליותר צפויה ויותר דומה לסוכן-מבוסס-עץ-ההחלטות-הלוגי. בהתחלה הרשת ברמה הכי נמוכה של משחק לוגי והגיוני, מה שיצר ירידה מסוימת באחוזי הניצחונות בהתחלה, אבל אז חלה אצלה עלייה ברמה (יותר אחוזי ניצחונות שלה) בשל העובדה שהסוכן שמבוסס עליה למד להתמודד מול הסוכן-מבוסס-העץ (שמבוסס על ידע מקדים), כאשר הוא משחק בצורה הגיונית יותר – עם הזמן לומד כך איך לשחק ומשתפר, וזה אכן מה שהתחלנו לראות בתוצאות.**
* **בעינינו התוצאות הנ"ל הן מכובדות ומצליחות "לפצח" את המשחק, וללמוד אותו בעזרת למידה בסדר גודל שאמנם גדול אך אינו ענק, וכמובן שכמו שציינו – עם עוד data יהיה ניתן להגיע לתוצאות אפילו יותר טובות.**

**מסקנות ודיון**

**כעת נענה על השאלה שעומדת מאחורי בעיית הפרויקט – האם המשחק Among Us ניתן לפתרון יעיל באמצעות AI?**

תמונה שמכילה טקסט, אדם, איש, מקורה

התיאור נוצר באופן אוטומטי

* **התוצאות של הריצות מעודדות ומעידות שהמשחק ניתן לפתרון ביעילות, והצלחנו לייצר סוכן לומד שטוב יותר משמעותית מהסוכן הרנדומלי וטוב במידה מסוימת מסוכן-עץ-ההחלטה (שמבוסס על ידע מקדים), גם בתור crew וגם בתור impostor. אלו תוצאות מעודדות שמראות שהמשימה אכן אפשרית עם זמן הרצה משמעותי יותר עבור הסוכנים-מבוססי-הרשתות (גם רשת ה-NN וגם רשת ה-PGM), אבל בכלים החישוביים שיש לנו כרגע לא ניתן לתת מענה לבעיית הפרויקט בעזרת רשת בייסיאנית, כפי שרצינו בתחילת הפרויקט.**
* **כדאי לאמן את סוכן-רשת-הנוירונים גם נגד סוכן-עץ-הלוגיקה (שמבוסס על ידע מקדים) וגם נגד הסוכן הרנדומלי, ובהמשך גם נגד עצמו (כחלק מניסיון ליצור "סביבה מרובת-סוכנים").**
* **רשת הנוירונים היא יעילה והיא כלי טוב למימוש הבעיה, אבל היא גם דורשת המון data וגם עדיין לא "מבינה" תוצאות אולטימטיביות (ניצחון משמעותי על ה-logic) כפי שציפינו מהרשת הבייסיאנית.**
* אז האם עדיפה רשת בייסיאנית או רשת נוירונים עבור בעיית הפרויקט? **הרשת הבייסיאנית היא אכן זו שמתאימה לבעיה, אך לצערנו היא לא התאפשרה להרצה מבחינה חישובית במסגרת מגבלות הפרויקט. רשת הנוירונים אמנם פחות מתאימה למקרה הזה, אך עם זאת היא מתפקדת בצורה מספקת בזמני ריצה טובים מאוד (תוך מס' שעות סך הכל).**

**השלכות ויעדים לעתיד**

* **כמסקנה עבור פרויקט עתידי – ניתן לייעל את הקשרים בין ה-nodes ברשת הבייסיאנית (כמו שהוסבר גם ב"דרכי הפתרון") ולמצוא כלים נוספים שבעזרתם אפשר להריץ את הרשת הבייסיאנית בזמן חישובי מתקבל על הדעת ובכך להביא (ככל הנראה) תוצאות טובות יותר.**
* **בשל העובדה שמדובר במשחק אותו משחקים נגד שחקנים אמיתיים ולכל אחד יש סגנון משחק שונה, היינו רוצים לנסות ללמד את הסוכן לשחק בצורה אופטימלית כנגד כל סוג שחקן (אנושי) שהוא משחק נגדו, תוך התחשבות בפרמטרים הכלליים עבור כל השחקנים ותוך נקיטת פעולות כנגד השחקן האנושי וחולשותיו/נטיות המשחק שלו.**
* **אם היה לנו יותר זמן, היינו רוצים להוסיף את כל הפונקציונליות האפשרית עבור המשחק וגם להוסיף מפות משחק שונות, כך שהסוכן ילמד לפעול גם במצבים החדשים האלו, מה שישפר את אמינות המשחק ויהפוך אותו לדומה אף יותר למשחק המקורי.**
* **באופן כללי היינו רוצים להריץ יותר איטרציות במהלך הפרויקט, וספציפית בזמן הלמידה ב"סביבה מרובת-הסוכנים" שבה הסוכנים לומדים לשחק אחד נגד השני. לדעתנו במקרים כאלה יש פוטנציאל שיפור גדול לאלגוריתם.**
* **בנוסף, כאתגר לעתיד היינו רוצים להעלות את הגרסה של המשחק לרשת ולתת לאנשים אמיתיים לשחק בה, ובעזרתה ללמד עוד את הסוכנים עם דגימות מגוונות ורבות במיוחד שגם יאמנו את הסוכנים למשחק מול שחקנים אמיתיים.**
* **כפי שכבר ציינו בהקדמה, קיים פתרון למשחק מאותה קבוצה, הפתרון ההוא משתמש לפי המאמר (**[**4**](https://news.mit.edu/2019/deeprole-ai-beat-humans-role-games-1120)**) בשיטה של counterfactual regret minimization, זו שיטה שלומדת לשחק ע"י משחק עצמי-נגדי שוב ושוב, ומשולבת בה תיאוריית הסקה. זו שיטת מעניינת שגם ראינו בפתרון של משחקים נוספים – למשל סוכן חכם עבור פוקר, והיינו רוצה להשוות את זמני הריצה והביצועים שלה לאלו שלנו.**
* **אחד הדברים החשובים שנוכחנו לנו בפרויקט הוא הביצועים המרשימים של הסוכן מבוסס ה-logic decision tree שיצרנו (שמבוסס על ידע מקדים). לדעתנו העץ כאמור הוא כלי חשוב למימוש בעיות דומות.**
* **מתוצאות הניסוי שעשינו – פתרון בעיות כמו במשחק Among Us יכול להיות מוצלח בעזרת AI. עם זאת, בעיות כאלו לא פשוטות מבחינה חישוביות ויש לכך משמעויות שנראות לנו די מעניינות. אנו חושבים שבינה מלאכותית מסוגלת להתמודד ולנתח לא רק נתונים יבשים ומידע קבוע, אלא גם אלמנטים יותר פסיכולוגיים ותלויי בחירה אישית של בני אדם, בהתחשב גם בנטיות שונות שלהם במהלך המשחק או אפילו במהלך המציאות.**
* **במהלך המשחק ביצענו בכל פעם ניתוח חשדות והכרעה מי האנשים ה"חשודים", כלומר בעלי מאפיינים נסתרים/שביצעו "רצח" במשחק, בהתבסס על מידע חלקי/השערות/מידע לא אמין בוודאות (שניתן מגורמים אחרים) ושקלול כלל הגורמים – כל זאת כדי לקבל החלטה על "מה קרה בפועל?" גם אם לא הייתה ידועה לנו התשובה על כך באופן סופי. בהמשך לכך, כשאלה לעתיד היינו רוצים לגלות – "האם ניתן להשתמש באמצעים שמתבססים על הכלים בהם השתמשנו בפרויקט כדי לפענח פשעים/חקירות משטרה במציאות?".**

**סיכום חוויית העבודה על הפרויקט**

* **מתחילת החשיבה בפרויקט חיפשנו רעיון מאתגר, שלא נוגע רק בנושאי הקורס שנלמדו או פותר רק משחק סטנדרטי ופשוט, אלא רעיון עם משמעויות מעניינות יותר.**
* **הרעיון של המשחק Among Us הגיע מכך ששיחקנו בו לא מעט בחופשת הקיץ בשנה שעברה, וגם בעיקר מכך שמדובר במשחק מורכב שיש בו הרבה אלמנטים שלא מסתכמים ב"איך לפעול?" אלא ב"מה נחשב טוב?" ו"איך מכמתים הצלחה במשחק?".**
* **העבודה על הפרויקט הייתה לא קלה, מבחינת לוגיקת המשחק ומבחינת ה-GUI שלו היינו צריכים ליצור הכל מאפס, תוך למידה של הכלים הנדרשים לשם כך. הפרויקט גם דרש מאיתנו ללמוד הרבה נושאים חדשים עבורנו ב-AI כמו PGM ו-fuzzy logic כדי לפתור את המשחק. לאורך כל העבודה על הפרויקט בשבועות האחרונים נשארנו מהבוקר עד הערב ביחד בבניין בית הספר לשם יצירת המשחק עצמו ויצירת סוכני הלמידה שיתחשבו בכל הפרטים האפשריים של המשחק. כל זאת על מנת לקבל תוצאות ברורות בהקשר לבעיית הפרויקט כפי שהגדרנו אותה.**
* **מבחינה אישית, הפרויקט הסתכם עבורנו כחוויה מאתגרת, מרתקת ומהנה. זו הייתה ההתנסות הראשונה שלנו באוניברסיטה בעבודה משותפת על פרויקט כקבוצה, ולא רק כיחידים או כזוג. מימשנו בעצמנו חלק מנושאי הקורס, והתנסינו גם בנושאי AI מעבר ל-scope של הקורס, ע"י חקר באינטרנט וגם בזכות עזרה של סגל הקורס. הפרויקט היה סיכום של החוויה המעניינת והמלמדת שלנו לאורך כל הסמסטר בקורס.**

**הרצת התוכנית**

**איך מריצים את התוכנית?**

**יש להתקין את החבילות והספריות הנדרשות בעזרת הרצת הפקודה pip install requirements.txt.**

**הכנו לשם ההרצה קובץ JSON שמכיל מפה קלאסית של המשחק וקונפיגורציית משחק עם 2 "מתחזים" ו-8 "חברי צוות".**

**כדי לבחור אילו סוכנים (רנדומלי/עץ לוגיקה/רשת נוירונים) יהיו חלק מה-crew ואילו סוכנים יהיו חלק מה-impostors, יש לערוך בהתאם את הקובץ example.json, כלומר לבחור היכן מבין השדות למקם את הערך 8 (שמייצג את מס' חברי ה-crew) – או ב-num\_crewmate\_random או ב- num\_crewmate\_tree או ב-num\_crewmate\_pgm (הכוונה בפועל היא ל-NN שדנו בה, לא ל-pgm שתכננו במקור), ולבחור היכן מבין השדות למקם את הערך 2 (שמייצג את מס' ה-impostors) – ב-num\_impostors\_random או ב-num\_impostors\_tree או ב-num\_impostors\_pgm (הכוונה בפועל היא ל-NN שדנו בה, לא ל-pgm שתכננו במקור). דוגמה לקונפיגורציה:**

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**אנו ממליצים להריץ את המשחק עם סוכן מסוג אחד נגד סוכן מסוג שני כדי להצליח להבחין בבהירות בהבדלים בין ביצועי הסוכנים.**

**בתוך הקובץ Main.py יש להוציא ממצב הערה את section הקוד הרלוונטי להרצה: האחרון – משחק יחיד שישוחק עם GUI, או האמצעי – מס' משחקים שישוחקו ללא GUI ויצירת קבצי CSV עבורם והדפסת סטטיסטיקה שלהם, או הראשון – מס' משחקים שישוחקו ללא GUI ויצירת קבצי CSV עבורם בלבד.**

**על מנת להפעיל את המשחק דרך מחשבי האוניברסיטה יש להריץ את הפקודה הבאה:**

**-"python3 Main.py"**

**בנוסף, בתוך הקובץ GameRunner.py הוספנו הערות על פרמטר האתחול train של רשת הנוירונים.**

**אם יש בעיה כלשהי, נשמח אם תפנו אלינו.**

**מה רץ ויוצג על הלוח?**

**

**מה שנראה על הלוח יהיה תצוגת GUI של המשחק שרץ, לוח בגודל 25X25 שמייצג את כל תכונות הלוח של המשחק, ריבוע שחור ייצג קיר (שלא ניתן לעבור/לראות דרכו), ריבוע חום ייצג את השולחן המרכזי, ריבוע ירוק בהיר ייצג משימה (לא יראו שחקנים בזמן משימה), ריבוע לבן ייצג מיקום ריק, ושאר הצבעים ייצגו את השחקן שנמצא עליהם. בעת הצבעה לרגע קצר השחקנים חוזרים להיות ליד השולחן המרכזי.**

**ביבליוגרפיה וחומרי עזר**

**"Games of Incomplete Information" של Jonathan Levin מאוניברסיטת סטנפורד מפברואר 2002.**

**"No limit: AI poker bot is first to bear professionals at multiplayer game", כתבה מ-"NATURE" מיולי 2019.**

**הערך בויקיפדיה של "Social deduction game".**

**מאמר מ- MIT Newsמנובמבר 2019.**

**דף האינטרנט** [**https://gamingonphone.com/news/among-us-stats-show-imposters-having-higher-win-rate-than-crewmates/**](https://gamingonphone.com/news/among-us-stats-show-imposters-having-higher-win-rate-than-crewmates/) **מדצמבר 2020.**

**חומרי עזר מומלצים לקריאה וצפייה:**

[**http://www.cim.mcgill.ca/~scott/RIT/researchPaper.html**](http://www.cim.mcgill.ca/~scott/RIT/researchPaper.html)**,Probabilistic Reasoning and Bayesian Networks – מאמר של Scott McCloskey משנת 2000.**

[**https://www.youtube.com/watch?v=kopoLzvh5jY**](https://www.youtube.com/watch?v=kopoLzvh5jY) **– סרטון על multi agent במשחקhide and seek, בערוץ היוטיוב של OpenAI.**

**נספחים**

נספח א' – כמות פרמטרים אפשרית עבור בחירת צעד

**בכל צעד של שחקן אנחנו קוראים לסוכן שלנו עם כמות של 28 פרמטרים מה-observations, והוא מבצע שקלול שלהם על מנת לבצע את המהלך. מצורפת מטה רשימה של הפרמטרים עבורimpostor (עבור crew יש מעט פחות פרמטרים, ורובם חופפים) כאשר לצד כל פרמטר מופיעה כמות הערכים האפשרית, ומצורף חישוב משוער של כמות הקומבינציות האפשריות לפרמטרים שמתקבלים:**

* **מה ה-kill\_cool\_down שלי (עוד כמה זמן אני יכול לרצוח שוב) – עד 30 מהלכים**
* **המשימות שזייפתי ביצוע שלהן – מוגבל ל-10 משימות**
* **זמן שזייפתי ביצוע משימות – מוגבל ל-80,000 מהלכים**
* **מיקום השחקן (קואורדינטה) – 400 קואורדינטות פנויות**
* **צבע השחקן – 12 שחקנים**
* **כמה זמן עבר מתחילת המשחק – מוגבל ל-80,000 מהלכים, אין הגבלה על הזמן**
* **כמה משימות נשאר לבצע**
* **אילו משימות נשארו**
* **מס' הסיבוב הנוכחי – יש עד 10 סיבובים**
* **מס' הסוכנים שאני רואה כרגע – מקסימום 12 סוכנים**
* **הזמן שנשאר עד שניתן "לקרוא" לשולחן (ניתן לבצע call meeting מתוך השולחן רק כעבור פרק זמן מסוים) – 20 מהלכים**
* **מס' הקריאות לשולחן שנותר לי לבצע (מס' הקריאות ל-call meeting באמצעות קרבה לשולחן הוא מוגבל) – עד 5 קריאות**
* **כמה זמן עבר מתחילת הסיבוב – מוגבל ל-80,000 מהלכים**
* **עבור כל סוכן – על ידי מי הוא נראה נרצח – 12 לסוכן**
* **עבור כל סוכן – האם הוא impostor – אינדיקטור של 2 לסוכן**
* **עבור כל סוכן – המיקום האחרון שראיתי אותו (קואורדינטה) – 400 לסוכן**
* **עבור כל סוכן – הזמן האחרון שראיתי אותו – 80,000 לסוכן**
* **עבור כל סוכן – כמה משימות שונות ראיתי אותו עושה – 10 לסוכן**
* **עבור כל סוכן – אילו משימות ראיתי אותו עושה – 12 לסוכן**
* **עבור כל סוכן – האם ראיתי אותו רוצח**
* **עבור כל סוכן – האם הוא מת – אינדיקטור של 2 לסוכן**
* **עבור כל סיבוב – מי מת בסיבוב הזה – מוגבל ל-12 סוכנים לסיבוב**
* **עבור כל סיבוב – מי הודח בסיבוב הזה (מי שקיבל הכי הרבה הצבעות) – מוגבל ל-12 לסיבוב**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב – מוגבל ל-12 לסיבוב**
* **עבור כל סיבוב – מי השתמש בשולחן באותו סיבוב – מוגבל ל-12 לסיבוב**
* **עבור כל סיבוב – איזו גופה דווחה באותו סיבוב – מוגבל ל-12 לסיבוב**
* **עבור כל סיבוב – מיקום הגופה שדווחה באותו סיבוב – מוגבל ל-400 לסיבוב**
* **עבור כל סיבוב – מי רצח (לפי המדווח) באותו סיבוב – מוגבל ל-12 לסיבוב**

סה"כ 28 פרמטרים.

**נחשב את כמות הקומבינציות האפשריות בהתאם:**

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**זוהי כמות ענקית, לשם השוואה מס' האטומים ביקום הינו 10 בחזקת 80, פה מדובר על בערך 10 בחזקת 274.**

נספח ב'- עץ ההחלטות המלא

**עץ החלטה עבור הצבעה – crew**

100% אדווח לשאר על הרצח

100% אצביע למי שראיתי רוצח

לא

כן

90% אצביע למי שדווח כרוצח

6% להצבעה אחרונה

4% למי שדיווח

כן

70% למי שחשוד כ"חשוד חזק"

15% לפי "טבלת חשדות"

15% לפי הצבעה אחרונה

100% לפי "טבלת חשדות"

לא

כן ויש לי חשוד חזק

כן

75% לפי "טבלת חשדות"

25% לפי הצבעה אחרונה

60% למי שחשוד כ"חשוד חזק"

20% לפי "טבלת חשדות"

20% לפי הצבעה אחרונה

כן ויש לי חשוד חזק

לא

כן

לא

20% למי שזימן את הפגישה

65% לפי "טבלת חשדות"

15% לפי הצבעה אחרונה

**טבלת חשדות**

**הפרמטרים: מרחק מן הגופה (המרחק שממנו שחקן דיווח על הגופה מהמיקום שראיתי אותו בו בפעם האחרונה), זמן שעבר (לפני כמה זמן ראיתי אותו), כמה זמן ראיתי אותו שלא בזמן ה-kill\_cool\_down הראשוני שבו אף שחקן לא יכול להרוג, כמה משימות ראיתי אותו עושה, כמה פעמים הוא הצביע לי (אלו פרמטרים שאמורים להעיד על כך שהבן אדם "חשוד"). החישוב הוא:**

**החשיבה מאחורי נוסחה זו היא שברוב המקרים חבר הצוות יאמין לדיווחים על רצח או על מציאת גופה ויצביע לפיהם, ואם לא קיים חשד רציני כלפי אחד מן האנשים הוא יבצע את החישוב על פי שקלול של כמה אמינות יצר כל שחקן אחר במהלך הסיבוב. החישוב נועד לתת משקל חיובי אם המרחק היה קרוב לגופה ושלילי אם לא, ובמקביל הוא נותן חשיבות לזמן שעבר וכך מגדיל את "אמינות הדיווח".**

**עץ החלטה עבור משחק – crew**

100% לקרוא לפגישה/לדווח

לא

כן

90% לדווח על גופה

10% לזוז רנדומלית

כן

100% לעשות משימות

לא

כן

225%לקרוא לפגישה

75% לפי "שקלול כללי"

2

לא

כן

לא

אם אין משימות – רק להסתובב בין חדרים.

נחשב את "שקלול כללי" לפי: 5 \* כמות המשימות שעשיתי + כמות הזמן שעברה מתחילת הסיבוב.

אם "שקלול כללי" הוא מעל 50% אז הפעולה העדיפה תהיה לחפש בחדרים, אחרת היא תהיה לעשות משימות.

נבחר לבצע ב-80% את הפעולה העדיפה וב-20% את הפעולה הלא עדיפה.

**החשיבה מאחורי כך היא שככל שהמשחק מתקדם יש יותר מידע שניתן לאסוף/גופות שאפשר למצוא, ולכן עם הזמן השחקן ייטה לעזוב את המשימות ולהתחיל לעבור בין החדרים ולחפש מידע על מנת למצוא את הרוצח.**

**שחקן טוב הוא שחקן שיודע לשלב בין ביצוע משימות לבין איסוף מידע ומציאת רוצח, והעץ מיועד להדריך אותו כיצד לעשות זאת.**

**עץ החלטה עבור הצבעה – impostor**

25% להאשים ולהצביע crew רנדומלי

75% לא להאשים אף אחד ולהצביע לפי "שקלול כללי"

90% לרוצח

10% לפי "שקלול כללי"

לא

כן

כן אבל לא אותי

כן

אותי

90% למי שהאשים אותי

10% לפי "שקלול כללי"

70% מצביע לו

30% לפי "שקלול כללי"

70% למואשם

30% לפי "שקלול כללי"

לא

כן ומאשים אותי

כן

50% למתחזה המואשם

40% למדווח

10% לפי "שקלול כללי"

כן ומאשים את המתחזה האחר

לא

100% לפי "שקלול כללי"

**השקלול הכללי**

**4 \* מס' ההצבעות נגדי + מס' ההצבעות נגד המתחזים האחרים.**

**לאחר מכן הצבעה ב-15% למתחזה השותף/המתחזים השותפים, ב-55% למקום הראשון בשקלול וב-30% למקום השני בשקלול.**

**החשיבה מאחורי כך היא שבשלב ההצבעות מטרת ה"מתחזה" היא "למזער נזקים", אם יהיה צורך הוא יצביע כנגד המתחזה השותף כדי לא לסכן את עצמו, אך לא יעשה זאת בכל מקרה אפשרי וינסה להימנע מזה. בכל אופן מטרתו תהיה "להנמיך פרופיל" ולא לסכן את עצמו בשלב זה.**

**עץ החלטה עבור משחק –impostor**

כן

100% משימות

95% לדווח על הגופה

5% לזוז רנדומלית

לא

כן

לא

כן

לא

20% לדווח על הגופה

80% לזוז רנדומלית

Crew 1

2 impostor

2 crew

2 impostor

1 crew

1 impostor

55% לרצוח

45% לעשות משימות

85% לרצוח

15% לעשות משימות

70% לרצוח

30% לעשות משימות

לא

כן

15% לעשות משימות

85% להסתובב בין החדרים

65% לעשות משימות

35% להסתובב בין החדרים

**החשיבה מאחורי כך היא שבמהלך המשחק המתחזה ינסה קודם להתנהג כמו חבר צוות, ולא להיתפס ליד גופה כשלא מדווח, ואם הוא לא במצב כזה אז הוא יבחר בין האופציה להסתובב בין החדרים ולמצוא את ההזדמנות הטובה לרצח (על פי עץ ההחלטה שקובע לו באיזו הסתברות לרצוח במצבים שונים) לבין האופציה לעשות משימות ובכך להגביר את האמינות שלו.**

נספח ג'- סקיצת רשת בייסיאנית

**פרמטרי חיזוי של המתחזים – החשד בכל אחד**

**פרמטרי חיזוי רצויים – הסתברות ניצחון, הצבעה למתחזה, לשרוד את הסיבוב ועוד...**

פרמטר פעולה נוכחי

**פרמטרים מה-observations (מפורטים ב"תיאור הבעיה")**