פרוייקט סיום – רשתות תקשורת

פרויקט סיום –קורס רשתות תקשורת, אוניברסיטת אריאל,
בהעברת פרופ׳ עמית דביר

מגישים:
נריה פילבר – 211377700
איתי שגב- 315112482
שלומה טימסיט – 332376813
טליה כהן - 322797002

תוכן עניינים:

חלק א: שאלות פתוחות

שאלה 1:

משתמש מדווח שהעברת הקבצים שלו איטית, ואתה צריך לנתח את שכבת התעבורה כדי לזהות את הסיבות הפוטנציאליות לכך. אילו גורמים יכולים לגרום להעברה איטית וכיצד היית פותר אותה?

תשובה: הגורמים שעלולים להשפיע על העברה איטית הם

1. **עומס ברשת** – כאשר הרשת עמוסה בחבילות, יותר מידי חבילות עוברות ברשת ונוצר עומס בטיפול בהן. העברת הקבצים הופכת להיות איטית מכיוון שברגע שקיים עומס ברשת חבילות יכולות להיאבד ובנוסף לפגוע בתקשורת.

- גודל הבאפר: כאשר גודל הבאפר לא מספיק גדול וקיים עומס של חבילות יכול להיגרם מצב שלא מספיקים לטפל בכל החבילות שנמצאות בבאפר ואז אין מקום להכניס עוד חבילות חדשות, דבר הגורם לאיבוד חבילות. ברגע שנאבדת חבילה נדרש לשלוח אותה מחדש וכתוצאה מכך ההעברה של כלל המידע שצריך להיות מועבר הופך להיות איטי יותר כיוון שיש צורך לשלוח את החבילה מחדש.
- עיכובים בתור של הנתב: חבילות נתונים מחכות בתור של הנתב על מנת להישלח. במצב בו התור ארוך והחבילות מחכות זמן רב הנתב עלול לזרוק חבילות כדי להפחית מהעומס, מה שגורם למעין " איבוד חבילות " ודורש שליחה של חבילה מחדש. דבר זה מעכב את העברת הקבצים הכוללת.

2. **מהירות RTT-** RTT מחושב על ידי זמן העיבוד של הנתבים, קצב השידור, אורך הקו, מהירות התפשטות הקווים וגודל האובייקט המועבר. כל אלו משפיעים על מהירות ההעברה של האובייקט.

   זמן עיבוד של כל הנתבים- ברגע שזמן העיבוד בכל נתב לוקח יותר , ההעברה מתעכבת.

   קצב שידור- ככל שהוא איטי יותר ההעברה מתעכבת יותר.

   אורך קו-ככל שאורך הקו ארוך יותר להעברה לוקח זמן רב יותר ולכן מתעכבת.

   מהירות התפשטות הקווים- ככל שהמהירות של ההתפשטות איטית לוקח יותר זמן לחבילות לעבור , כמו כן זה מושפע גם מגודל האובייקט המועבר (אם הוא גדול או קטן) ולכן ההעברה הכוללת מתעכבת.

3. **רוחב פס נמוך-** הקצב שבו נתונים יכולים לעבור ברשת נמוך מהנדרש עבור אובייקטים מסוימים.

   דבר זה משליך על כך שפחות אובייקטים מעוברים במקביל, ישנם עיכובים והאטה בהעברה של הקבצים. רוחב פס נמוך גורם לכך שלוקח זמן רב יותר להעביר את המידע ולכן גורם לעיכובים.

4. **תיעדוף חבילות ברשת-** אם הגדרות הרשת נותנות תעדוף לחבילות מסוימות ברשת ולא לחבילות של המשתמש הנוכחי אז יקח יותר זמן עד שכל החבילות שלו ישלחו.

כיצד היינו פותרים את הבעיה ?

מנגנון בקרת עומסים- ברגע שנזהה איבוד חבילות ברשת ועומסים רבים נוכל להשתמש בבקרת עומסים. לבקרת עומסים יש כמה מנגנונים בניהם חלון שליחה ו- AIMD .

1. **חלון שליחה-** בפרוטוקול TCP , גודל חלון השליחה קובע כמה נתונים את יכולה לשלוח בבת אחת מבלי לקבל אישור ACK על כל חבילה מהמקבל.
   ברגע שמגדירים חלון שליחה שהוא קטן מידי ביחס לכמות החבילות שמועברות ולקצב העברת החבילות יכול להיווצר מצב של עומס ובכך להאט את העברת החבילות.
   כלומר, ניתן להעביר כמות קטנה של חבילות בכל פעם עד לקבלת אישור ולכן ההעברה היא איטית. ולכן נגדיר גודל חלון מתאים שיאפשר העברה של מספיק חבילות בבת אחת כך שלא יוצר עומס ברשת.
   נשתמש בפתרון זה כאשר נזהה המתנות רבות מידי ברשת.
2. **AIMD** - פועל כך שהוא מגביר את הקצב של החבילות לאט לאט אך ברגע שנוצר עומס הוא מקטין אותו בבת אחת.
3. **הגדרות תיעדוף הרשת-** נשנה את הגדרות תיעדוף הרשת כך שהמשתמש הנוכחי יהיה זה שמקבל את העדיפות ולא חבילות של משתמש אחר.
4. **Timeout-** ניתן לשנות את הtimeout בהתאם לעומס הקיים ברשת.
5. **שדרוג החומרה-** במתגים, נתבים וכרטיסי רשת ישנים.
6. **הוספת נתבים ושרתים-** מאפשר לחלק את העומס בין יותר שרתים ונתבים ולכן העומס פוחת.

<u>שאלה 2:</u>

נתח את ההשפעות של מנגנון בקרת זרימה של TCP על העברת נתונים.

איך זה ישפיע על הביצועים כאשר לשולח יש כוח עיבוד גבוה משמעותית מהמקבל?

פתרון:

מנגנון בקרת זרימה של TCP הוא מנגנון שבה לפתור את הבעיה שהמקבל לא יוצף בחבילות מהשולח כך שהוא לא מספיק לעבד את כולן ונוצר עומס של חבילות שמצטברות.

TCP פותר בעיה זו באמצעות שדה בכותרת הTCP שנקרא rwnd שמגדיר כמה נתונים המקבל יכול לקבל בכל רגע נתון.

מנגנון זה עוזר לשולח להתאים דינמית את קצב שליחת החבילות לפי כמה מקום יש למקבל לקבל חבילות.

במידה ואין יותר מקום למקבל הוא יכול לעדכן את השולח בכך שהחלון הוא אפס ולא ניתן לשלוח יותר חבילות (דבר זה מונע איבוד חבילות ושליחה מיותרת של חבילות שלא יוכלו להתקבל אצל המקבל ולכן יאלצו להישלח בשנית).

במידה והתפנה, המקבל יכול לעדכן את השולח בכך. מצב זה נקרא TCP Persist .

TCP Persist state – בשביל להתגבר על מצב בו העדכון הולך לאיבוד השולח מידי פעם שולח חבילה קטנה והמקבל מגיב בACK שכולל את החלון העדכני. כלומר כמה חבילות הוא יכול לשלוח.

<u>איך זה ישפיע על הביצועים כאשר לשולח יש כוח עיבוד גבוה משמעותית מהמקבל</u>?

במצב בו לשולח יש כוח עיבוד גבוה יותר משל המקבל נוצר מצב בו למקבל לוקח זמן רב לנתח יותר את החבילות ולכן חלון הזמן שלו מצטמצם עד כדי 0 .

ברגע שהחלון מגיע לאפס, השולח לא יכול לשלוח עוד חבילות. ובכך מתבזבזת היכולת שלו להעביר חבילות רבות בקצב מהיר. דבר זה גורם להשהיה כוללת של השולח כך שהוא מחכה לאישור מהמקבל לשלוח עוד חבילות. וקצב התקשורת מוגבל מאוד כך שייוצרו מלא פעמים שיהיה בחלון אפס. במצב זה השולח מבזבז את יכולת כוח העיבוד המהיר שלו.

כאשר לשולח יש כוח עיבוד מהיר יותר ,המקבל מחזיר בכל פעם חלונות קבלה קטנים יותר לשולח כדי להאט את קצב שליחת הנתונים.

חלונות קבלה קטנים מגבילים את כמות הנתונים שהשולח יכול לשלוח לפני שהוא מקבל ACK מהמקבל. דבר זה יגרום לצוואר בקבוק בקצב העברת הנתונים, מכיוון שהשולח לא יכול לשלוח נתונים בקצב המקסימלי האפשרי לו.

<u>שאלה 3:</u>

נתח את תפקיד הניתוב ברשת שבה קיימים מספר מסלולים בין המקור ליעד. כיצד בחירת המסלול משפיעה על ביצועי הרשת, ואילו גורמים יש לקחת בחשבון בהחלטות הניתוב ?

תשובה :

ברשת קיימים מספר נתבים בין המקור ליעד. תפקיד הניתוב ברשת הוא לנתב את המידע ברשת בין המקור ליעד בעזרת הנתבים בדרך הטובה ביותר ובנוסף מאזן עומסים של הרשת.

בחירת המסלול האופטימלי היא קריטית לשיפור ביצועי הרשת ולהבטחת אמינות התקשורת.

<u>ישנם כמה גורמים המשפיעים על בחירת המסלול :</u>

רוחב פס – במידה והוא רחב יותר הוא יאפשר העברת נתונים במהירות גבוהה יותר ובכך ימנע עיכובים וישפר את ביצועי הרשת.

עומס על נתבים ברשת -אם הנתבים במסלול עמוסים זה עלול לגרום לעיכובים ולאיבוד חבילות ולכן בחירת מסלול עם נתבים פחות עמוסים ישפר את ביצועי הרשת.

אובדן חבילות- בחירת מסלול עם פחות נקודות כשל הוא מסלול אמין יותר ובטוח יותר עבור חבילות ובכך מונע צורך בשליחת חבילות כמה פעמים בשל איבודן ובנוסף משפר את אמינות התקשורת.

אורך פיזי של המסלול- בחירת מסלול קצר יותר מבחינת כמות נתבים או מרחק בין נתב לנתב יגרום לעיכובים קטנים יותר.

לדוגמא מסלול שעובר בין 3 נתבים עדיף על מסלול שעובר בין 5 נתבים אם שאר הגורמים של המסלולים דומים.

לסיכום, בחירת המסלול משפיעה על ביצועי הרשת בכך שהיא קובעת כיצד החבילה עוברת מהיעד למקור(דרך איזה נתבים).

מסלול עם רוחב פס גבוה יאפשר העברת נתונים מהירה יותר, בעוד מסלול עם עיכובים נמוכים ישפר את זמן התגובה. עומס נמוך על הנתבים לאורך המסלול יקטין את הסיכוי לאיבוד חבילות ולעיכובים נוספים. בנוסף, מסלול קצר יותר מבחינת כמות הנתבים או המרחק יפחית את העיכובים הכוללים.

בחירה נכונה של מסלול תוביל לשיפור בביצועים , באמינות , ובמהירות התקשורת ברשת תוך מניעת אובדן חבילות ויצירת עומס וצוואר בקבוק ברשת.

נבנה גרף בין כל הנתבים הקיימים ברשת . כל נתב הוא קודקוד וכל קשר בין נתב לנתב שקיים הוא צלע. ונבנה פונקציית משקל לצלעות שמתחשבת בכל השיקולים שפירטנו להעיל. לאחר מכן נבצע דייקסטרה על מנת לקבל את המסלול הקל ביותר.

<u>שאלה 4:</u>

כיצד MPTCP משפר את ביצועי הרשת?

תשובה :

תחילה נסביר מהו MPTCP ולאחר נסביר איך הוא משפר לנו את ביצועי הרשת.

MPTCP הוא בעצם הרחבה של פרוטוקול TCP הרגיל, היתרונות שלו הוא בעזרת כך שמשתמשים במספר נתיבים שונים על בו זמנית עבור אותו חיבור נוכחי, דבר אשר קורה בצורה שקופה לאפליקציה, מבחינת האפליקציה הוא מממש TCP רגיל לחלוטין בעוד שמאחורי הקלעים הMPTCP דואג לפיצול הנתונים שליחתם וחיבורם מחדש.

הרעיון הכללי הוא להשתמש במספר זרמי נתונים בו זמנית על מנת שזרימת הנתונים תהיה מהירה יותר כמובן האם קיים מספר דרכים של זרימת נתונים למקור (רשת סלולארית, WIFI, קווי..).

כעת נבחן את אופן השיפור של זה על ביצועי הרשת.

ניצול מספר נתיבי רשת:

בTCP רגיל אנו שולחים את כל המידע דרך נתיב יחיד, סלולר, WIFI, קווי וכו', באמצעות פרוטוקול זה ניתן לשלוח את הנתונים דרך יותר מנתיב 1 דבר אשר מפחית את העומס בנתיב ספציפי, ולכן גם קצב העברת הנתונים גדל, מכיוון שלכל הנתונים יש כרגע יותר מנתיב 1 למעבר.

שיפור אמינות ועמידות הרשת לתקלות:

פרוטוקול זה יודע להתמודד עם כשלים בנתיבי השליחה ולהתמודד עם זה באמצעות שליחה בנתיבים חדשים וכו׳, כלומר כאשר קיימת בעיה בנתיב מסוים, האטה (עומס) בנתיב, ניתוק של איזשהו נתיב (לדוגמא פתאום נפלה הרשת הסלולארית) אז זה לא גורם לניתוק החיבור בין הלקוח והשרת (דבר אשר היה קורה בTCP רגיל), אלא פרוטוקול זה יודע להתמודד עם זה ולהעביר את התעבורה שהייתה אמורה לעבור בנתיב הבעייתי לנתיבים אחרים שכן פועלים מבלי לנתק את החיבור, דבר זה מעלה את אמינות התקשורת.

איזון עומסים:

כפי שפירטנו בסעיף מעל, ובהסבר על הפרוטוקול, ברגע שיש לפרוטוקול זה אפשרות להעביר את המידע דרך מספר נתיבים בו זמנים אנחנו גורמים לניצול מירבי של המשאבים ברשת, אנחנו מפחיתים את זמן השהייה של חבילות בנתב מסוים (לא צריכה לחכות שנתיב ספציפי יתפנה מכיוון ששולחים במספר נתיבים) , דבר אשר מקטין בין היתר את העומס על נתבים (לא כל החבילות מחויבות לעבור לנתב הבא דרך הנתיב הספציפי, יש עוד נתיבים לנתבים אחרים גם).

לכן לסיכום, באמצעות פרוטוקול זה ניתן לנו האופציה לניצול מספר רב של משאבי רשת מה שמשפר לנו את הביצועים גם תחת סביבות רשת מורכבות ומאפשר איזון טוב יותר של העומסים ברשת, הן בנתיב מסוים, והן בנתב מסוים, וכן גם גורם לקשרים בין המקור והיעד להיות אמינים יותר, שכן

תקלה בנתיב אחד לא גורמת לקריסת הקשר אלא הפרוטוקול יודע לנתב את התעבורה בצורה חכמה ולתמוך בתקלות. מה גם שבקצה משפר את חווית המשתמש, הופך את כל הגלישה והכל למהירה יותר.

<u>שאלה 5:</u>

אתה עוקב אחרי תעבורת הרשת ומבחין באובדן מנות גבוה בין שני נתבים. נתח את הגורמים הפוטנציאלים לאובדן מנות בשכבות הרשת והתחבורה והמלץ על צעדים לפתרון.

תשובה :

כאשר אנו מנתרים תעבורת רשת, ומגלים שיש איבוד רב של חבילות ברשת בין נתבים שונים הדבר יכול להצביע על בעיות הן בשכבת הרשת והן בשכבת התעבורה.

תחילה נשים לב לבעיות שיכולות להיות שכבת הרשת:

יכול להיות תחילה שיש בעיות פיזיות/חיצוניות

כמו נזק בתשתיות (כבל שנחתך לדוגמא), תקלה בחומרה של הנתבים, או כל מיני חיבורים שהם לא בצורה מיטבית.

בנוסף יכולות להיות בעיות פנימיות בתוך הנתבים:

טבלאות ניתוב לא מעודכנות/שגויות, הגדרות שגויות בתוך הנתבים (יכול לקרות מאיזשהי תקלה בנתב), תור מלא בנתב כך שלא יכול לקבל עוד חבילות וזרק חבילה.

וכמובן כאשר יש עומס רב ברשת, כאשר יש עומס חריג, או שהרבה חבילות מנותבות מאיזשהי סיבה לאותו נתב מסוים כך שהוא עמוס ולא יכול לעבד את כמות הנתונים הנכנסת, מה שגורם לזריקת חבילות ואיבודם.

כעת נתייחס לבעיות היכולות להיות בשכבת התעבורה:

תחילה נשים לב שאם אננו משתמשים בפרוטוקול תעבורה (UDP) שזה פרוטוקול שהוא אינו אמין, הדבר תורם לכך שבמידה ותאבד חבילה גם לא נוכל לדעת מי זו ולטפל בזה, ולכן חבילה זו תאבד ולא תישלח שוב.

אם נתייחס לTCP נשים לב שחלון הזזה גדול מידיי, יכול לגרום לעומס ברשת ולאיבוד חבילות

שינוי פתאומי בחלון ההזזה יכול גם לגרום לאיבוד חבילות, כחלק ממנגנוני בקרת העומד של TCP

כאשר מדברים על TCP אנחנו גם יכולים להיות מושפעים מתקיפות DDOS מה שיכול לגרום לעומס רב על הרשת.

כעת נסביר על דרכים שבהן היינו מתמודדים על מנת לפתור את הבעיות בשכבת התעבורה והרשת לצמצום של איבוד החבילות

תחילה אפשר לעשות בדיקה פיזית של הנתבים והחיבורים ולבדוק שהחיבורים הינם תקינים, כלומר שאין שום תקלות פיזיות, לאחר מכן מה שהיינו עושים היה לבדוק את הגדרות הנתבים, נבדוק שהטבלאות מתעדכנות באמת, ואם לאחר שנטפל בהכל עדיין יאבדו חבילות יכול להיות שנעלה את

כמות הפעמים שהטבלה מתעדכנת ביחידת זמן מסוים. אם נדרש עדכונים מסוימים לתוכנה או לנתבים נדאג לעשות אותם.

לאחר מכן נרצה לבדוק האם התקלות קורות נרצה לטפל בעומס ברשת, נשים לב תחילה שבשאלה שאלו על כך שאנו יכולים להבחין שחבילות שאבדו, דבר אשר אי אפשר לשים לב אליו בתעבורת UDP מכיוון שאין אמינות, וגם אין איך לטפל בבעיה זו מעל UDP ולכן נתייחס לשיפורים בTCP.

נבדוק האם קיימים באמת עומסים חריגים, אם כן נבדוק האם הסיבה לכך זה התקפות מסוימות של הרשת (DDOS) ואם כן נפעיל חומות הגנה מתאימות ונטפל בכך באמצעי הגנה.

וכעת נטפל בפרוטוקול הTCP עצמו, נתאים את הזרימה והחלונות כך שנוריד את העומס ברשת, העומס יכול להיווצר גם משליחה חוזרת של חבילות שלא אבדו, ולכן נבדוק אופציה להגדיל את הטיימר לחבילה.

ולכן כוונון נכון של הפרמטרים כגון גודל חלון, טיימר יכולים לסייע בהורדת כמות החבילות האבודות.

באמצעות כלים אלו נוכל לטפל בבעיה זו ולהפחית אותה.

חלק ב – קריאת מאמרים וניתוחם

מאמר FlowPic encrypted internet traffic classification is as easy as image recognition:

1. שיטה זו לניתוח התעבורה המוצפנת באה לתת גישה חדשנית לסיווג התעבורה המוצפנת בעזרת הכלי שנקרא FlowPic גישה זו לוקחת תעבורת אינטרנט וממירה אותה לתמונה ואז משתמשת בכל מיני שיטות לניתוח תמונות על מנת לנתח את תעבורת הרשת,
באמצעות רשתות נוירונים וכו׳ (CNNs) היתרונות של גישה זו ומה שהוך אותה ליעילה יותר היא שהיא עובדת על זרימה חד כיוונית, לא צריך לתפוס את החבילות מ2 הצדדים אלא עובדת רק על צד אחד ספציפי,
בנוסף לכך באמצעות שיטה זו לא צריך לנתח כלל את תוכן החבילה עצמה היא לא נכנסת לתוכן החבילה, מה ששומר על פרטיות החבילות
ולבסוף היא מסוגלת להתמודד גם עם אפליקציות שהיא לא ראתה בעבר בשלב האימון מה שתורם לבדיקה אמיתית עבור כל מיני דברים שירצו לבדוק (לא מוגבלת רק למה שאימנו עליו את רשת הנוירונים)

2. בשיטה זו משתמשים ב2 מאפיינים יחסית פשוטים על מנת לבנות את התמונה, גודל החבילות וזמני ההגעה של החבילות
לגבי גודל החבילות רוב גדלי החבילות הן בדרך כלל לא יותר גדולות מ1500 בתים. (מעל 95 אחוז) לכן במידה ומתופסים חבילות שגודלן גדול יותר זורקים אותן
את החבילות במקרה הספציפי תפסו כ60 שניות (היה אופציה גם ל30, 120 וכו׳) ואת פרך הזמן הזה נרמלו לטווח בין 1 ל1500
ולפי זה בעצם בנו את ה FlowPic ציר הx היה לפי זמן ההגעה המנורמל, ציר ה yהיה לפי גודל החבילה. (עד 1500 בתים) ולכן נקבל איזשהי תמונה שתגדיר את הזרימה של החבילות האלו בפרק הזמן המסוים שבדקנו.
כאן בעצם במה שם עשו אין שום מאפיינים חדשים שהכניסו אך שיטות הניתוח הן אלו החדשניות, האופן שבו מתייחסים לנתוני התעבורה כאל תמונה בעצם ואז משתמשים בטכניקות שהן כבר מבוססות על תמונות כמו רשתות נוירונים.
תפיסת המידע יחסית פשוטה כי אין צורך בחילוץ מידע מהחבילות רק את הגודל שלה ומתי הגיעה
ומערכת לומדת לבד את המאפיינים החשובים לגבי כל המידע דרך מערכת הCNN

3. בניסוי זה הם הציגו מספר תוצאות לגבי דיוק סיווג 3 סוגי התעבורות השונות (VPN, Non VPN, Tor) , בנוסף הם גם הראו תוצאות על סיווג אפליקציות לא מוכרות, שזה בעצם ניסוי חדשני על פי המאמר משהו שלא עשו בעבר, הם הראו את התוצאות במספר דרכים, תוצאות של בדיקת התיוק הספציפי הזה אל מול כל הקלאסים האחרים, דיוק על לתייק כל קלאס נכון, ואת התוצאות על אפליקציות לא מוכרות לדוגמא עם כל מיני סוגים שונים של וידאו, נראה תחילה על כל תוצאה ונסביר ולבסוף נסביר על המסקנות שיש מהתוצאות (גם באופן כללי וגם על מסקנות בהתחשב לתוצאות עבר שהיו:

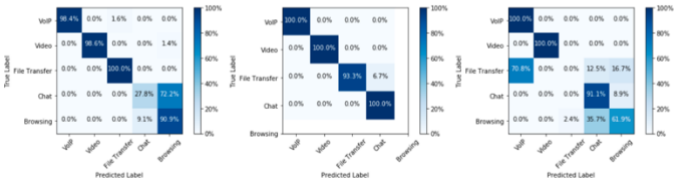
תחילה נדבר על התיוק באופן כללי על כל קלאס בפני עצמו,

Problem	FlowPic Acc. (%)	Best Previous Result	Remark
Non-VPN Traffic Categorization	85.0	84.0 % Pr., Gil *et al.* [15]	Different categories. [15] used unbalanced dataset
VPN Traffic Categorization	98.4	98.6 % Acc., Wang *et al.* [7]	[7] Classify raw packets data. Not including browsing category
Tor Traffic Categorization	67.8	84.3 % Pr., Gil *et al.* [15]	Different categories. [15] used unbalanced dataset

כאן בטבלה הראשונה ניתן לראות שאחוזי הדיוק על Non VPN היו כ85%, אחוזי הדיוק על VPN היו עם אחוזי דיוק של כ98.4% ובסוף ניתן לראות שהתוצאות על Tor היו 67.8% (משמעותית יותר נמוך מהניסויים האחרים ונסביר למה בפסקה האחרונה)

כעת נדבר על הסיווג של קלאס אל מול שאר הקלאסים
כאן ניתן לראות שיכולת לזהות עבור כל אחד מהקלאסים, על כל אחד מהסוגים היה בהצלחה יחסית גדולה, שכן ניתן לראות גם שעל הדפדפן לא השתמשו בVPN ולכן אין עליו תוצאות
כן גם כאן ניתן לראות שהתוצאות עבור Tor יצאו נמוכות יותר בהתחשב לשאר התוצאות, אבל כך התוצאות היו כפי שניתן לראות יותר מדויקות.

Class	Accuracy (%)			
VoIP	Training/Test	Non-VPN	VPN	Tor
	Non-VPN	**99.6**	99.4	48.2
	VPN	95.8	**99.9**	58.1
	Tor	52.1	35.8	**93.3**
Video	Training/Test	Non-VPN	VPN	Tor
	Non-VPN	**99.9**	98.8	83.8
	VPN	54.0	**99.9**	57.8
	Tor	55.3	86.1	**99.9**
File Transfer	Training/Test	Non-VPN	VPN	Tor
	Non-VPN	**98.8**	79.9	60.6
	VPN	65.1	**99.9**	54.5
	Tor	63.1	35.8	**55.8**
Chat	Training/Test	Non-VPN	VPN	Tor
	Non-VPN	**96.2**	78.9	70.3
	VPN	71.7	**99.2**	69.4
	Tor	85.8	93.1	**89.0**
Browsing	Training/Test	Non-VPN	VPN	Tor
	Non-VPN	**90.6**	-	57.2
	VPN	-	-	-
	Tor	76.1	-	**90.6**



(a) Non-VPN (b) VPN (c) Tor

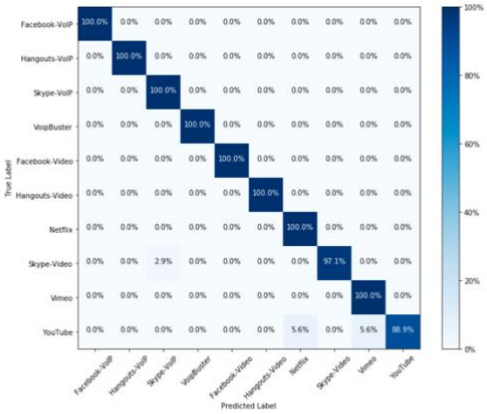


Figure 5: A confusion matrix of the VoIP and video applications identification problem.

נלך כעת לתוצאות האחרונות שמראות עבור בדיקה על אפליקציות שאינן נבדקו בעבר, כאן בדקו על וידאו מעוד אפליקציות שונות שאינן נראו בקבוצת שעליה אימנו את המודל. (גרף 5)
גם כאן ניתן לראות שאחוזי ההצלחה עבור כולם היו מאוד מאוד גבוהים ברובם 100%, בחלקם בחלק של יוטיוב היה קצת פחות מ90% הצלחה אך בסך הכל נראה שמאוד

הצליחה גם על אפליקציות שהמודל אינו ראה בעבר.

התחיל קודם כל מהסבר למה אחוזי הדיוק יצאו הרבה יותר נמוכים עבור המודל הספציפי כאן בחלק מהמקרים, זאת מכיוון שכאן דאגו לקחת מידע שהוא מפוזר יותר ללא overfitting כדי שהתוצאות יהיו אמינות, מה שלפי הנאמר לא בהכרח קרה בניסויים הקודמים.
ומסקנות מתוצאות הניסוי הן שהשיטה עובדת יחסית טוב מאוד עבור VPN ניתן להסיק שהמאפיינים של תעבורת Tor שונים מאוד מהמאפיינים של שאר התעבורות וכנראה בגלל זה התוצאות עבורה יצאו מאוד נמוכות יחסית 67.8%, ניתן לראות שהמודל מתמודד טוב מאוד עם אפליקציות שאינו ראה בעבר בשלב האימון מה שמצביע על למידה אמיתית ועמוקה על שלושת קטגוריות תעבורה אלו, בנוסף ששיטה זו יעילה כי היא מביאה תוצאות יחסית טובות ואמינות, עם אפשרות לבדוק על אפליקציות חדשות וזאת בכלל מבלי להיכנס אל תוך החבילות עצמן אלא רק על הגודל והזמן שלהן.

Early Traffic Classification With Encrypted ClientHello: A Multi-Country Study
מאמר –

מטרה מרכזית של המאמר:

מטרת המאמר המחקרי הזה היא פיתוח שיטה חדשה לסיווג מוקדם של תעבורת רשת מוצפנת, בהסתכלות יותר עמוקה על Encrypted Client Hello (ECH) .

השיטה המסורתית לסיווג תעבורה נתקלות באתגרים משמעותיים בגלל ההצפנה הקיימת, מה שמוביל לשיעורי דיוק נמוכים, המחקר מתעסק באלגוריתמים הקיימים ומדגיש את המגבלות והחסרונות שלהם בסיווג מדויק של תעבורה מוצפנת.

כדי להתמודד עם אתגרים אלה, החוקרים פיתחו מסווג היברידי חדש(hybrid Random Forest hRFTC- Traffic Classifier) שמשתמש בתעבורה לא מוצפנת של לחיצת הידיים של ה TLS(הוא פרוטוקול אבטחה שמשמש להצפנת תקשורת באינטרנט), מאפייני סדרות זמן וגודל חבילות וסטטיסטיקה. ה- hRFTC שואף להשיג ביצועים טובים יותר משל המסווגים המתקדמים ביותר כיום על ידי שילוב מערכי נתונים מגוונים והטרוגניים שנאספו ממיקומים גיאוגרפים שונים. בנוסף, המחקר מדגיש אתת החשיבות של אימון אלגוריתמי סיווג על נתונים מאותו אזור גיאוגרפי בו הם יופעלו, מכיוון שדפוסי התעבורה יכולים להשתנות משמעותיים באזורים שונים. באופן כללי, המחקר תורם להבנה טובה יותר של סיווג תעבורה מוצפנת ומציע פתרונות מעשיים לשיפר הדיוק והיעילות של סיווג תעבורה בנוכחות הצפנה.

המאפיינים הבסיסיים:

1. נתוני המטא של לחיצת היד TLS: זה כולל אלמנטים מלחיצת היד של TLS, כמו החבילות הראשוניות שמוחלפות במהלך הקמת החיבור, המספקות תובנות למרות ההצפנה. השתמשו ב IP/TCP ובמשתנים לא מוצפנים.

2. סטטיסטיקת גודל חבילות: גודל החבילות בשלבים המוקדמים של הזרימה עוזר להבחין בין סוגים שונים של תעבורה על בסיס מאפייני יישום טיפוסיים. בשמם איך שמופעים במאמר: DL PSs Cumulative Sum, DL PSs Sorted Unique, UL PSs Std, UL PSs Cumulative Sum ועוד הרבה מאפיינים שמחוברים ל PS.

3.מאפייני זמן בין חבילות: סכום כל זמני הביניים בין חבילות בכיוון ההורדה(DL IPTs Sum), סכום כל זמני הביניים בין חבילות בכיוון העלאה(UL IPTs Sum), האחוזון ה 25 של זמני הביניים בהלאעה(UL IPTs 25th percentile), החציון של זמני הביניים בהעלאה(UL IPTs 50th percentile), הזמן המקסימלי בין חבילות בהעלאה (UL IPTs max)

4. מאפיינים מבוססי זרימה: מדדים הנגזרים מזרימת החבילות, כמו זמני הגעה בין חבילות וסך הבתים שהועברו, ממוצע, סטיית תקן ושונות, מינימום ומקסימום, וחילוק לאחוזים של 25,50,75. Ps pattern- תבנית גדלי חבילות, ps unique- ווקטור של גדלי חבילות ייחודיים, היסטוגרמה של גדלי חבילות וסטטיסטיקות זמן בין חבילות (IPT).

6. Cipher suite– זה סוג של חבילת אבטחה שמגדירה איך המידע יוצפן ויאובטח בתקשורת בין דפדפן ללקוח, משתמש ב: TLS Cipher Suite length, supported cipher suites, Key share extensions

And Supported versions

מאפייני תעבורה חדשניים:

1. מאפייני זרימה סטטיסטיים חדשים: המחקר מדגיש מאפיינים סטטיסטיים חדשים המחולצים מזרימת התעבורה לפני שמועבר מידע יישומי כלשהו, שיכולים לכלול דפוסים או התפלגויות שלא היו בשימוש קודם בהקשר של eTC. גישה היברידית שמשלבת מאפייני זרימה ו payload.

2. וקטור מאפיינים היברידי: הגישה שננקטה במאמר לשלב מאפיינים מבוססי זרימה ומאפיינים מבוססי חבילות לווקטור מאפיינים היברידי יחיד מודגשת כשיטה חדשנית. שילוב מאפייני זרימה ומבוססי חבילות, העשרת נתוני סיווג ושיפור דיוק הסיווג דרך היברידיזציה.

3. הכללה על פני מאגרי נתונים מגוונים: איסוף מאגר נתוני תעבורת TLS ממדינות רבות, המציג סוגים שונים של תעבורה עם דרישות QoS שונות, הוא היבט חדשני. מאמץ זה לאסוף נתונים מגוונים עוזר להעריך את חוסן המסווגים על פני דפוסים והתנהגויות טיפוסיות במיקומים גיאוגרפיים שונים.

5. QUIC- שימוש ב connection id לזיהוי הזרימה במקום tuple, סוג ה QUIC, מספר חבילה, crypto frames content והחבילה המקורית. בנוסף, שינוי מבנה ה QUIC עבור ה recomposed payload. התאמת האלגוריתם עבור RF-RB והתמודדות עם מאפייני ה TCP/TLS

6. recomposed payload- ארגון מחדש של בתים לא מוצפנים לווקטורים בגודל קבוע, שיפור בשיטת חילוץ המאפיינים מהתרחישים של תעבורה מוצפנת ויישום ייחודי עבור RANDOM FOREST.

תוצאות עיקריות:

שיפור דיוק הסיווג- ביצועים גבוהים גם עם מאגרי נתונים קשים של תעבורה מוצפנת, התוצאות מראות שגודל החבילות תורם יותר מ 50% לדיוק, מה שמדגיש את החשיבות של הגודל חבילות גם בהקשרים מוצפנים.

יכולת הכללה- אלגוריתם ה hRFTC כשמקטינים את מאגר האימון מ 70% ל 10% אז הוא עמיד ויורד ממש בקטנה במדד הF-score זה מראה על יכולת הכללה טובה גם ממאגר אימון קטן מאוד.

השפעת ריפוד QUIC – הוא מסבך את סיווג התעבורה כי כל חבילות לחיצות הידיים של הפרוטוקול הן באותו אורך. האחידות מקשה על איכות הסיוג, במיוחד בסיווג תעבורת QUIC לדברים ספציפיים.

חשיבות המאפיינים- למרות שה ECH מנסה להצפין כמה שיותר הודעות TLS מאפייני ה payload עדיין שומרים על חשיבות של 30% מתוך הסיווג הכולל. זה מראה שגם תעבורה מוצפנת יכולה לספר תובנות חשובות לסיווג.

תובנות מהתוצאות:

ניתן לראות את יעילות השיטה ההיברידית ששילוב מאפייני מבוססי חבילות וזרימה ממוציא תוצאות יותר טובות בסיווג בזמן אמת ויעיל במיוחד עבור תעבורה מוצפנת. בנוסף, זה שניתן ליישם את זה בעולם האמיתי כלומר שהתאמה טובה של אלגוריתמים קיימים כמו RB-RF כדי לסווג תעבורת QUIK ומדגיש את חשיבות ההתקדמות בסיווג תעבורה ככל שיותר שירותים עוברים ל QUIK.

תובנה נוספת היא אי תלות במאגרי נתונים גדולים, כמו שאמרנו בתוצאות העיקריות היכולת לשמור על ביצועים טובים גם אם אימון מצומצם זה נותן יתרון כאשר אין לנו הרבה נתונים לעבוד איתם.

התוצאות משקפות בנוסף את הקושי שההצפנה שמה כאשר מנסים לסווג תעבורה, וזה מראה לנו שצריך לפתח עוד אסטרטגיות חדשות שיתמודדו עם זה

באופן כללי, התוצאות מדגישות את השינויים בשיטות סיווג תעבורת רשת כאשר קיימת הצפנה ופרוטוקולים חדשים, ומראות גם על התקדמות וגם על האתגרים שמוצבים בפנינו, נצרף תוצאות וויזואליות מהמחקר:

בטבלה זו ניתן לראות כי הם צמצמו את כמות המדינות כלומר הקבוצת אימון שלנו ירדה אבל הם כן ניסו שיהיה כמה שיותר מדינות עם דרישות QOS (מושג שמתייחס ליכולת לתעדף סוגי שונים של תעבורה כדי לספק להם רמת שירות מובטחת) שונות. ובנוסף זה גם מראה את ההבדל בין גישות שונות כאשר כמות הדאטה יורדת.

TABLE 4. Traffic volume distribution over geographic locations.

Country	Cities	Number of Flows
Germany	Munich	121,936
Kazakhstan	Aktau, Aktobe	17,228
Russia	Moscow, Dolgoprudny, Zelenograd	234,335
Spain	Girona	95,154
Turkey	Antalya, Kemer	160,131
USA	Miami	43,850

TABLE 11. Full dataset per class F-score for different classifiers.

Class	F-score [%]						
	Hybrid Classifiers			Flow-based Classifier	Packet-based Classifiers		
	hRFTC [proposed]	**UW** [35]	**hC4.5** [34]	**CESNET** [63]	**RB-RF** [24]	**MATEC** [33]	**BGRUA** [32]
BA-AppleMusic	**92.1**	89.5	80.2	89.2	25.5	13.1	14.5
BA-SoundCloud	**99.6**	98.9	97.8	98.7	84.4	81.8	82.0
BA-Spotify	**93.6**	90.8	89.0	88.5	16.3	0.0	3.6
BA-VkMusic	**95.7**	89.7	88.5	91.8	2.6	2.1	3.2
BA-YandexMusic	**98.5**	93.2	93.7	92.5	1.8	0.2	0.1
LV-Facebook	**100.0**	99.7	99.8	99.8	**100.0**	**100.0**	**100.0**
LV-YouTube	**100.0**	**100.0**	99.9	**100.0**	**100.0**	99.0	98.4
SBV-Instagram	**89.7**	74.7	76.5	78.8	10.0	6.3	6.4
SBV-TikTok	**93.3**	81.8	81.8	76.3	38.3	34.3	34.5
SBV-VkClips	**95.7**	94.0	91.3	92.4	53.2	37.7	46.0
SBV-YouTube	**98.2**	96.6	94.7	96.4	1.1	0.2	0.2
BV-Facebook	**87.7**	78.2	79.7	77.6	5.6	3.2	3.8
BV-Kinopoisk	**94.1**	84.1	85.8	89.8	5.4	4.0	4.1
BV-Netflix	**98.5**	97.2	95.2	93.7	50.7	52.3	56.1
BV-PrimeVideo	**91.3**	86.7	84.1	84.7	32.5	24.7	26.8
BV-Vimeo	**94.8**	90.5	90.2	81.4	72.0	19.5	68.6
BV-VkVideo	**88.6**	80.5	80.4	79.7	10.5	0.0	0.1
BV-YouTube	**85.9**	84.3	77.0	78.5	22.3	19.6	20.2
Web (known)	**99.7**	99.5	99.4	99.4	98.0	98.0	98.0
Macro-F-score (average)	**94.6**	89.9	88.7	88.9	38.4	31.4	35.1

LV is Live Video, (S)BV is (Short) Buffered Video, and BA is Buffered Audio.

כאן ניתן לראות כי זה מראה השוואת על בסיס סיווג בין אלגוריתמים שונים, ובאמת ניתן לראות בטבלה ש hRTFC מציגה את התוצאות הטובות ביותר.

בטבלה זו אנו רואים דירוג של חשיבות המאפיינים שהוצגו גם לעיל עם הסברים, והטבלה מדגימה לנו שגודל החבילה משפיע על הסיווג.

TABLE 12. hRFTC: the Gini-impurity-based feature importance normalized by the maximal observed value.

Rank	Feature	Impurity-based Feature Importance
1	CH Cipher Suites length	1.00
2	DL PSs Cumulative Sum	0.62
3	DL PSs Sorted Unique #1	0.50
4	UL PSs Std	0.46
5	UL PSs Cumulative Sum	0.45
6	UL PSs #2	0.44
7	DL IPTs Sum	0.43
8	DL PSs 25th percentile	0.43
9	DL PSs average	0.42
10	UL PSs 75th percentile	0.41
11	SH Cipher Suite	0.40
12	DL PSs Std	0.40
13	UL PSs Sorted Unique #2	0.40
14	UL PSs max	0.38
15	DL PSs 50th percentile	0.37
16	UL PSs average	0.36
17	DL PSs 75th percentile	0.33
18	UL IPTs Sum	0.28
19	UL PS #1 (CH length)	0.27
20	DL PS #2	0.26
21	UL IPTs min	0.26
22	UL PS 750-1000B freq	0.25
23	UL PSs Sorted Unique #3	0.25
24	CH Extensions Length	0.24
25	SH Extension Type #2	0.23
26	UL IPTs 25th percentile	0.23
27	UL IPTs 50th percentile	0.22
28	SH Extension Type #1	0.22
29	DL PS #3	0.21
30	UL IPTs max	0.2

בטבלה זו ניתן לראות שמאפייני ה payload שומרים על כ 30%

TABLE 13. **hRFTC: Sum of GI values over payload, PS, and IPT features normalized by the sum of all GI values.**

Payload		IPT	PS			
CH	SH	IPT Stats	PS Hist	PSs Unique	PS Stats	PS Pattern
0.18	0.09	0.16	0.05	0.11	0.16	0.26
0.27		0.16	0.57			

TABLE 14. **TC quality depending on training locations.**

Test Country	Share in Dataset	Training Country	Classifier Macro F-score [%]		
			hRFTC	hC4.5	UW
Germany	18.8%	Others	38.4	26.9	19.5
Kazakhstan	3.0%	Others	57.3	32.3	27.5
Russia	29.2%	Others	49.8	35.6	20.9
Spain	16.3%	Others	38.5	34.4	12.6
Turkey	25.2%	Others	35.1	26.0	16.4
USA	7.5%	Others	49.2	41.4	21.3

טבלה שמראה לנו ביצועי סיווג במקומות שונים בעולם.

זה פשוט מציג לנו בצורה וויזואלית מדגים את יכולת ה hRFTC-לשמור על ביצועים גם עם מאגר אימון מצומצם ו מציג השוואה בין גישות שונות כאשר מקטינים את כמות נתוני האימון.

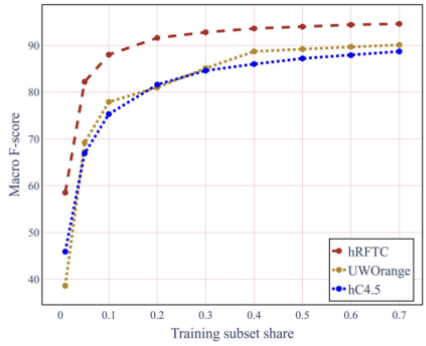


FIGURE 4. **F-score depending on the training subset share.**

מאמר: :
Analyzing HTTPS Encrypted Traffic to Identify User's Operating System, Browser and Application:

1. למאמר זה יש 2 תרומות עיקריות:
תחילה והעיקרית שבהן זה להראות ולתת דרכים איך אפשר לזהות דרך תעבורת HTTPS את האפליקציה, מערכת ההפעלה, והדפדפן ממנו בוצעה התנועה (גם אליו וגם ממנו), מראים גם את החוזי ההצלחה הגבוהים מה שמראה שלמרות הצפנות SSL עדיין מצליחים במעל 96 אחוזים את תעבורת הרשת.
בנוסף הוא מראה את הדרכים כיצד אפשר לזהות את התעבורה של המחשב ואיך לזהות את האפליקציה, הדפדפן ומערכת ההפעלה (הכל על המחשב, עוד לא פיתחו גם לנייד).
בנוסף הוא תורם במאמר בכך שהוא צירף גם דאטאסט של מעל 20000 זיהויים שהם מסומנים עם תוויות , וצירפו מי מערכות ההפעלה, הדפדפנים והאפליקציות הכלולות בדאטאסט ושאותן גם ניסו לזהות.

2. תחילה נסביר כל מיני מושגים על מנת שהמאפיינים יהיו ברורים, מכיוון שלרוב המאפיינים אין צורך שהסבר נקודתי, אלא הסבר של מספר נקודות.
דבר ראשון נסביר על חבילות קדימה, חבילות אחורה.
חבילות קדימה הינן חבילות שנשלחות מהלקוח אל השרת, החבילות אחורה הינן חבילות שנשלחות מהשרת חזרה אל הלקוח.
ונשים לב שהקטגוריות של המאפיינים הבסיסיים מתייחסים למאפייני זמן, מאפייני TTL, ומאפיינים בסיסיים כמו מספר חבילות מספר בתים כולל, גודל, וכו.
הקטגוריות של המאפיינים החדשים מתייחסים לדברים אחרים, כגון הצפנה (SSL), מאפייני TCP, מאפייני פרצים, שבעצם זה ממש כפי השם שלו, כאשר יש פרץ של תעבורה כלומר בזמן קצר עם הרבה תעבורה, מהגרף שהיה במאמר(גרף 3) ניתן לראות כי מדברים על תפוקת הפרץ והתזמון בין פרצים, נוסף על כך במאפיינים החדשים הסתכלו גם על חבילות keep alive שנועדו לשמור על החיבור פעיל שלא ייסגר.

המאפיינים הבסיסיים:

מאפיינים בסיסיים של חבילות:
מספר חבילות קדימה, סך בתים קדימה, הפרש מינימלי בזמני הגעה קדימה, ממוצע חבילות קדימה, סטיית תקן חבילות קדימה, מספר חבילות אחורה, סך בתים אחורה, ממוצע חבילות אחורה, סטיית תקן חבילות אחורה, סך כל החבילות, גודל חבילה מינימלי, גודל חבילה מקסימלי, גודל חבילה ממוצע, שונות גודל חבילה

מאפייני זמן:
הפרש מקסימלי בזמני הגעה קדימה, הפרש ממוצע בזמני הגעה קדימה, סטיית תקן של הפרשי זמני הגעה קדימה, הפרש מינימלי בזמני הגעה אחורה, הפרש מקסימלי בזמני הגעה אחורה, הפרש ממוצע בזמני הגעה אחורה, סטיית תקן של הפרשי זמני הגעה אחורה.

מאפייני TTL:
ערך ממוצע TTL קדימה, חבילה מינימלית קדימה, חבילה מינימלית אחורה, חבילה מקסימלית קדימה, חבילה מקסימלית אחורה.

המאפיינים החדשים:

מאפייני TCP:
גודל חלון TCP התחלתי, פקטור שינוי גודל חלון, גודל מקטע TCP מקסימלי

מאפייני דחיסה והצפנה:
מספר שיטות דחיסת SSL,מספר הרחבות SSL,מספר שיטות הצפנה SSL,אורך מזהה סשן SSL ,גרסת SSL קדימה

מאפייני פרצים:
תפוקה מקסימלית של פרץ קדימה, תפוקה ממוצעת של פרצים אחורה, תפוקה מקסימלית של

פרצים אחורה, תפוקה מינימלית של פרץ אחורה, סטיית תקן תפוקת פרץ אחורה, מספר פרצים קדימה, מספר פרצים אחורה, תפוקה מינימלית של פרץ קדימה, תפוקה ממוצעת של פרצים קדימה, סטיית תקן תפוקת פרץ קדימה, הפרש ממוצע בין זמני הגעת פרצים אחורה, הפרש מינימלי בין זמני הגעת פרצים אחורה, הפרש מקסימלי בין זמני הגעת פרצים אחורה, סטיית תקן של הפרשי זמני הגעת פרצים אחורה, הפרש ממוצע בין זמני הגעת פרצים קדימה, הפרש מינימלי בין זמני הגעת פרצים קדימה, הפרש מקסימלי בין זמני הגעת פרצים קדימה, סטיית תקן של הפרשי זמני הגעת פרצים קדימה,

<u>התייחסות לחבילות Keep Alive:</u>

מספר חבילות Keep Alive

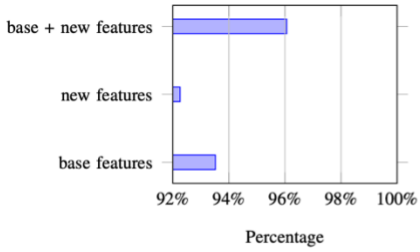
אחת הסיבות שהמאפיינים החדשים מוסיפים לנו ומשפרים את התוצאות הן מכיוון, שבעזרת פרצים אפשר לזהות דפדפנים שונים מכיוון שדפדפנים שונים משדרים בפרצים שונים, וצורת הניהול של התעבורה משתנה בין מערכות הפעלה שונות.

3. התוצאות העיקריות הן לכך שבלי המאפיינים החדשים יש אחוזי דיוק של 93.52 אחוז ועם המאפיינים החדשים הגיעו לאחוזי דיוק של 96.06 אחוז

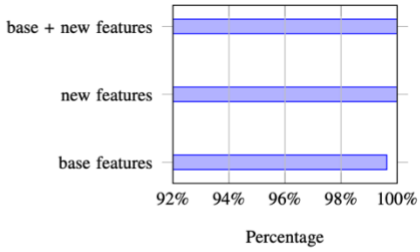
אחת מהתוצאות מראה גם שמה שבעיקר מוריד את אחוזי ההצלחה זה אלו שלא היה אפשר לזהות כלל את הערכים של מה הם באמת (unknown) ולכן יכול להיות שזה הצליח כן לסווג אותם נכון ואנחנו לא יודעים ולכן יכול להיות שאחוזי הדיוק אפילו יותר גדולים

התובנות העיקריות הן: ניתן לזהות פרטים על המשתמש גם אם התעבורה שלו מוצפנת בעזרת כל מיני דרכים, שהמאפיינים החדשים שהצליחו לנתח בעזרתם משפרים את הדיוק של הניסוי, ושסיווג מערכת ההפעלה יותר מדויק מסיווג האפליקציה והדפדפן (יכול להיות שזה גם בגלל שיש מהם כמות יותר קטנה)

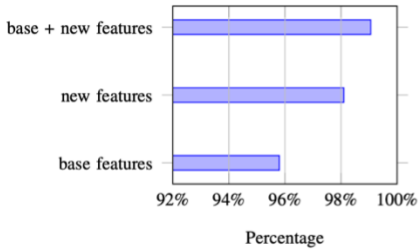
בתוצאות כאן מטה אפשר לראות בפלט a שאחוזי הדיוק עם הפיצ'רים החדשים והישנים היו הרבה יותר מוצלחים בכל הרמות, וגם את רמת הדיוק לגבי כל אחד ספציפית, התוצאות על מערכת ההפעלה (b) התוצאות על הדפדפן (c) והתוצאות על האפליקציה (d).



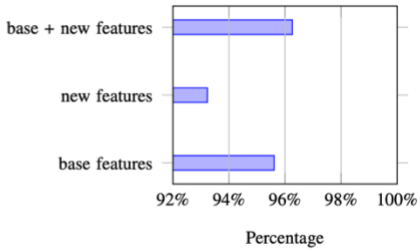
(a) Tuple Accuracy Results



(b) OS Accuracy Results



(c) Browser Accuracy Results



(d) Application Accuracy Results

ניתן לראות גם כאן בתוצאות האלו שמראות את הדיוק ספציפית על כל פרמטר מבין מה שמדדנו שבאמת יכול להיות שמה שהאלגוריתם שלהם תפס כunknown יכל להיות Facebook באמת ששם יצאה הירידה המשמעותית בדיוק באפליקציה בשאר הדברים נראה שהוא חזה די טוב.

Predicted labels

Real labels

	Windows	Ubuntu	OSX
Windows	1	0	0
Ubuntu	0	1	0
OSX	0	0	1

(b) OS Confusion Matrix

Predicted labels

Real labels

	Chrome	Firefox	IExolorer	Safari	Non-Browser
Chrome	.97	.02	0	0	0
Firefox	.01	.98	0	0	0
IExolorer	0	0	1	0	0
Safari	.01	0	0	.99	0
Non-Browser	.03	0	0	0	.96

(c) Browser Confusion Matrix

Predicted labels

Real labels

	Dropbox	Facebook	Google-background	Microsoft-Background	Teamviewer	Twitter	Youtube	Unknown
Dropbox	.98	0	.02	0	0	0	0	0
Facebook	0	.62	.04	0	0	.04	0	.29
Google-background	0	0	.95	0	0	.01	.01	.03
Microsoft-Background	0	0	0	.96	0	0	0	.04
Teamviewer	0	0	0	0	1	0	0	0
Twitter	0	0	0	0	0	.98	0	.01
Youtube	0	0	.03	0	0	.02	.93	.01
Unknown	0	.02	.04	.01	0	.05	.01	.86

(d) Application Confusion Matrix

חלק 3 – ניתוח תעבורה ויצירת מודלי חיזוי לסיוע ניתוח התעבורה:

תחילה בחלק זה נרצה להסביר על הדרישות הקיימות להרצת הקוד, והסברים על כל הקבצים שלנו נאמר קודם כל שבקובץ הreadme המצורף לפרויקט יש את כל הדרישות לפרויקט שלנו, ובנוסף הסבר מפורט על כל קובץ וקובץ בפרויקט, החל מקבצי הפרויקט עצמו, ועד לקבצי הקלטות, כך שאם מישהו רוצה להשתמש בקבצים אחרים, כל שעליו הוא להתאים את עצמו לתעבורה הקיימת בקובץ, לסוג הקובץ, ופשוט לשים את הקובץ הנ"ל עם השם המתאים בתיקייה הנדרשת של הפרויקט.

מכיוון שאין אנו רוצים לרשום דברים פעמים נסביר איך להתמצא וליצור את מה שברצונכם לעשות.
בראש קובץ הרידמי, קיים מקרא עם קישורים שממנו אפשר להגיע אל קובץ הרידמי הנתון לאותו חלק.
תחילה נלחץ על Project Structure, ולאחר מכן בעצם נוכל לראות סוגים שונים של קבצים PDF Files – שמתאר בתוכו את קבצי המאמרים אותם ניתחנו, קובץ דרישות הפרויקט, והקובץ הנ"ל.
Jupyter Notebooks – כאן קיימים לנו 2 קבצים, מחברות אלה מאפשרות לנו ליצור מאין מסמך שבו אפשר לפרט ולהסביר על הקוד הנראה מולנו, ולהוציא את כל הגרפים בצורה מסודרת בתוך הקובץ אותו אנחנו מסבירים, וכך אפשר גם להריץ כל תא בפני עצמו (למי שלא מכיר את מחברות אלה מומלץ לקרוא עליהם לפני לינק להסבר , ולשים לב שאם בתא מסוים יש תלותיות מתא אחר אז חייב להריץ את התאים שבהם הוא תלוי מלפני)
הקובץ הראשון זהו המחברת שבה בעצם ניתחנו את תעבורת הרשת שלנו והראינו עבור כל תעבורת רשת סוגים שונים של גרפים, ולאחר מכן ניתחנו את המסקנות שלנו מתוך כך
במחברת השנייה קיימים לנו מודלי החיזוי השונים בהם השתמשנו על מנת לחזות את תעבורות הרשת.
על 2 מחברות אלה ניתן לראות הסבר נוסף, בנוסף למחברות גם כאן בקובץ הPDF החל מהעמוד הבא.
Data Files – בחלק הזה נוכל לראות את כל קבצי המידע בהם השתמשנו, בחלק זה ניתן לראות שהם מחולקים לפי סוג התעבורה שנותחה. את סוג הקובץ אפשר לראות לאחר ה. (קיימים 2 סוגי קבצים .pcap, .csv), את הפירוט מה יש בכל קובץ ניתן לראות בתיאור הנמצא צמוד לשם הקובץ.
נבקש שבעת החלפת הקבצים לקבצים אחרים יש להיצמד לניתוח התעבורה, לצירוף הקבצים הנ"ל עם השמות הללו.
בנוסף בחלק זה ניתן לראות גם את הקובץ שבו השתמשנו עבור למידת המכונה שלנו, את הקובץ הזה לא ניתן להחליף שכן הוא מנתח ומשתמש בנתונים האלו שנמצאו בדאטה – סט שלקחנו מאתר Kaggle ושאושר לשימוש ע"י פרופ' עמית דביר. (קישור לדאטה הספציפי), נבקש גם לשים לב, שבגלל שהקובץ הינו מאוד גדול, יש להוריד אותו מהאתר ולמקם אותו ביחד עם כל הפרויקט באותה תיקייה.

נרצה גם להסב את תשומת הלב לדברים הבאים:
יש לעקוב אחר הוראות ההרצה וההתקנה לפי קובץ הרידמי המצורף, יש להיצמד לדרישות המדויקות של פרויקט זה ולהתקין את הגרסאות אותן דרשנו (אפשר להתקין בלחיצת כפתור בקובץ הרידמי)
נוסף על כך מכיוון שבשלב מודלי החיזוי רצינו ליצור מודל שינתח בצורה מלאה את אותו הדאטה שלקחנו, והשתמשנו בספריות כמו TensorFlow שדורשת CPU גבוה ומעבדים חזקים, יכולה להיות בעיה של הרצה על מחשבים מסוימים, וגם במחשבים שיוכלו להריץ הקוד כולל בתוכו בדיקות מרובות וניתוח של הדאטה בצורה מלאה ומשתמש במספר רב של מודלים, ועבור כל אחד בוחר את הארגומנטים הנכונים לו לפי המודל בעת ההרצה, ולכן כל אלו גורמים לזמן הרצה ארוך של קובץ זה.
ולכן אנחנו מציינים זאת כאן כדי שדבר זה יהיה כחלק מהידע שלכם על הפרויקט, והבנה שמכונה וירטואלית אינה מספיקה בשביל להריץ את המודלים הללו כי אין באפשרותה להקצות מספיק משאבים לכך.
לכן הקוד ירוץ על הסביבות, Windows, Linux במידה ויהיו להם מוקצים משאבים מספקים.

ניתוח תעבורה ויצירת גרפים

תחילה נרצה להסביר על הדאטה שאותה הקלטנו בעזרת גרפים ולהסביר אותם, בשביל זה בעצם כדי שהדבר יקרה בצורה מסודרת, עבדנו עם מחברות Jupiter Notebooks עשינו את זה על מנת שנוכל להסביר בכל גרף מה אנחנו רואים ולהסביר את כל המידע של הגרף

בהתחלה הסברנו על הדברים שאותם רצינו לבדוק, כלומר סוגים של כל מיני תעבורות רשת שונות לפי שימושים שונים ברשת:

לפי שיחות וידאו: ניתחנו תעבורה של שיחות זום ושיחות של גוגל פגישות

לפי וידאו סטרימינג: ניתחנו תעבורה של בYouTube

לפי אודיו סטרימינג: ניתחנו תעבורה של Spotify

לפי גלישה ברשת: ניתחנו תעבורה של גלישה בדפדפנים מסוגים שונים (כרום, אדג׳)

בכל אחד מתעבורות אלו הוצאנו מספר גרפים שונים שבעזרתם רצינו לנתח את תעבורת הרשת.

נשים לב תחילה שהמטרה שלנו הייתה להבין איך המידע של תעבורת הרשת מתנהג כאשר התעבורה מוצפנת, בעצם אין לנו דרך להגיע אל הדאטה עצמה כי היא מוצפנת לכן כלל הגרפים שלנו התייחסו למספר גורמים, הזמן בו החבילה הגיעה, גודל החבילה, IP מקור, IP יעד, ופרוטוקול.

מכיוון שרוב הדאטה שלנו מתייחס בעצם רק לכמות הביטים ולזמני ההגעה שלהם נוכל לתאר אותם כך שבעצם אנחנו התוקפים ואלו דברים שיהיו לנו גישה אליהם, כי בתור תוקף שתופס חבילות נוכל לראות את התדירות שלהם, ואת הגדלים ולפי זה בעצם ניתחנו את התעבורה, חלק מהגרפים מתייחסים גם לפרוטוקול ול-IP שבעצם אלו דברים שאם רק הדאטה ברמה הנמוכה מוצפנת יהיו לנו גישה אליהם, ומעבר לכך לIP כן יש גישה כפי שלנתבים עצמם יש גישה על מנת שיוכלו לנתב את החבילות.

כעת נרצה להסביר את ששת הגרפים השונים אותם יצרנו עבור כל אחד מהתעבורות.

גרף ראשון - 'Distribution of Packet Lengths with <X> bins' גרף היסטוגרמה

בעצם בגרף זה אנחנו מראים את התדירות של הפקטות שהגיעו בחלוקה לפי גודל שלהם, על זה בנינו גרף קווי שמראה בערך את ההתפלגות של הגרף לפי תדירות. כלומר- הגרף מראה את התדירות של פקטות מגודל מסוים, ובנוסף רשום בסוף על הגודל הכללי של כלל הפקטות יחדיו וכמה פקטות סה״כ נתפסו.

גרף שני- 'Protocol Distribution' – גרף עוגה

בעצם בגרף זה אנחנו יכולים לראות את התפלגות הפרוטוקולים בחבילות שאותם קיבלנו באותה תעבורת רשת, דבר זה מסייע לנו לראות שכל סוג תעבורה משתמש בפרוטוקולים אחרים בכמות אחרת, דבר אשר יסייע לנו לזהות את ההבדלים בין תעבורות הרשת השונות.

גרף שלישי + רביעי- תחילה נשים לב ש2 הגרפים הללו קשורים אחד לשני ולאחר מכן נסביר על כל גרף בנפרד, בגרף אחד נראה מה קורה בכל שנייה נתונה לתוך ההקלטה של הדאטה שלנו, ובגרף שאחריו נראה במצטבר מה קורה עד אותה השנייה.

גרף שלישי- 'Histogram of Packet Arrivals Over Time' – גרף היסטוגרמה.

בגרף זה בעצם אנחנו מראים כמה חבילות הגיעו באותה שנייה ספציפית בתעבורה הזו, ניתן לראות שבחלק מהגרפים זה די קבוע ורציף ובחלק זה לא

גרף רביעי- 'Cumulative Number of Packets' – גרף קווי.

בגרף זה נראה כמה חבילות הגיעו במצטבר עד לאותה שנייה, בעצם כאן נוח לנו יותר לראות ולהשוות בעזרת גרף קווי בין תעבורות רשת שונות ודבר זה נוח ויסייע לנו בהמשך, מה גם שכאן אנחנו ממש משתמשים רק בכמה חבילות הגיעו באיזשהי שנייה (בלי קשר לגודל ומה יש בחבילה)

גרף חמישי- Packet Length Over Time – זהו בעצם גרף נקודות, ונסביר מה רואים בו.

תחילה נראה שבחלק מהתעבורות הראינו רק גרף אחד, ובחלק הראינו עם ובלי האנומליות (הקצה) אם הוא היה מאוד קיצוני.

בגרף זה ניתן לראות נקודות מפוזרות על מרחב הגרף, כך שכל נקודה בעצם זו חבילה, והמיקום שלה מתאר את הזמן בו היא הגיע ומה הגודל שלה.

קשה מאוד לשים 2 גרפים כאלו אחד על השני מ2 תעבורות אך בהשוואה אחד מול השני ניתן לראות שחלקם הבדלים יותר גדולים, ובחלקם הבדלים ממש מינימליים.

גרף שישי- 'Traffic Between IP Addresses' – גרף heatmap
בעצם בגרף זה אנחנו יכולים לראות את תדירות החבילות שמגיעות ממקור מסוים ליעד
ככל שהצבע כהה יותר כך יש פחות חבילות בין המקור ליעד הזה (או שכלל אין כאלו) וככל שהצבע יותר בהיר כך יש יותר כאלו.
זהו גרף שאם ניקח 2 תעבורות רשת שונות, ניצור את הגרף הזה ונשים אחד מול השני ניתן לראות הבדלים משמעותיים, וכן אפשר לראות זאת על הגרפים גם.

לבסוף מה שעשינו היה להשוות בין 2 סוגי גרפים שיצרנו
השווינו לפי הגרף הרביעי, כך שלקחנו את כל התוצאות של כלל התעבורות ושמנו אותם תחת גרף אחד מה שמראה את ההבדלים בין התעבורות השונות ואיך כל אחד מתנהג
ובנוסף השווינו גם לפי הגרף השני של הפרוטוקולים השונים, שמנו את ששת גרפי העוגה אחד ליד השני כך שניתן לראות את ההבדל המשמעותי בין אחד לשני

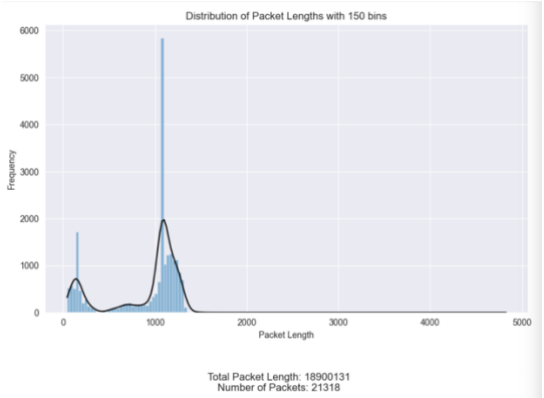
לבסוף פירטנו את מסקנותינו מן הגרפים הנ"ל שהצגנו.

נרצה לציין שוב את עניין התוקף, ניתן לראות שוב שרוב הגרפים שיצרנו הינם לא מתייחסים כלל למידע שנמצא בתוך חבילות הרשת אלא רק למידע חיצוני כמו גודל וזמן הגעה שבעצם תוקף שמושך את חבילות אלו יוכל לקבל בדיוק את המידע הזה, ובעזרת הגרפים והמידע שהצגנו כאן לסווג את התעבורה לפי נתונים אלו.

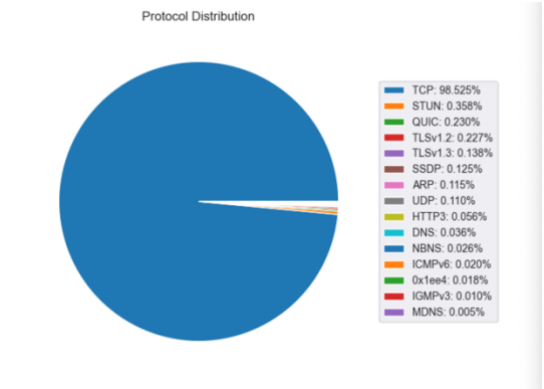
כל המידע נמצא גם במחברת הקוד, עם הסברים מפורטים והסברים על כל גרף וגרף מה אנו רואים ולבסוף מסקנות מכלל המידע שצפינו בו וניתחנו אותו.

נראה דוגמאות לגרפים, שאפשר לראות אותם במחברת Network_traffic_analysis.ipynb: את הדוגמאות נראה מתעבורות רשת שונות ולא תעבורה ספציפית.

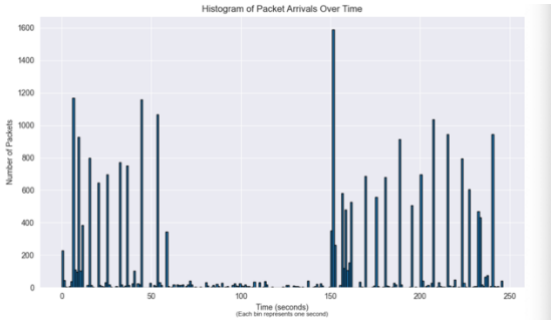
דוגמא לגרף מספר 1:



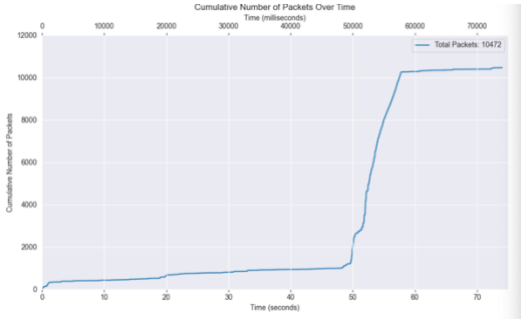
דוגמא לגרף מספר 2:



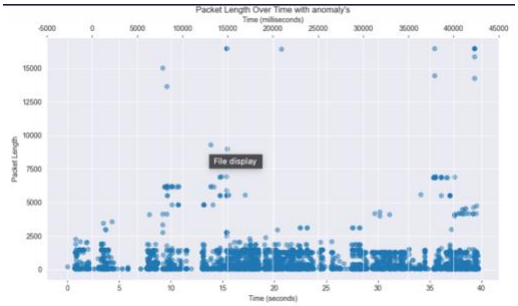
דוגמא לגרף מספר 3:

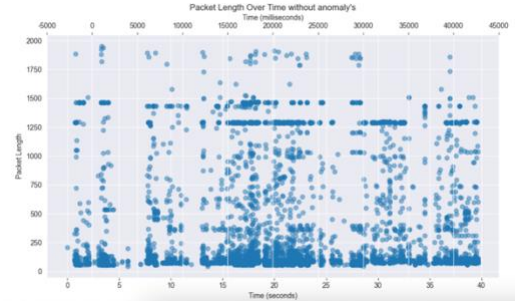


דוגמא לגרף מספר 4:

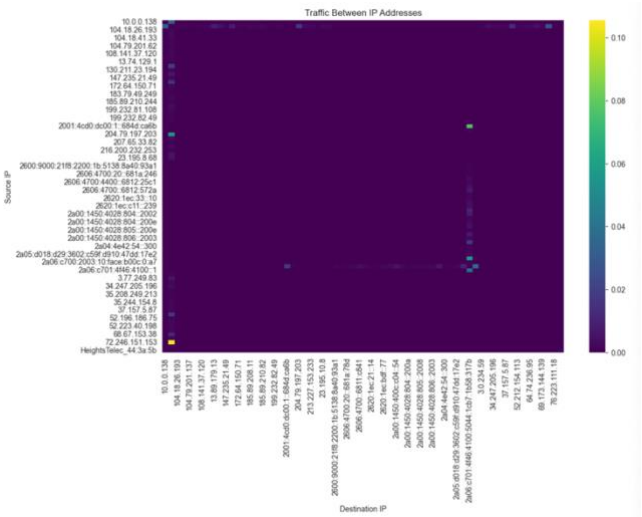


דוגמא לגרף מספר 5 (עם אנומליות ובלי):



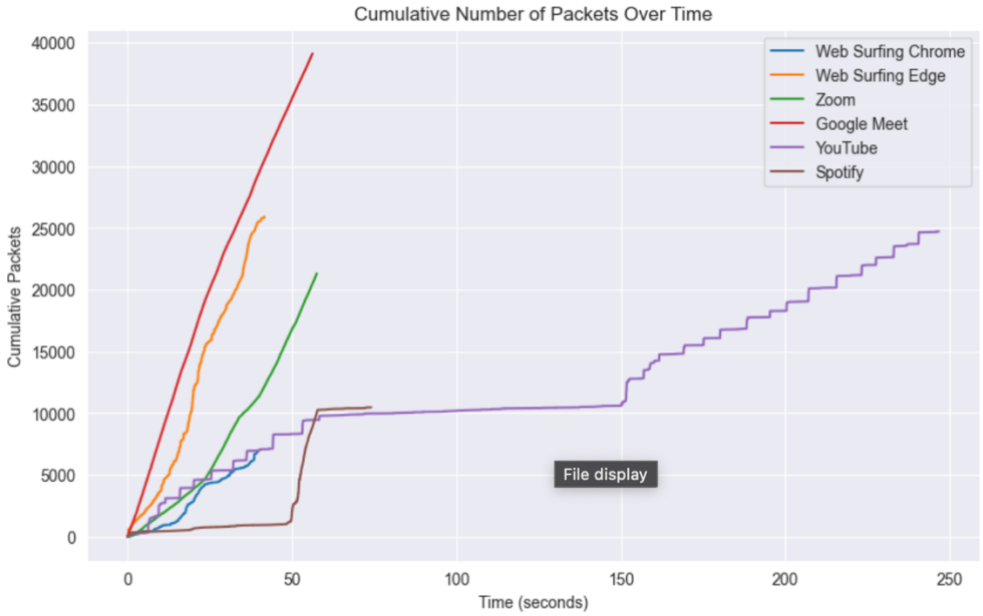


דוגמא לגרף מספר 6:

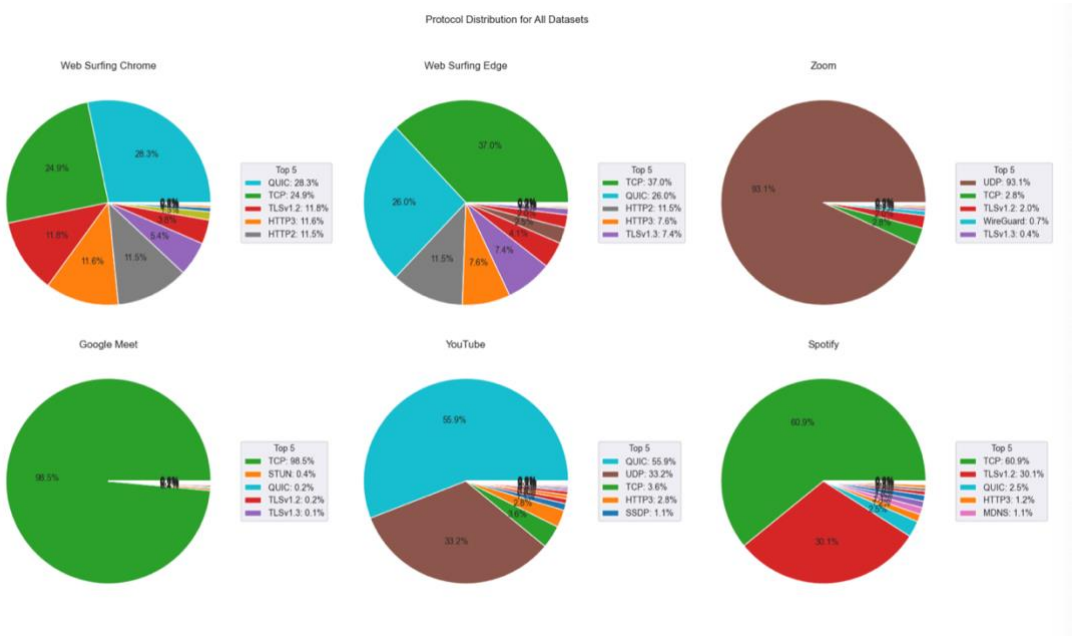


כעת נרצה להראות את 2 הגרפים שאותם יצרנו על מנת להשוות בין תעבורות הרשת השונות כדי שתראו את הדברים עליהם דיברנו, והם מוסברים בהרחבה במחברת.

<u>הגרף אשר מראה את ההבדלים בין הגרפים מסוג גרף 4: (כמות חבילות שהגיעו עד נקודת זמן)</u>



<u>וכעת נראה גם את ההבדלים בין הגרפים מסוג גרף 2: (הבדלים בין פרוטוקולים בתעבורה)</u>



את כלל המידע הנוסף וכל הגרפים ניתן לראות במחברת המצורפת כולל פירוטים והסברים בפרט על כל גרף בפני עצמו, ומסקנות מכל התהליך.

כלל הרעיונות לסוגי הגרפים השונים נלקחו גם מהמאמרים אותם קיבלנו על ידכם.

למידת מכונה, לניתוח המידע בצורה חכמה:
את כל המידע ואופן העבודה אפשר לראות במחברת:
Network_trafic_recognition_model.ipynb, אנו ממליצים לקרוא את המידע הנרשם פה, לצד המחברת וכך לראות גם את הגרפים והמידע הרלוונטיים.

מציאת הדאטה:
תחילה נרצה להסביר על למידת המכונה, הדאטה שבו השתמשנו, ועל תהליכים מסוימים אותם בהם השתמשנו.
נסביר תחילה על הליך בחירה הדאטה (שגם עבר אישור של פרופ׳ עמית דביר).
רצינו לבנות מודל שיחזה בצורה חכמה את התעבורה הניתנת לו, לשם כך חיפשנו דאטה שיראה לנו מידע שהוא לא נכנס לתוך המידע של החבילות עצמן אלא מתייחס בעיקר לפרטים יבשים של החבילות כפי מה שהשתמשנו בו בחלק הקודם.
חיפשנו במספר מקורות כגון מאמרים שונים ואתר Kaggle ורוב הדאטה שהייתה היא הייתה או לא מותאמת למידע הקיים לנו לפרויקט מהסוג הזה או שזו הייתה דאטה שהייתה זמינה רק בתשלום.
לאחר חיפושים ומאמצים מצאנו באתר Kaggle דאטה שהתאימה הכי טוב שהצלחנו למצוא בשביל הפרויקט הזה, על מנת להשתמש בדאטה הזו שלחנו בקשה בפורום לאישור על שימוש בה ואושרנו על ידי פרופ׳ עמית דביר.
לאחר מכן כפי שלמדנו בקורסים קודמים כאשר מתחילים לבנות מודל יש מספר שלבים, מציאת/ איסוף הדאטה, ניקוי הדאטה, הבנה של המצב, הפעלת למידת מכונה או מודל חיזוי, ולבסוף שימוש בנתונים.
לכן עד נקודה זו, מה שעשינו היה רק למצוא את הדאטה עליה נרצה לעבוד.
הדאטה כולל את הדברים הבאים: 145,670 תעבורות מסוגים שונים.
86 פיצ׳רים שונים שלפיהם מדדו כל חבילה (את הפירוט שלהם ניתן לראות במחברת)
וסיווג של החבילה למספר קבוצות שונות (וידאו סטרימינג חיים, נגני וידאו, נגני האודיו, העלאת קבצים, הורדת קבצים, ודפדפן או תעבורה אחרת)

ניקיון הדאטה:
כעת מצאנו דאטה שקיימים בה 86 פיצ׳רים, כלומר 86 עמודות שכל עמודה מתארת לנו מידע שונה על כל חבילה.
ניקיון הדאטה שלנו הולך להתבצע על העמודות ולא על השורות, כלומר נרצה לראות כמה פיצ׳רים נוכל להוריד ועדיין לקבל תוצאות טובות.
לשם כך נשתמש במספר כלים אותם נסביר (PCA (Principle Component Analysis , ולאחר מכן נבצע את בחירת הפיצ׳רים.
בPCA בעצם מה שאנחנו רוצים לעשות זה הדבר הבא- אנחנו נרצה לראות כמה העמודות קשורות אחת לשנייה ובעצם לראות בערך כמה עמודות מספיקות לנו בכך שהן מתארות מספיק טוב את רוב הדאטה לפי הסיווגים
באמת ראינו שבאמת אם נרצה מספר פיצ׳רים שיתארו כ95 אחוזים מהדאטה נשתמש בכ24 פיצ׳רים בערך, אך כאשר נרצה משהו שיהיה קרוב מאוד ל100 אחוז מהדאטה, נשתמש בערך בכ36 (מעל 35) פיצ׳רים.
וכעת נגיע לשלב בחירת הפיצ׳רים, כאן השתמשנו בספריית sikitlearn.feature_selection שבעצם בדקנו מספר שונה כל פעם של פיצ׳רים שבהם נרצה להשתמש, (שהוא בוחר אותם בעזרת עצי החלטה) מ1 עד בערך ל42 פיצ׳רים. וניתן לראות בגרף שיש קו כחול חזק, שמתאר את הממוצע של אותו k (מספר פיצ׳רים שחברנו לבדוק) ועוד תחום בהיר יותר מסביב שמתאר את סטיית התקן של אותו פיצ׳ר.
רצינו שהמכונה תהיה אוטונומית לחלוטין ללא מגע יד אדם, כלומר שלא נצטרך לנתח את הגרף ולהחליט מה הk הכי טוב לנו, אלא שיתבצע רק על ידי המכונה. ולכן בחרנו את המקסימלי, שכאן היה באמת מספר 36. וכעת רצנו עם הדאטה שיש לנו, ו36 הפיצ׳רים שנבחרו בשלב זה אל השלב הבא.

נרצה להגיד שבשלב בחירת הפיצ׳רים כבר ביצענו ניתוח והבנה של המידע הקיים לנו בדאטה-סט הנ״ל ולכן יכולנו להמשיך קדימה אל שלב המודלים.

מודלי חיזוי:
ניתן כעת מעבר על כלל המודלים איתם בחרנו לחזות, ניתן הסבר קטן על כל מודל, איך הוא עובד, ולאחר מכן נריץ כל אחד, בדרך טיפה שונה על מנת לגרום לאותו מודל להיות הכי טוב (שהוא יכול להיות).
סוגי המודלים בהם השתמשנו מיועדים לבעיות סיווג (קלסיפיקציה), שכן זו הבעיה הקיימת כאן, קיים לנו מידע מסוים ואותו נצטרך לסווג לקטגוריה 1 ויחידה.

סוגי המודלים השונים בהם בחרנו להשתמש על מנת לנסות לחזות הם:
SVM, Random Forest, XGBoost, Neural Networks,
KNN, Logistic Regression, Ensemble Methods

כעת נרצה להסביר בקצרה על כל אחד, ההסברים על כל אחד בפרט מפורטים גם במחברת בה עבדנו על המודלים האלו לחיזוי.

KNN- בעצם זה מודל יחסית פשוט אשר אינו דורש אימון על הדאטה ופשוט טוב כאשר יש איזשהי חזרתיות מסוימת והתאמה של פיצ׳רים לסיווג של אותו נתון, הרעיון הכללי של מודל זה הוא לראות מי השכנים הכי קרובים לאותו מידע שאליו אני מנסה לנבא לאיזה חבילה הוא שייך, וכך הוא מחליט לפי הרוב, איזה סיווג לתת, במודל זה לרוב נרצה להתאים את הK הכי טוב שהוא משתנה תמיד בהתאם לסוג הבעיה, לכן תחילה מה שעשינו היה להריץ כל פעם עם K שונה ובסוף לבחור את זה שהיה הכי מוצלח, שנתן לנו את התוצאות הכי טובות.
לאחר שמצאנו את הK הזה הרצנו את המודל על ה test שלנו.

Logistic Regression- בעצם מודל זה מספק לנו איזשהי מסגרת הסתברותית שבעזרתה נבין מהי רמת הביטחון בסיווג, מודל זה יעיל מאוד מבחינה חישובית ומציאת תוצאה, והוא מודל שנחשב יחסית פשוט אך נותן תוצאות טובות. השם שלו יכול לרמז על הדרך שבעצם בה הוא עובד, הוא בעצם מנבא מה ההסתברות שדוגמא מסוימת שייכת לאיזשהי קטגוריה ולאחר שנריץ את פונקציות ההסתברות לכל הקטגוריות בעצם נוכל לראות מה ההסתברות הכי גבוהה לאותה חבילה להיות בקטגוריה מסוימת, את מודל זה מריצים מספר פעמים, עושים לו איטרציות מרובות, ולכן נרצה תחילה למצוא את כמות האיטרציות שתיתן לנו את התוצאה הכי טובה לחיזוי הזה

SVM- שמו המלא של המודל הוא מכונות וקטורי תמיכה (support vector machine)
ובעצם מודל זה אמור להיות יעיל במיוחד במרחבים רב ממדיים כאלו של תעבורת רשת, והן יעילות מבחינת זיכרון כי בעצם מודל זה משתמש רק בתת קבוצה של וקטורי תמיכה.
מודל זה הוא מודל שעובד באופן הבא: המודל מנסה למצוא מישור/ היפר- מישור אם אנחנו מדברים על מודל רב ממדי כמו אצלנו, ונפריד ל2 קבוצות, הווקטורים שיהיו קרובים להפרדה הן יהיו הקריטיות כי הן בעצם מקבעות את המיקום של אותו מישור, ובעצם בסוף נעשה איזשהי טרנספורמציה על מנת למצוא את הקטגוריות. בשביל להתאים את זה לבעיה שלנו של קטגוריות מרובות בעצם נוכל להפעיל מספר רב של פעמים את המודל. ובכך ליצור את ההפרדות. בקוד ברגע בעצם שאנחנו מאמנים את המודל על הדאטה הוא עושה זאת לבד, ומפעיל את כל המודל לבד בעצם ללא בחירה שלנו בנתונים.

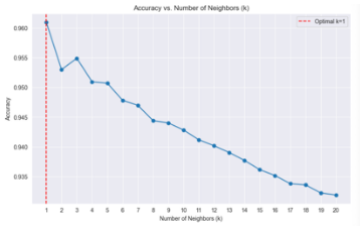
Random Forest- תחילה נדבר על עצי החלטה לאחר מכן נדבר על יערות, עץ החלטה שהוא בעצם עץ שיכול להגיע עד לעומק שנבחר שבעצם מסתכל על מספר פרמטרים ומחלק כל פעם לפי המידע שמפריד הכי הרבה בין סוגים שונים של סיווג
ביער בעצם יהיו לנו מספר עצי החלטה, כך שכל עץ מאומן של דגימה אחרת של הנתונים, ולפי הפיצ׳רים השונים מפריד בצורה הטובה ביותר כך שבסוף מגיע להחלטה.
זה עובד מאוד מאוד טוב עם ערכים מספריים ויחסים מורכבים בין דברים ממש כפי שיש לנו בתעבורת הרשת.
במודל זה נרצה להשפיע על עומקי העצים, כי מצד אחד אולי ככל שהעומק גדול יותר נגיע לדיוק יותר גבוה (לא בהכרח) אבל הסיבוכיות גדלה כי זה עוד רמה שלמה של צמתים.
לכן נבדוק מה העומק האופטימלי אותו שבו נרצה להשתמש.

XGBoost- מודל זה נגזר מהמודל (Gradient boosting) שכן זה Extreme הרעיון של מודל זה הוא שימוש במספר רב של מודלים חלשים, כך שכל מודל שמוסיפים מתמקד בעיקר בתיקון השגיאות של המודל הקודם, וכן הלאה וכן הלאה, כלומר כל מודל פנימי הוא פשוט ומטרתו היחידה לתקן את הטעויות של המודל הנמצא לפניו. הוא מתחיל עם ניבוי פשוט, מסתכל על שגיאות ומאמן מודל חדש לניבוי השגיאות האלה וכן הלאה והלאה, עד למצב של התכנסות, כלומר שהגענו למצב שהוא יחסית יציב, או מספר איטרציות שהוגדרו, בנוסף במודל זה משלב דבר הנקרא פונקציית אובדן שמסייע למנוע התאמת יתר של המודל לדאטה עליה אנחנו מאמנים. בדרך כלל מודל זה משתמש בעצי החלטה יחסית בעומק נמוך.

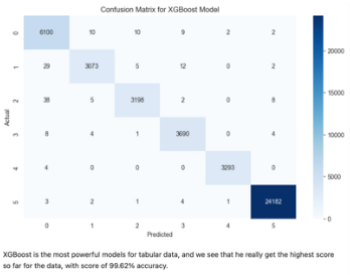
Neural Networks- מודל מסוג זה מסוגל לזהות דפוסים מורכבים ולא ליניאריים על נתונים מסויימים, מודל זה מתאפיין גם בתהליך הלמידה שלו, אמנם קיימים סוגים שונים ומגוונים של רשתות נוירונים שונים, אבל הרשת עובדת כך, קיימים לנו נוירוני קלט שאליהם אנחנו מכניסים את המידע, באמצעות המידע שקיים בהם כל נוירון מחליט האם להיות מופעל או לא (״לירות או לא לירות״) במידה והוא מופעל הוא מחובר לנוירונים אחרים עם כל מיני סוגי משקלים שונים, ולפי כלל המידע שמגיע לנוירון הבא מכל הנוירונים שלפניו, אותו נוירון מחליט האם להיות מפעל או לא וכן הלאה, עד שבסוף התוצאה אמורה להדליק נוירון יחיד שהוא בעצם הסיווג שלנו.
האימון שלנו מחשב את ההפרש בין מה שהיה אמור להיות לבין התוצאה ומחשב את שוב את המשקלים מחדש (מעדכן אותם), עד שנגיע למספר איטרציות שאותו הגדרנו או להתכנסות.
מכיוון שלא תמיד אפשרי להגיע להתכנסות ברשתות נוירונים, בדקנו אצלנו על מספר איטרציות (Epochs) שונים וראינו איזה מביא לנו את התוצאות האופטימליות.

Ensemble Methods- בעצם מודל זה משלב לנו מודלים שונים, ומחשב לנו באמצעותם את הסיווג של המידע, למודל זה נוכל להכניס באיזה מודלים אנחנו רוצים שהוא ישתמש, ומה הנתונים של כל מודל לפי אותו סוג מודל, ובעצם הוא מפעיל את כולם ביחד ובוחר בסיווג המתאים לפי המודלים שבהם השתמש, בדוגמא שלנו רצינו לראות האם השימוש בכלל המודלים יחד ייתן לנו תוצאות טובות יותר מאשר כל המודלים בנפרד
לכן השתמשנו כאן בכל ששת המודלים אותם הסברנו לפני.

בכל מודל הצגנו בין גרף 1 ל2 גרפים.
גרף 1 (אופציונלי, לא היה בכל המודלים):
הראינו בגרף זה שאם יש איזשהו משהו שאותו רצינו לשנות במודל ולראות איזה נתון ייתן לנו את התוצאה הטובה ביותר אז הראינו את השינוי בבחירה בו ובסוף את הנתון שאותו בחרנו:
הדוגמא שנציג כאן היא מהKNN ששם בעצם רצינו לראות מה המספר שכנים שייתן לנו את התוצאה האופטימלית (K) לכן הצרנו את האלגוריתם מספר רב של פעמים, כל פעם עם K אחר על מנת למצוא את ההכי טוב. בדוגמא כאן זה היה K=1.



גרף 2:
שקיים בכולם זהו בעצם בconfusion matrix שבעצם פועל כך, ציר הY אלו בעצם הסיווגים האמיתיים, ציר הX אלו הסיווגים שהמודל חזה, ולכן כל מה שעל האלכסון זה מספר החבילות שהמודל צדק בחיזוי שלהם, כל מה שמעל ומתחת אלו בעצם טעויות חיזוי של המודל, שהוא סיווג לקבוצה מסוימת כאשר בעצם זה היה שייך לקבוצה אחרת. הצבע כהה יותר ככל שיש יותר חבילות שבהן בקטגוריה הנוכחית (X,Y)



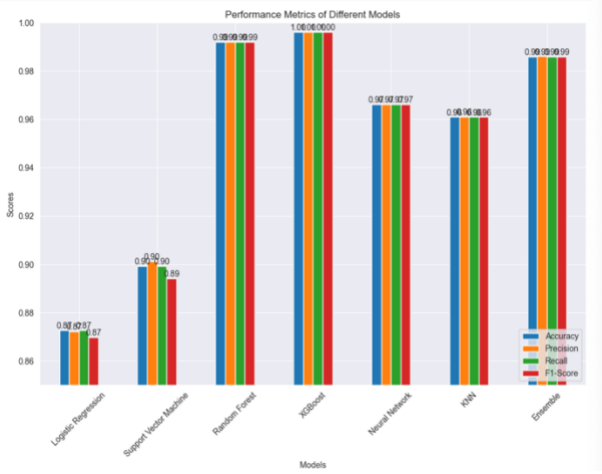
נוסף לגרפים אלו הצגנו גם עבור כל מודל את כל התוצאות שלו: precision, Recall, F1-Score שעליהם נפרט מטה, וגם רשמנו עבור כל קבוצה שאותה המודל נדרש לסווג (סה"כ 5 קבוצות) מה הSupport שלו כלומר מה הכמות שיש מכל קבוצה שעליה אנחנו בודקים.
אם יש 6000 שורות של תעבורה מסוג מסוים אז בעמודת ה Support יהיה רשום 6000.
הטבלאות מראות לכל קבוצה את כל אחד מהנתונים הללו ובסוף סיכום של כולם.
בנוסף לפני הטבלה נוכל לראות את הAccuracy כלומר הדיוק של כל מודל.
נראה דוגמא גם עבור גרף כזה

```
ACCURACY: 0.9919912132167864
Classification report for test set
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      6133
           1       0.98      0.96      0.97      3121
           2       0.99      0.97      0.98      3251
           3       0.98      0.99      0.98      3707
           4       1.00      1.00      1.00      3297
           5       1.00      1.00      1.00     24193

    accuracy                           0.99     43702
   macro avg       0.99      0.98      0.99     43702
weighted avg       0.99      0.99      0.99     43702
```

לאחר הצגת תוצאות של כלל המודלים בעצם סיכמנו בגרף יחיד את כלל המודלים עם המידע הבא:
Accuracy- דיוק, הדיוק הכללי של כל מודל, כמה הוא צדק
Precision- מתוך כל מי שהמודל חזה בקבוצה מסוימת, כמה באמת היו שייכים לקבוצה זו, כלומר אם המודל חזה משהו, כמה באמת אנו יכולים לסמוך עליו
Recall- מתוך כל אלו שהם באמת שייכים לקבוצה מסוימת, כמה באמת הצלחנו לחזות, כלומר האם המודל שלנו מצליח לתפוס את כל המקרים (או הרוב המוחלט של המקרים) בקבוצה מסוימת
F1-Score- זהו מדד אשר משלב את 2 המדדים הקודמים, יכול להיות שכאשר נרצה להתעסק בחקר מסוים יהיה אכפת לנו יותר מהאחד על השני, אך לא תמיד, ובשביל זה F1-Score קיים.



ראינו בגרף שיצא ששני המודלים של Logistic Regression, SVM נתנו תוצאות מאוד לא טובות (ביחס לשאר) דיוק של לא יותר מ90 אחוזים
למודלים של KNN, Neural Networks ראינו אחוזי הצלחה יותר גבוהים, מעל 95 אחוזי הצלחה ודיוק, שאלו תוצאות טובות לKNN מכיוון שמאוד פשוט, אך יכול להיות שעבור רשתות הנוירונים יכולנו לעשות שינויים או להשתמש ברשתות נוירונים מסוגים שונים על מנת לשפר את המודל.
שלושת המודלים שנתנו לנו את התוצאות הכי גבוהות היו, יערות החלטה, XGBoost, והלמידה המשולבת שמשלבת את כלל המודלים.
ניתן לראות שאל מול כולם, XGBoost נתן את התוצאות הטובות ביותר ולכן כנראה שזהו המודל הכדאי לבעיות מהסוג הזה

	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.872729	0.872213	0.872729	0.869857
Support Vector Machine	0.899181	0.901217	0.899181	0.894072
Random Forest	0.991991	0.992027	0.991991	0.991981
XGBoost	0.996202	0.996210	0.996202	0.996200
Neural Network	0.966203	0.966208	0.966203	0.966179
KNN	0.960986	0.961006	0.960986	0.960987
Ensemble	0.985996	0.986051	0.985996	0.985958

לבסוף הראינו וסיכמנו את מסקנותינו מהמודל הזה אל מול הצגת התוצאות, ונראה שגם עם פרטים שלא מתייחסים למידע פנימי של החבילות אפשר להסיק הרבה מאוד על תעבורת הרשת, דבר אשר העלה לנו שאלות רבות בנושא אבטחת מידע, ואמינות של הבטחת מידע מקצה לקצה.

**כל המידע שהיה דרוש לנו לחלק זה של הפרויקט, הן בהצגת הגרפים, הן בהצגת הנתונים, והן בשימוש של מודלים וידע במודלים של חיזוי, ניתן לנו במסגרת הקורסים הבאים: מבוא למדעי הנתונים, הדמיית נתונים, ומבוא לנוירו-חישוביות.
ארבעתנו סטודנטים במסגרת המסלול של מדעי הנתונים, ובינה מלאכותית, ולכן היה לנו את הידע הדרוש לביצוע משימות אלו, ללא שימוש במקורות נוספים אלא בידע הקיים לנו.

ביבליוגרפיה:

חלק א:

שאלה 1,2:

מצגת פרק שלישי- שכבת התעבורה. – מתוך חומר הקורס הנלמד

שאלה 3:

לפי המצגות של ד"ר איתמר כהן על שכבת הרשת. – מתוך חומר הקורס הנלמד

שאלה 4:

MPTCP

data tracker

שאלה 5:

Fortinet

webservertalk

חלק ב:

בחלק ב, כל שנעזרנו בו היה במאמרים שכן, בהם היה מוסבר כל הידע הדרוש לנו.

חלק ג:

כל החומר הדרוש לנו לחלק זה ניתן לנו במסגרת המסלול של מסלול מדעי הנתונים ובינה מלאכותית, באוניברסיטת אריאל.

חומר הקורס הנלמד בקורס מבוא למדעי הנתונים – הועבר ע״י פרופ׳ עמוס עזריה
חומר הקורס הנלמד במסגרת קורס הדמיית נתונים – הועבר ע״י ד״ר רועי יוזביץ
חומר הקורס הנלמד במסגרת מבוא לנוירו- חישוביות- הועבר ע״י גב׳ מירב שוקרון