## Review 61: PonderNet: Learning to Ponder

**פינת הסוקר:**

**המלצת קריאה ממייק**: מומלץ בעיקר להרחבת אופקים

**בהירות כתיבה**: גבוהה

**ידע מוקדם**: הבנה בסיסית ברשתות ובחוקי הסתברות

**יישומים פרקטיים אפשריים**: --

**פרטי מאמר:**

**לינק למאמר:** [זמין להורדה](https://arxiv.org/pdf/2107.05407.pdf).

**לינק לקוד**: [לא רשמי 1](https://github.com/labmlai/annotated_deep_learning_paper_implementations), [לא רשמי 2](https://github.com/lucidrains/ponder-transformer)

**פורסם בתאריך**: 02.09.2021, בארקיב (v2)

**הוצג בכנס**: (8th ICML Workshop on Automated Machine Learning (AutoML 2021

**תחומי מאמר:**

* מודלים (במקרה הזה רשתות נוירונים) בעלי זמן חישוב אדפטיבי (תלוי במורכבות משימה)

**ידע מוקדם:**

* מודלים (במקרה הזה רשתות נוירונים) בעלי זמן חישוב אדפטיבי (תלוי במורכבות משימה)
* מרחק KL בין התפלגויות
* התפלגות גיאומטרית

**תמצית מאמר:**

המאמר הנסקר משתייך לתחום שלא הייתי מודע לקיומו עד שקראתי אותו. התחום דן ברשתות נוירונים שמסוגלות להתאים את כמות החישובים למשימה נתונה בהתאם לרמת המורכבות של המשימה. כלומר עבור המשימה "קלה" רשת כזו תבצע פחות חישובים מאשר למשימה "מורכבת יותר".

המטרה העיקרית כאן היא להקנות למודל את היכולת להפסיק את תהליך האימון במצב בו נראה כי הוא ״הצליח ללמוד את מה הוא היה שצריך״, וממילא המשך האימון לא צפוי לשפר את ביצועי המודל באופן משמעותי. אם לעומת זאת המודל ״רואה״ כי איטרציות אימון נוספות עשויות להניב תוצאות טובות, הוא בוחר להמשיך את האימון. אציין כי רמת המורכבות של משימה אינה מועברת כקלט לרשת אלא הרשת צריכה "להחליט" on-the-fly עד כמה המשימה מסובכת ולהתאים את כמות החישובים הנדרשת.

המאמר הנסקר מציע שיטה, הנקראת PonderNet, ההוכפת רשת נוירונים נתונה ל״אדפטיבית מבחינה חישובית״, כלומר כזו שיודעת להתאים את כמות החישובים לפי רמת המורכבות של בעיה. PonderNet דורשת שינויים קלים לארכיטקטורת הרשת, ו״**מצליחה להשיג איזון בין ביצועי המודל על סט אימון, כמות החישובים הנדרשת ויכולת הכללה של הרשת**" (לשון המאמר).

**הסבר של רעיונות בסיסיים:**

הרעיון של המאמר הוא די פשוט וטבעי. בהרצה הראשונה של הרשת מזינים לה את הקלט המקורי (נגיד, תמונה, טקסט או קטע אודיו) ומקבלים כפלט את הייצוג החבוי (ה"לטנטי") שלו. ייצוג לטנטי זה משמש כקלט להרצת רשת הבאה. לאחר מכן מריצים את הרשת פעם אחרי פעם כאשר הקלט (ייצוג חבוי - hidden state) לכל הרצה (איטרציה) הוא הפלט של האיטרציה הקודמת (n-1). בנוסף, לאחר כל איטרציה הרשת מספקת את החיזוי עבור המשימה המקורית של הרשת ואת ההסתברות לעצירת הריצה . כלומר, פלט של רשת אחרי איטרציה n הוא השלישיה , כאשר s הוא מיפוי הממודל באמצעות רשת נוירונים כללית (LSTM, MLP, encoder-decoder). במאמר s נקראת פונקציית מדרגה (step function).

כעת נדון בדקות מעניינת לגבי ההסתברויות לעצירה . כאמור מתארת הסתברות לעצירת ריצה של רשת באיטרציה n. באופן פורמלי היא הסתברות **מותנית של עצירה בשלב n בהינתן אי עצירה (המשך) בשלב (n-1)**. זה, להבדיל מהסתברות לעצירה **הבלתי מותנית** לאחר איטרציה n שניתן לחשב אותה באופן הבא:

המאמר הנסקר מציין כי העבודות הקודמות ניסו לחזות דווקא את ולא .

נציין כי מגדירה התפלגות הסתברותית תקינה כאשר מספר האיטרציות המקסימלי אינו מוגבל. כמובן שזה עלול להיות בעייתי עבור שימושים פרקטיים של השיטה המוצעת. המאמר מציע שתי דרכים להתמודד עם סוגיה זו ונדון בהן בהמשך הסקירה.

**איך מתבצע חיזוי עם PonderNet:** אחרי שהסברנו מה הרעיון שעומד מאחורי PonderNet נשאלת השאלה: איך בעצם מבצעים חיזוי עם הרשת הזו? כאמור, בכל איטרציה הפלט של הרשת מורכב מהחיזוי עבור המשימה המקורית, יחד עם האומדן של ההסתברות לעצירה לאחר האיטרציה הנוכחית . אז איך אנו יודעים מתי לעצור את הריצה? פשוט מאוד - מבצעים דגימה אחת של משתנה בינארי עם הסתברות הצלחה ומחליטים על סמך התוצאה האם לעצור או להמשיך. במילים אחרות אחרי כל איטרציה ״זורקים״ מטבע (לרוב לא הוגן) כאשר על צדדים של כתוב ״המשך״ ו״עצור״ כאשר הסתברות של ״עצור״ הוא . במקרה של עצירה החיזוי האחרון משמש כחיזוי סופי של PonderNet עבור המשימה שבנידון.

**איך מאמנים PonderNet:** פונקציית לוס של PonderNet בכל איטרציה מורכבת משני איברים:

1. **הלוס המקורי של משימת הרשת**: לוס על ״איכות״ החיזוי , כמו למשל, כאשר y הוא הלייבל האמיתי (ground-truth). במאמר השתמשו בלוס הריבועי או בקרוס אנטרופי.
2. **לוס נוסף עבור הסתברות עצירת האימון**: איבר רגולריזציה בצורה של מרחק KL בין ההתפלגות לבין התפלגות פריור Pg. התפלגות Pg נבחרה ע״י המחברים בתור התפלגות גיאומטרית עם פרמטר מקונפג (היפרפרמטר). כמובן בשביל לחשב את מרחק KL בין לבין Pg צריך להריץ את PonderNet מספר מקסימלי של הרצות בלי לעצור אותו. למעשה לאיבר רגולריזציה זה יש שתי מטרות עיקריות: הראשונה ״לכפות״ על הרשת להיעצר לאחר הרצות (בממוצע) והשניה היא למנוע מרשת להוציא כפלט הסתברויות אפסיות לעצירה כל הזמן (סוג של עידוד exploration).

**נרמול של התפלגות** : כעת אספק הבהרה לגבי הסוגיה של נרמול ההתפלגות שהעלינו באחד הפרקים הקודמים. כאמור אנו לא יכולים להריץ את PonderNet לאורך מספר בלתי מוגבל של איטרציות ביישומים פרקטים. המאמר קובע את המספר המקסימלי של הרצות N, וקל לראות שהסדרה כבר לא מהווה פונקציה התפלגות תקינה כי סכום של הסדרה אינו שווה ל 1. המאמר מציע שתי דרכים לנרמול של :

1. לנרמל באופן סטנדרטי באמצעות חלוקה של כל בסכום של הסדרה.
2. ״להעביר״ את כל המסה ההסתברותית הנותרת לפני האיטרציה האחרונה להסתברות עצירה של האיטרציה האחרונה .

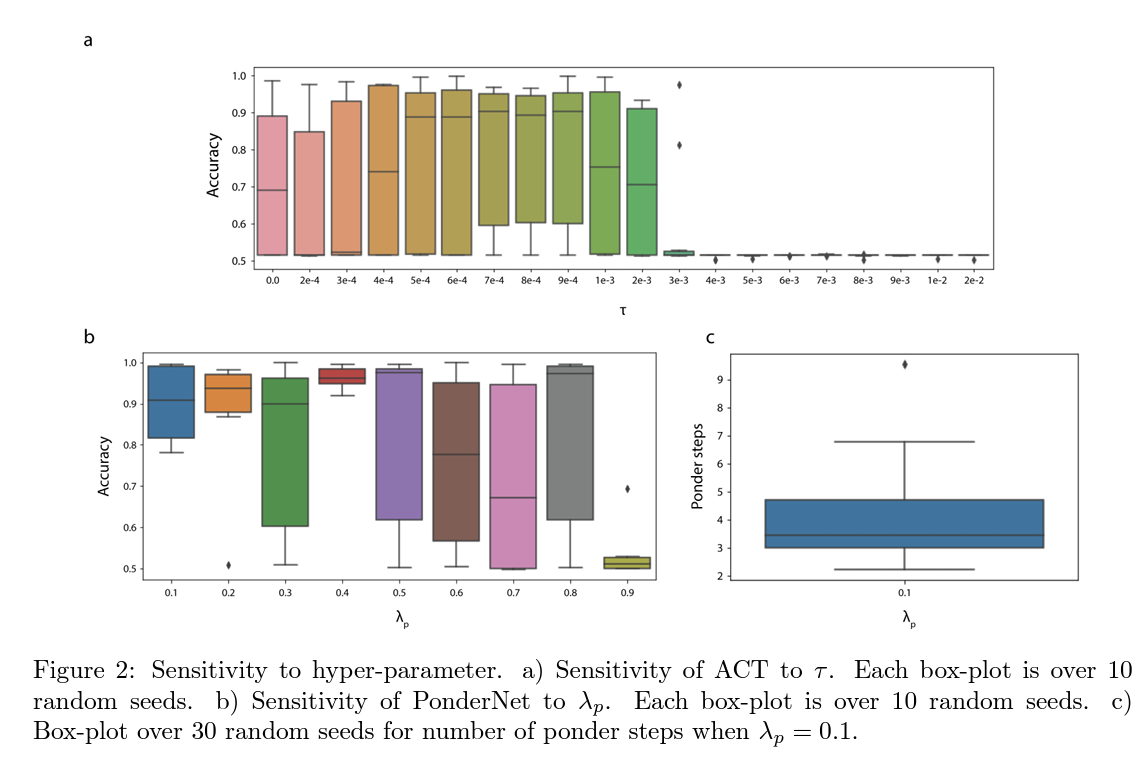
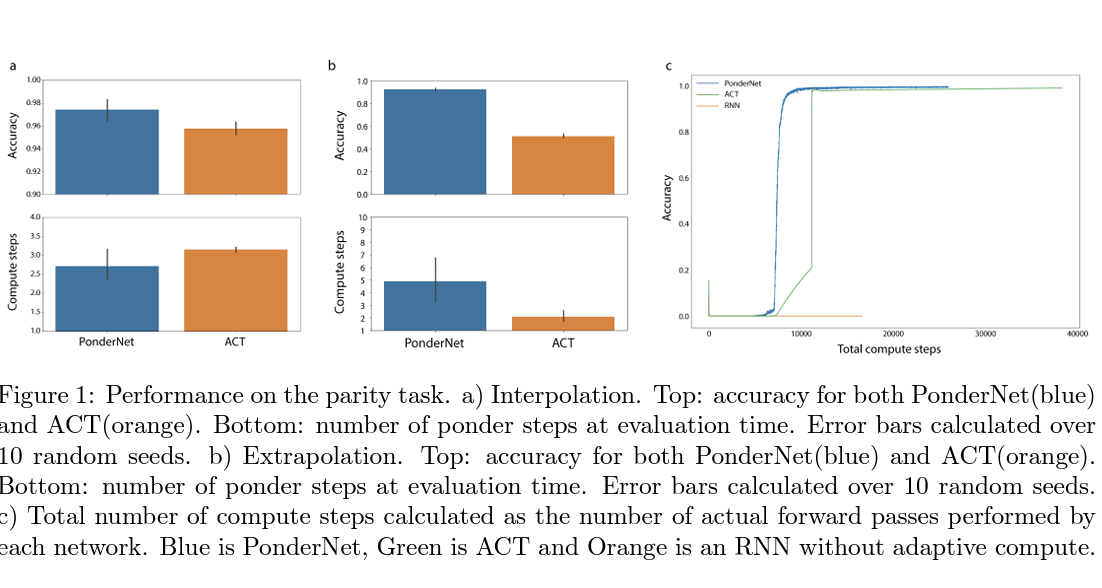
**איך קובעים את מספר האיטרציות המקסימלי N:** הנקודה המעניינת האחרונה שאני רוצה להתייחס אליה היא בחירה של מספר האיטרציות המקסימלי N. כמובן ניתן לאפטם אותו כמו כל היפרפרמטר אחר, אבל המחברים מציעים להגדיר אותו דרך "שארית של המסה הסתברותית לעצירה של הריצה". כלומר בוחרים מספר חיובי קטן c (במאמר בחרו ב- 0.05) ומגדירים את N כמספר המינימלי של איטרציות הנדרשות כדי שהסכום של יהיה גדול יותר מ- 1- c). זה כמובן נעשה במהלך אימון של PonderNet.

**הישגי מאמר:**

המחברים בחרו כמה משימות (שרובן "אינן מככבות" במאמרים בנושא הרשתות) והראו כי הביצועים של השיטה המוצעת עדיפה על גישות "אדפטיביות" האחרות עבור כמה ארכיטקטורות של פונקציית המדרגה s. למשל אחת המשימות היא חישוב של parity עבור סדרה בינארית ארוכה. ההשוואה התמקדה בעיקר בשיטה, הנקראת ACT, שכנראה נחשבה ל SOTA לפני כן. המחברים הראו ש- PonderNet מצליח גם במשימות של question answering, הנקרא bAbI (המורכב מ- 20 תת-משימות שונות). השיפור בביצועים התבטא בדרך כלל ביכולת להגיע לאותן ביצועים בפחות זמן מאשר הגישות המתחרות.

**נ.ב.**

לא הכרתי בעבר מאמרים הדנים במודלים בעלי זמן ריצה אדפטיבי והיה מגניב לצלול לנושא החשוב הזה. המאמר קל לקריאה, הרעיון העיקרי שלו אינטואיטיבי ומובן להפליא אולם נראה כי כרגע אין הרבה משימות ודומיינים שניתן ליישם אותו בהם.



## Review 62: Taming Transformers for High-Resolution Image Synthesis

**פינת הסוקר:**

**המלצת קריאה ממייק**: חובה ללא ספק!

**בהירות כתיבה**: גבוהה

**ידע מוקדם**: הבנה טובה בגאנים, טרנספורמרים ו VQ-VAE די הכרחית להבנת המאמר

**יישומים פרקטיים אפשריים**: יצירת תמונות באיכות מרהיבה (לא פחות!!)

**פרטי מאמר:**

**לינק למאמר:** [זמין להורדה](https://arxiv.org/abs/2012.09841).

**לינק לקוד**: [זמין כאן](https://github.com/CompVis/taming-transformers) (בנוסף יש עוד 3 מימושים ״לא רשמיים״)

**פורסם בתאריך**: 21.06.21, בארקיב (v3)

**הוצג בכנס**: CVPR 2021

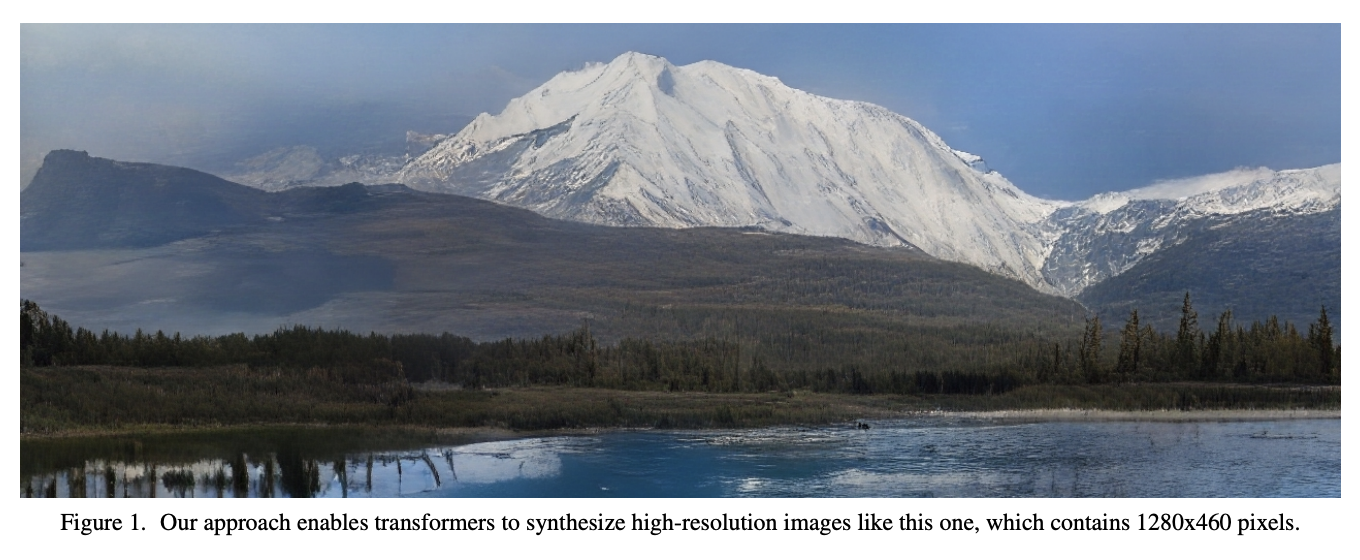
**תחומי מאמר:**

* מודלים גנרטיביים ליצירת דאטה בתחום הויזואלי

**ידע מוקדם:**

* [VQ-VAE](https://arxiv.org/abs/1711.00937)
* גאנים
* טרנספורמרים

**תמצית מאמר:**

****

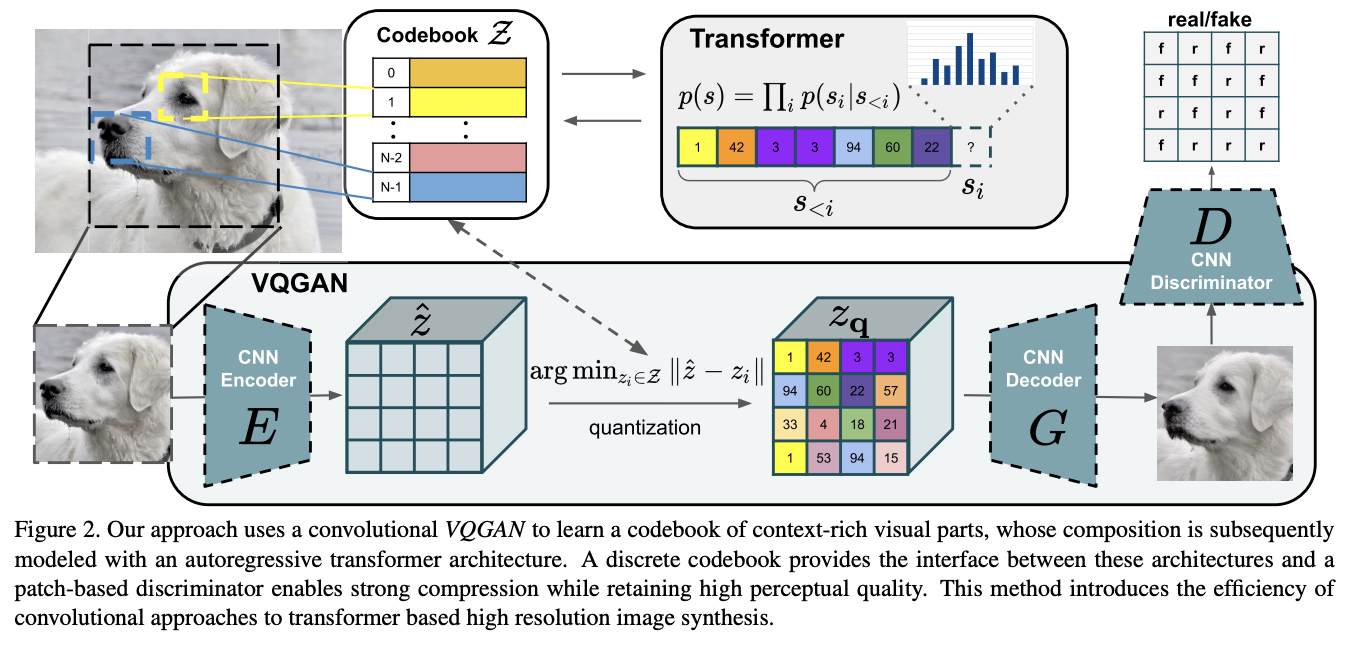
מודלים גנרטיביים ליצירת פיסות דאטה חדשות בתחום הויזואלי הגיעו לתוצאות מרשימות ב-3 השנים האחרונות. מודלים גנרטיביים כמו [StyleGAN3](https://nvlabs.github.io/stylegan3/) ו[-VQ-VAE2](https://arxiv.org/pdf/1906.00446.pdf) מצליחים לגנרט תמונות באיכות מאוד גבוהה במגוון רזולוציות. יתרה מזו התמונות הנוצרות באמצעות מודלים אלו נראות ממש פוטוריאליסטיות וכבר לא ניתן להבחין בין תמונה מגונרטת ל"טבעית".

רוב המודלים הגנרטיביים בעלי ביצועי SOTA בדומיין הויזואלי הינם גאנים בעלי ארכיטקטורה מבוססת על שכבות קונבולוציה (למרות שבשנה האחרונה [VAE](https://openreview.net/pdf?id=LcSfRundgwI)-ים, [מודלי דיפוזיה](https://machinelearning.co.il/10071/diffusion-models-beat-gans-on-image-synthesis-%d7%a1%d7%a7%d7%99%d7%a8%d7%94/) ו- [גישות אחרות](https://proceedings.neurips.cc/paper/2020/file/92c3b916311a5517d9290576e3ea37ad-Paper.pdf) התחילו להחזיר להם מלחמה). ביצועים עדיפים של רשתות קונבולוציה בדומיין הזה נובעים מה- "inductive bias" האינהרנטי שמאפיין רשתות מסוג זה. Inductive bias של רשתות קונבולוציה מנצל תלויות לוקאליות חזקות הקיימות בתמונות הטבעיות. לעומת זאת לטרנספורמרים אין bias כזה שמקשה עליהם ללמוד את האופיינים של התפלגות הדאטה בדומיין התמונות הטבעיות. עקב כך רוב הרשתות מבוססות טרנספורמרים בדרך כלל:

* או מצוידות ב-backbone הבנוי משכבות קונבולוציה להפקת פיצ'רים ״לֹוקאליים״
* או מוסיפים את ה-inductive bias לטרנספורמרים, כלומר נותנים יותר משקל לקשרים בין פאצ'ים קרובים בתמונה.

המאמר הנסקר שילב את שתי הגישות הנ״ל ועשה את הדבר הבא:

1. אימון של VAE שבו הן המקדד והן המפענח הינם רשתות קונבולוציה. למעשה המחברים השתמשו ב-VAE מקוונטט בו המרחב הלטנטי (המכיל פלטים של המקדד) הוא למעשה אוסף דיסקרטי של וקטורים הנקרא codebook; ארחיב על זה בהמשך).
2. שימוש בטרנספורמר (ובפרט במפענח שלו) עם תיבול קל של ״inductive bias" המתאים לדומיין התמונות, בשביל ללמוד את ההתפלגות מעל המרחב הלטנטי הדיסקרטי.
3. גנרוט של תמונה מתחיל מיצירה של וקטור לטנטי באמצעות הטרנספורמר המאומן. לאחר מכן מזינים את הוקטור הנוצר לרשת המפענחת של הטרנספורמר ליצירת תמונה.



עד כאן הכל טוב ויפה אבל בשם של המאמר מופיע גם מילה GAN והוא לא הוזכר עד עכשיו. למעשה VAE מהשלב הראשון מאומן בצורה לא סטנדרטית. לפונקציית הלוס הרגילה של VQ-VAE (שגם עברה מתיחת פנים) מוסיפים את פונקציית הלוס הסטנדרטית של הגאן. כלומר **בנוסף** למקודד ולמפענח של VQ-VAE מאמנים דיסקרימינטור D. מטרתה של D היא לזהות אם התמונה שנוצרה באמצעות המפענח של VQ-VAE או שזו תמונה אמיתית.

למעשה VQ-GAN היא ״חתונה משולשת״ ומוצלחת במיוחד של VQ-VAE, גאן והטרנספורמר.

**הסבר של רעיונות בסיסיים:**

אחרי שהבנו מה מה הן אבני הבנייה העקריים של VQ-GAN בואו נצלול לפרטים נבין איך החיה המורכבת הזאת עובדת. בשביל להסביר את אופן האימון של VQ-GAN אנו קודם כל נרענן בזכרון מה זה VQ-VAE שעל בסיסו בנוי שלב האימון הראשון של VQ-GAN.

**:?VQ-VAE מה זה**

VQ-VAE הינו סוג של (Variational AutoEncoder (VAE בעל מרחב לטנטי סופי (אך מאוד גדול). נזכיר ש- [VAE](https://arxiv.org/abs/1312.6114) רגיל הוצע ב- 2014 על ידי Kingma ו- Welling. למעשה VAE מהווה הכללה של [AutoEncoder](https://www.jeremyjordan.me/autoencoders/) סטנדרטי שהוא שיטה להורדת מימד לא לינארית. החידוש של VAE יחסית לאוטו-אנקודר הוא תוספת של הדרישה על התפלגות הייצוגים הלטנטיים של דאטה. כלומר בנוסף לכך ייצוג לטנטי של דאטה צריך לשמר את התכונותיו החשובות, וקטורי הייצוג עצמם צריכים להיות מפולגים לפי התפלגות נתונה (לרוב גאוסית סטנדרטית). תוספת זו מאפשרת לגנרט דאטה חדש באמצעות המפענח של VAE כאשר הקלט אליו הוא וקטורי ייצוג הנדגמים מהתפלגות זו. פונקצית לוס של VAE מורכבת מסכום של לוס השחזור הריבועי (המודד עד כמה טוב הצלחנו לשחזר את הקלט) ואיבר רגולריזציה הכופה התפלגות נתונה על הפלט של האנקודר (מרחק KL).

VQ-VAE הינו מודיפיקציה של VAE שבה המרחב הלטנטי (הנקרא codebook) הוא למעשה דיסקרטי ומכיל מספר סופי של וקטורי הייצוג. כדי לגנרט פיסת דאטה חדשה בוחרים וקטור מהמרחב הדיסקרטי הזה (שמכיל כמות עצומה של וקטורים ולכן בכל זאת מאפשר גנרוט של דאטה מאוד מגוון) ומעבירים אותו דרך המפענח המאומן.

כאשר משתמשים ב- VQ-VAE עבור יצירת תמונות בדרך כלל מחלקים תמונה ל- M פאצ'ים כאשר כל פאץ' מקדד באחד הוקטורים מה-codebook. במקרה הזה תמונה היא בעצם מערך באורך M של וקטורים מ-codebook (עם חשיבות לסדר כמובן!). למשל עבור M=8 (במציאות יש הרבה יותר פאצ'ים) ייצוג של תמונה יכול להיראות כך: [22, 46, 2, 11 ,98 ,17 ,9 ,78] כאשר כל איבר במערך זה הוא מספר סידורי של וקטור ייצוג מה-codebook. האימון של VQ-VAE הוא קצת טריקי כי בנוסף לפרמטרים של המקדד והמפענח צריך לאמן גם את וקטורי ה-codebook. וקטורים אלו נבחרים באמצעות **פעולה לא גזירה** מהפלט **z** של המקדד (בוחרים את הוקטור הכי קרוב **z** במונחי מרחק L2) שמקשה על ה-backprop. למי שמתעניין איך מתמודדים עם הסוגיה הזו ממליץ להעיף מבט ב- [בבלוג מעולה של ברקלי](https://ml.berkeley.edu/blog/posts/vq-vae/).

המבנה של פונקציית לוס של VQ-VAE מורכב מלוס השחזור הריבועי של VAE הסטנדרטי והמרחק הריבועי בין פלט של המקדד והוקטור הקרוב מה-codebook (למעשה זה טיפה יותר מורכב עקב הפעולה הלא גזירה שתוארה קודם).

**פונקציית לוס של VQ-GAN:**

המאמר הנסקר בחר להחליף את לוס השחזור הריבועי בסכום של:

* [הלוס הסטנדרטי](https://developers.google.com/machine-learning/gan/loss) של גאנים (שכמובן מצריך אימון רשת דיסקרימינטור).
* הלוס הפרספטואלי ([perceptual loss](http://svl.stanford.edu/assets/papers/JohnsonECCV16.pdf)).

המאמר טוען כי פונקציית הלוס המוצעת באה לתת ״טיפול שורש״ בסוגיית הכואבת של VAE: התמונות המטושטשות (יחסית לגאנים למשל) שהוא מייצר. הסיבה לכך טמונה במבנה של איבר השחזור של פונקציית הלוס הריבועית של VAE, שממזערת את השגיאה **הממוצעת** הגורמת לתמונה המשוחזרת להיות קרובה לתמונה המקורית, **״אך רק בממוצע**״. תופעה זו, של קושי של רשתות עמוקות להתמודד עם תדרים גבוהים בקלט ובפלט, ידועה ונכתבו עליה לא מעט עבודות לאחרונה ([[1]](https://arxiv.org/abs/1911.06465), [[2]](https://arxiv.org/abs/2003.01826), [[3]](https://arxiv.org/abs/2012.05535), [[4]](https://arxiv.org/abs/2102.06108)).

**הלוס הפרספטואלי:**

כעת נסביר מהו הלוס הפרספטואלי Lper. המטרה של Lper היא למדוד דמיון בין הפיצ'רים של התמונה המשוחזרת לבין הפיצ'רים של התמונה המקורית. אבל אלו פיצ'רים ניקח בשביל השוואה הזו? הרי המטרה היא למדוד את ״רמת פוטוריאליסטיות״ של התמונה המשוחזרת אז הפיצ'רים צריכים לשקף את ״המאפיינים החשובים״ של התמונה המקורית. כדי להפיק פיצ'רים כאלו בדרך כלל לוקחים רשת מאומנת כמו [VGG](https://keras.io/api/applications/vgg/) או [ResNet50](https://viso.ai/deep-learning/resnet-residual-neural-network/), ומחשבים מרחק (בד"כ L2) בין פלטים של השכבות שלהן עבור התמונה המקורית למשוחזרת.

אציין שהלוס של גאן מחושב כממוצע על פני כל הפאצ'ים של תמונה בדומה ל-[PatchGAN](https://arxiv.org/pdf/1611.07004.pdf). זה כופה על התמונה המגונרטת לא רק להיות כמה שיותר "דומה לאמיתית כמקשה אחת" אלא דורשת שדמיון זה יתקיים בכל פאץ'.

אני מאמין שסכום של שני לוסים אלו מאפשרים למפענח של VQ-VAE ליצור תמונות מרהיבות.

**״למידת״ מרחב לטנטי של VQ-GAN:**

אוקיי, הצלחנו להפיק ייצוג חזק מתמונה המהווה מערך של וקטורים מה-codebook (כל וקטור מיוצג ע״י המספר הסידורי שלו) כאשר כל וקטור מהווה ייצוג של פאץ' של התמונה. בשלב השני של אימון VQ-GAN המטרה היא לאמן מודל ליצירה של ייצוגים אלו (סדרות של ייצוגי פאצ'ים). כך נוכל להשתמש במודל זה לגנרוט של ייצוג לטנטי של תמונה שמוזן לאחר מכן למפענח ליצירת תמונה.

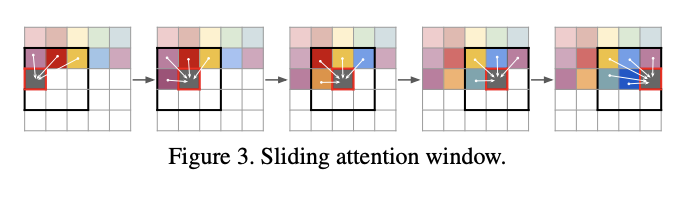
איך עושים זאת? מאחר וניתן ליצור תמונה באופן אוטורגרסיבי (פאץ' אחרי פאץ') מאמנים מפענח של הטרנספורמר (המאמר השתמש בארכיטקטורה דומה לזו של [GPT2](https://openai.com/blog/better-language-models/)) בשביל לחזות ייצוג לטנטי (מספרו הסידורי ב-codebook) של פאץ' הבא בהינתן כל הפאצ'ים שכבר גונרטו. במשימה פאצ'ים של תמונה הם ״משחקים תפקיד של טוקנים״ של משימות של השפה הטבעית.

מעשית **לאחר סיום אימון של VAE בשלב הראשון**, לוקחים את כל הייצוגים הלטנטיים של התמונות מהדאטסט ומאמנים דקודר של הטרנספורמר לחזות ייצוג של פאץ' בהינתן הפאצ'ים הקודמים

**הערה:** לפני תחילת שלב האימון השני, ״מקפיאים״ את כל הפרמטרים של המקדד, המפענח ואת ה-codebook.

**״תיבול״ של inductive bias:**

רגע, אבל מה עם התבלין מסוג inductive bias שהבטחתי קודם לכן? המחברים מצאו כי שימוש בפאצ'ים גדולים מ-16x16 פוגע בביצועים של המודל. מצד שני עקב משאבי החישוב המוגבלים שעמדו לרשותם, הם לא הצליחו לאמן טרנספורמר עבור יותר מ-256 פאצ'ים. איך יוצאים מהמצב הזה ומגנרטים תמונות גדולות יותר מ-256x256? פשוט משתמשים רק בפאצ'ים הקרובים לפאץ' הנחזה - והנה קיבלתם ה-inductive bias המובטח :)

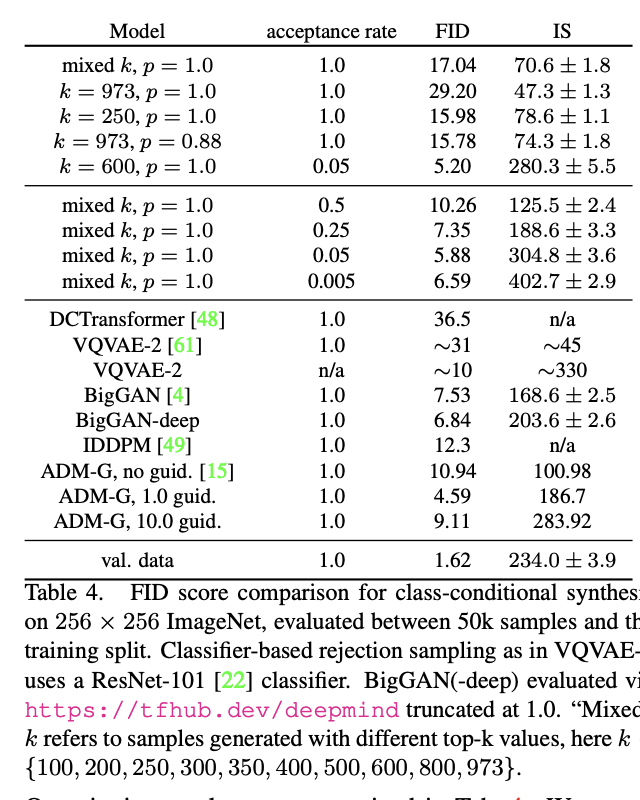


**סיכום קצר של שלבי אימון VQ-GAN:**

* מאמנים VQ-VAE כאשר פונקצית הלוס היא שילוב [הלוס הסטנדרטי](https://developers.google.com/machine-learning/gan/loss) של גאן והלוס הפרספטואלי ([perceptual loss](http://svl.stanford.edu/assets/papers/JohnsonECCV16.pdf)).
* מקפיאים את כל הפרמטרים של כל הרשתות שאומנו בשלב הראשון
* לוקחים את כל הייצוגים הלטנטיים של התמונות מהדאטהסט
* מאמנים מפענח של הטרנספורמר לחיזוי הוקטורים הלטנטיים מהשלב הקודם
* **מגנרטים תמונות:** יוצרים ייצוג לטנטי של תמונה באמצעות מפענח מאומן של הטרנספורמר ומעבירים ייצוג זה דרך המפענח של VAE שאומן בשלב הראשון **ונותר ללא שינוי מאז**

**הישגי מאמר:**

המחברים הראו שיפור מבחינת (Frechet Inception Distance (FID ו- (Inception Score (IS מול מודלים גנרטיביים חזקים עבור כמה דומיינים ורזולוציות.



**נ.ב.**

ממש אהבתי את המאמר כי הוא משלב גישות מאוד מעניינות בלמידה עמוקה: VQ-VAE, GAN וטרנספורמרים וגם מנצל את ה-inductive bias הקיים בדומיין הויזואלי. מומלץ בחום רב!

## Review 63: Soft-IntroVAE: Analyzing and Improving the Introspective Variational Autoencoder

**פינת הסוקר:**

**המלצת קריאה ממייק**: מומלץ לאוהבי מודלים גנרטיביים

**בהירות כתיבה**: גבוהה.

**ידע מוקדם**:

* הבנה של עקרונות [VAE](https://arxiv.org/abs/1312.6114) (למשל שימוש ב- ELBO לשערוך של הנראות המירבית).
* עקרונות אימון של מודלים גנרטיביים עם פונקציית לוס אדוורסארית (משחק mini-max כמו בגאן) - **נא לא לבלבל שיטות אלו עם שיטות לאימון רשתות ״חסינות״ נגד התקפות אדוורסריות (adversarial examples)**.
* ידע בהסתברות אם אתם רוצים להבין את הפרקים עם ההוכחות.

**יישומים פרקטיים אפשריים**: יצירה (אינפרנס) של תמונות באיכות גבוהה בסיבוכיות חישובית של VAE רגיל.

**פרטי מאמר:**

**מאמר:** [זמין להורדה](https://arxiv.org/pdf/2012.13253.pdf).

**קוד**: [כאן](https://github.com/taldatech/soft-intro-vae-pytorch)

**פורסם בתאריך**: 25.03.21, בארקיב.

**הוצג בכנס**: CVPR 2021

**תחומי מאמר:**

* מודלים גנרטייביים ליצירת דאטה ויזואלי (תמונות)

**כלים מתמטיים, מושגים וסימונים:**

* [ELBO](https://mbernste.github.io/posts/elbo/)
* [VAE](https://arxiv.org/abs/1312.6114)
* אימון מודלים גנרטיביים עם פונקציית לוס אדוורסרית

**מבוא:**

****

היום אני סוקר עוד מאמר המציעה גישה לשיפור של איכות התמונות הנוצרות באמצעות מודל גנרטיבי מסוג VAE. אחת נקודת התורפה של VAE היא הקושי שלהם לגנרט תדרים גבוהים הנובע בין השאר ממבנה של איבר השחזור (reconstruction term) המופיע בפונקציית הלוס שלה. איבר השחזור מכיל מרחק L2 בין התמונה המקורית למשוחזרת, המתקבלת באמצעות העברת התמונה המקורית דרך האנקודר והדקודר. מזעור איבר זה גורם לתמונות המשוחזרות להיות דומות למקוריות בממוצע שמתבטא ביצירת תמונות בעלות אזורים חלקים ונטולי פרטים (תדרים גבוהים). ניתן לנסח את הבעיה באופן הבא: **VAE יודע "לתת" הסתברויות גבוהות לדגימות אמיתיות, אך מתקשה "לתת" הסתברויות נמוכות לדגימות מטושטשות.** מכיוון שגאנים מודרניים כבר לא סובלים מבעיה זו (יש לגאנים חסרונות אחרים כמו מוד קולפס ואימון לא יציב) אז נראה לכאורה ששילוב של גאן ו-VAE עשוי לתת מענה לסוגיית התדרים הגבוהים של VAE.

ב[סקירתי האחרונה](https://machinelearning.co.il/11009/vq-gan) תיארתי מודל שהוא שילוב של VAE וגאן, הנקרא VQ-GAN. המאמר הנסקר אז הציע להוסיף ל-VAE רשת הדיסקרימינטור D במטרה להבחין בין התמונה המקורית לבין התמונה המשוחזרת (בדומה לגאנים). כלומר המחברים "הוסיפו גאן" ל- VAE כאשר רשת הדקודר משחקת תפקיד של גנרטור של גאן. שילוב של הלוס האדוורסרי הסטנדרטי של גאן בפונקציית לוס של VQ-GAN הצליח להתגבר על בעיית ה״תדרים הגבוהים״ בתמונות הנוצרות. למעשה רשת הדיסקרימינטור **מאומנת להבחין בין התמונות המשוחזרות לבין המקוריות** וזה מאפשר ל- VQ-GAN לגנרט תמונות פוטוריאליסטיות מדהימות.

**תמצית מאמר:**

כמובן שהוספת רשת נוספת ל-VAE מקשה על האימון של VQ-GAN. זה מוביל אותנו לשאלה הבאה: האם ניתן להשיג את היתרון של הלוס האדוורסרי של גאן מבלי לשנות את המבנה המקורי של VAE? מתברר שהתשובה על השאלה הזו היא חיובית וזה בדיוק מה שעשה המאמר שאסקור הפעם.

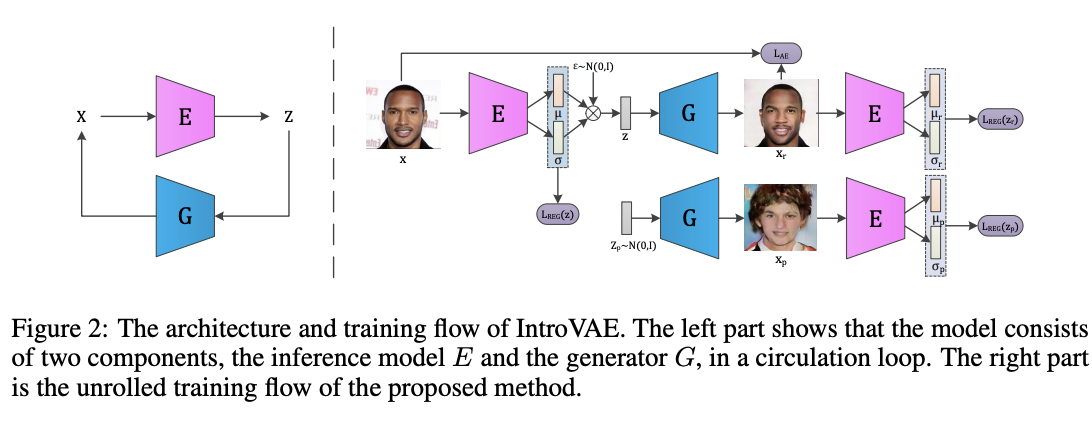
המאמר הנסקר מציע שיטה שכלשונו מאפשרת "לאמן את VAE בצורה introspective, כלומר VAE עצמו מאומן לחפש הבדלים בין תמונות מגונרטות באמצעותו לבין התמונות המקוריות". כדי לעשות זאת רשת **האנקודר לובשת כובע של הדיסקרימינטור**, כאשר **הדקודר משחק תפקיד של הגנרטור**. בפרק הבא נתאר איך ניתן לבנות פונקציית לוס המשלבת את הלוס האדוורסרי של גאן עם הלוס של VAE.

**תקציר מאמר:**

כעת נתאר את המבנה של פונקציית לוס של Soft-IntroVAE. קודם כל נציין שהמאמר הנסקר הוא שיפור של מאמר, המציע מודל בשם [IntroVAE](https://arxiv.org/pdf/1807.06358.pdf) שלראשונה הצליח להכניס גאן ל-VAE ללא תוספת של רשתות נוספות.

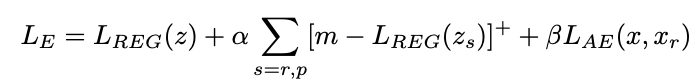
נתחיל מתיאור של פונקציית לוס של IntroVAE. להבדיל מ- VQ-GAN ״ההבחנה״ בין תמונות מגונרטות למקוריות נעשית לא במרחב המקורי (של תמונות) **אלא במרחב הלטנטי**. הרעיון העיקרי של IntroVAE הוא לבנות משחק מיני-מקס בין הדיסקרימינטור (הנקרא כאן אנקודר E) לגנרטור (אשר בנוי כדקודר אבל כאן נקרא G) כאשר:

* **דיסקרימינטור** מצד אחד מנסה למזער את מרחק KL בין התפלגות הייצוגים הלטנטיים של תמונות מקוריות לבין הפריור (בדרך כלל גאוסי איזוטרופי עם מטריצת קווריאנס c**I**). מצד שני הוא מנסה להגדיל את מרחק DL בין התפלגות הייצוגים הלטנטיים של התמונות המגונרטות לבין התפלגות הפריור.
* **גנרטור** מצידו ינסה ״לקמבן״ את הדיסקרימינטור תוך מזעור של מרחק DL בין התפלגות הייצוגים הלטנטיים של התמונות המגונרטות לבין התפלגות הפריור.



אוקיי, אבל איך זה נעשה בפועל. איך ניתן לשלב בין הלוס הרגיל של VAE לבין הלוס האדוורסרי שמוגדר למעלה? האמת שהמאמר בחר בגישה די אינטואיטיבית. פונקציית לוס של האנקודר (דיסקרימינטור) מורכבת מסכום של ו- כאשר:

**:** הלוס הרגיל של VAE (למעשה זה הלוס של [β-VAE](https://deepmind.com/research/publications/2019/beta-VAE-Learning-Basic-Visual-Concepts-with-a-Constrained-Variational-Framework) כאשר איבר KL בא עם מקדם β) ומרחק KL בין התפלגות הפריור (p(z לבין ייצוגים לטנטיים של תמונות מגונרטות. מתמטית הלוס נראה באופן הבא:



כאן:

1. מסמן מרחק KL ממוצע בין התפלגות של וקטור zg המוווה ייצוג לטנטי של תמונה משוחזרת לבין התפלגות פריור של וקטור לטנטי של תמונה מהדאטהסט (כלומר התפלגות גאוסית (N(0,c**I.**
2. .שימו לב כי בביטוי עבור מופיע סכום של ו- כאשר ו- הינם ייצוגים לטנטיים של תמונות מגונרטות המתקבלות באופן קצת שונה - הסבר לגביהם יינתן בהמשך.
3. הוא לוס השחזור בין תמונה x לבין גרסתה המשוחזרת .
4. הם הייפר-פרמטרים ו-

: מורכב מסכום של לוס השחזור אותו מרחק KL בין ייצוגים של תמונה מגונרטת ולזה של תמונה אמיתית:



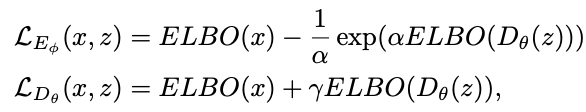
כאן מסמן את התוצאה של העברת התמונה באנקודר.

המאמר מציע שתי דרכים לבנייה של ייצוגים לטנטיים ו- של תמונות מגונרטות:

1. לוקחים תמונה משוחזרת (לאחר העברתה של תמונה מהדאטהסט דרך האנקודר ולאחר מכן דרך הדקודר) ומזינים אותה לאנקודר כדי לבנות את הייצוג הלטנטי.
2. דוגמים מהפריור (p(z וקטור z ומעבירים אותו דרך הדקודר ולאחר מכן דרך האנקודר ויוצרים ייצוג לטנטי .

כעת נדון בחולשה המרכזית של IntroVAE שאותה Soft-IntroVAE בא לתקן. המחברים של המאמר הנסקר טוענים כי **אימון של IntroVAE עלול לסבול מאי-יציבית עקב רגישותו הרבה לבחירת פרמטר m (מהלוס ).** בנוסף, הניתוח התאורטי של משחק מינמקס שהוצג ב- IntroVAE לא לקח בחשבון חלק מהאיברים של פונקציית הלוס והסתפק רק בניתוח של איברים המכילים מרחקי KL.

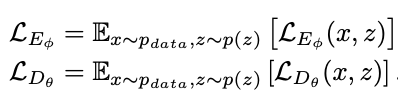
כדי להתגבר על הקושי, הם מציעים להחליף את מרחק KL מהלוסים ו- לביטוי המלא של [ELBO](https://mbernste.github.io/posts/elbo/) עבור התמונה המגונרטת. כלומר פונקציות לוס עבור הדיסקרימינטור (אנקודר) והגנרטור (דקודר) מקבלות את הצורה הבאה:



כאשר כלומר התפלגות אפוסטריורית של תמונה x בהינתן ייצוג לטנטי z, ו- הם שני הייפר-פרמטרים חיוביים. שימו לב כי משוואה זו מבטאת את ה״משחק״ בין האנקודר ודקודר כאשר בדומה ל- IntroVAE,

* האנקודר מאומן להבחין בין תמונות אמיתיות (ELBO נמוך) לתמונות הנוצרות באמצעות הדקודר (ELBO גבוה),
* הדקודר מנסה ״לעבוד על״ האנקודר ומנסה לגנרט פיסות דאטה כמה שיותר משכנעות.

מבחינה מתמטית פונקציית הלוס של Soft-IntroVAE מקבלת את הצורה הבא:



כאשר היא התפלגות של דאטה אמיתי ו- הינה התפלגות פריור גאוסית איזוטרופית. מעניין כי להבדיל מ- IntroVAE, המשחק בין האנקודר לדקודר מחושב במרחב המקורי של הדאטה (תמונות) ולא במרחב הלטנטי. בעיית אופטימיזציה זו פותרים בדומה ל-VAE הרגיל באמצעות Gradient Descent ו[טריק רפרמטרזיציה (reparametrizarion trick)](https://gregorygundersen.com/blog/2018/04/29/reparameterization/)**.**

שימו לב כי הנוכחות של האיבר בביטוי של בעיית האופטימיזציה של האנקודר, משעמותו מינימיזציה של עבור דגימות המגונרטות באמצעות הדקודר. ניזכר כי מהווה חסם תחתון על הנראות המירבית של (תמונות מגונרטות). כלומר מינימיזציה של עלולה לפגוע ב-״איכות״ של התמונות המגונרטות.

אם זה אכן המקרה, זה כמובן מאוד בעייתי. אז בואו נבין מה קורה כאן? קודם כל המאמר מוכיח כי [שיווי משקל של נאש](https://corporatefinanceinstitute.com/resources/knowledge/economics/nash-equilibrium-game-theory/)של בעיית האופטימיזציה של Soft-IntroVAE פותר מתכנס ל- כאשר:



כאשר היא פונקציית האנטרופיה של התפלגות q. כלומר עבור הפתרון אינו מתכנס להתפלגות של הדאטה אלא מהווה פתרון של בעיית אופטימיזציה עם איבר רגולריזציה שמעודד פתרונות בעלי אנטרופיה גבוהה מספיק. אבל האם זה טוב? המאמר טוען כי:

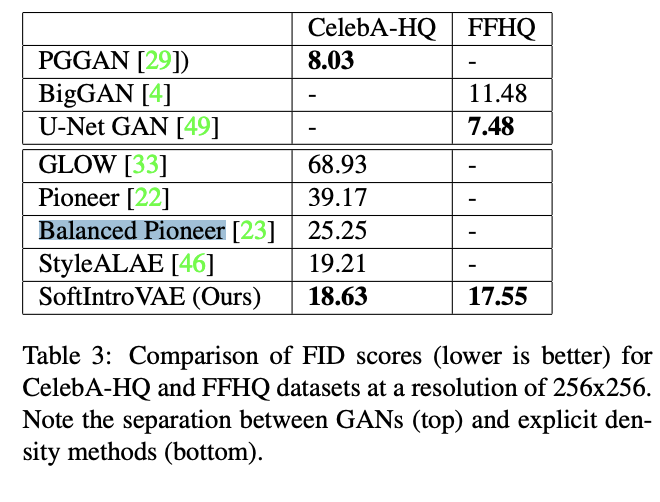
“Soft-IntroVAE learns distributions with sharper supports than a standard VAE, but without negative effects such as mode dropping”

על כמה דאטסטים פשוטים (toy datasets). על הדאטהסטים של תמונות ניתן לראות כי איכות התמונות המגונרטות מאוד גבוהה והתמונות עצמן די מגוונות.

**מימוש:** המאמר השתמש במודיפיקציה של ELBO שהוצע ב- [β-VAE](https://deepmind.com/research/publications/2019/beta-VAE-Learning-Basic-Visual-Concepts-with-a-Constrained-Variational-Framework).

**הישגי מאמר:**

המחברים הראו כי הגישה המוצעת מצליחה להשיג ביצועים יותר טובים מכמה שיטות גנרטיביות אחרות כמו [StyleALAE](https://arxiv.org/abs/2004.04467), [GLOW](https://arxiv.org/abs/1807.03039), [Balanced Pioneer](https://arxiv.org/abs/1904.06145) וכמה אחרים על שני דאטהסט [CelebA-HQ](https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html) ומראה תוצאות קרובות לגאנים עבור דאטהסט [FFHQ](https://github.com/NVlabs/ffhq-dataset)**.** כמובן הגאנים עדיין מנצחים את השיטות המבוססות VAE.



**נ.ב.**

מאמר עם רעיון מעניין, יש ניתוח מתמטי רציני של המודל, איכות התוצאות מרשימה. חסרות לי השוואות ביצועים עם עוד מודלים חזקים מבוססי VAE עבור דומיינים נוספים. אבל בסך הכל המאמר מומלץ בחום!

## Review 64: PDE-GCN: Novel Architectures for Graph Neural Networks Motivated by Partial Differential Equations

**פינת הסוקר:**

**המלצת קריאה ממייק ומעידו**: מומלץ מאוד למי שרוצה להבין האם יש קשר בין משוואות דיפרנציאליות חלקיות (PDE) ובין רשתות נוירונים גרפיות (GNN)

**בהירות כתיבה**: בינונית פלוס

**ידע מוקדם**: הבנה בסיסית ב-PDE וב-GNN ובנוסף היכרות עם מושגי יסוד בתורת הגרפים במשוואות דיפרנציאליות.

**יישומים פרקטיים אפשריים**: פתרון של בעיות בביולוגיה חישובית, עיבוד תמונה, גרפיקה, ראיה, ניתוח רשתות חברתיות ועוד.

**פרטי מאמר:**

**לינק למאמר:** [זמין להורדה](https://arxiv.org/pdf/2108.01938.pdf).

**לינק לקוד**: ---

**פורסם בתאריך**: 26.10.21, בארקיב.

**הוצג בכנס**: NeurIPS 2021.

**מקום השתייכות של המחבר הראשון**: אוניברסיטת בן-גוריון בנגב.

**תחומי מאמר:**

* [רשתות נוירונים על גרפים (GNN)](https://distill.pub/2021/gnn-intro/)
* אנליזה נומרית של משוואות דיפרנציאליות חלקיות
* פיזיקה חישובית (computational physics)

**כלים מתמטיים, מושגים וסימונים:**

* [רשתות קונבולוציה על גרפים](https://distill.pub/2021/gnn-intro/)
* [החלקת יתר (over-smoothing) ברשתות נוירונים על גרפים](https://arxiv.org/abs/1909.12223)
* אופרטורים דיפרנציאליים כמו דיברגנץ, לפלסיאן
* רשתות קונבולוציה קלאסיות (CNNs)

**תמצית מאמר:**

אחת הסוגיות המרכזיות ברשתות קונבולוציה על גרפים (GCN) הינה החלקת יתר (over-smoothing) של ייצוג דאטה המופק באמצעות הרשת. החלקת יתר של ייצוג מאופיינת בשינויים הולכים וקטנים של בין ייצוג של פיסות דאטה שונות (כגון embeddings של קודקודים וקשתות) המופק באמצעות GCN. בעיה זו מחריפה בשכבות העמוקות של GCN כלומר ייצוגי קודקודים וקשתות בשכבות אלו נהיים זהים זה לזה. תופעה זו היא הסיבה העיקרית לכך שכותבי מאמרים בתחום ה-GCN נוטים להסתפק בכמות קטנה של שכבות לעומת רשתות הקונבולוציה הקלאסיות (CNNs). חולשה נוספת של GCNs היא הצורך בהתאמת הארכיטקטורה שלהן לדומיין ולמשימה. לדוגמה, כותבי המאמר PDE-GCN מציינים כי GCN, המבצעת משימת סיווג בענן נקודות, עלולה להפגין ביצועים ירודים במשימת סיווג של קודקוד בגרף ציטוטי מאמרים (citation network).

המאמר המסוקר מציע גישה מעניינת ומקורית הבא לתת מענה לסוגיות אלו. השיטה המוצעת מנצלת את הקשר הקיים בין רשתות נוירונים לבין משוואות דיפרנציאליות חלקיות (PDE). קשר זה נחקר בצורה אינטנסיבית בתקופה האחרונה ([1](http://arxiv.org/abs/1804.04272), [2](http://arxiv.org/abs/1705.03341), [3](http://arxiv.org/abs/2103.15419), [4](http://arxiv.org/abs/1811.09885), [5](https://arxiv.org/abs/2109.00095)). הדינמיקה של מפות הפיצ'רים לאורך שכבות של CNN ניתנת לתיאור באמצעות מערכת דינמית המתוארת על ידי משוואה דיפרנציאלית חלקית. כלומר ניתן להתייחס לכל שכבה של CNN כאל ״צעד בזמן״ של הגרסה הדיסקרטית של משוואה דיפרנציאלית חלקית. זהו משפט מאוד אבסטרקטי שקושר שני דברים שאנחנו לא רגילים לחשוב עליהם בעת ובעונה אחת, ולכן סקירה זו תנסה להתיר במקצת את הקשר הסבוך שעליו בונים מחברי PDE-GCN את התיאוריה במאמר.

הסקירה תחולק לשלושת החלקים הבאים:

* רקע על ההקבלה בין מד״ח לרשתות קונבולוציה
* מבוא מזורז לרשתות קונבולוציה על גרפים
* החיבור ביניהם והחידוש העיקרי של המאמר.

**חומר רקע:**

**הקשר בין רשתות למשוואות דיפרנציאליות:**

ננסה כעת להסביר (קצת בנפנופי ידיים - אם אתם רוצים הסבר ריגורוזי תעיפו מבט [PDEs and Convolutions](https://mitmath.github.io/18337/lecture14/pdes_and_convolutions) או באחד המאמרים המקושרים בפסקה הקודמת) איך PDEs קשורים ל-CNNs. נתחיל מרשת נוירונים די בסיסית ונראה איך ניתן ״להפוך״ אותה ל-PDE באמצעות מניפולציות די פשוטות. נניח שיש לנו רשת המורכבת מ-T שכבות בעלות חיבור שיורי (residual connection). העיקרון שנדון בו כאן רלוונטי לכל CNN, אבל ההסבר בהיר יותר כאשר מתייחסים אל ארכיטקטורה מבוססת ResNet, ובכל מקרה זו ארכיטקטורה סטנדרטית של רשת קונבולוציות.

כאמור, פלט של שכבה ניתן לתיאור[[1]](#footnote-0) כ:

כאשר הן מטריצות קונבולוציה, ו- הינה פונקציית אקטיבציה לא לינארית כמו סיגמואיד או ReLU. כעת, אם נעביר איבר אחד שמאלה נקבל את המשוואה הבאה:

ואם נכליל את המשוואה הזו ונחליף את המקדם 1 של צד ימין בקבוע :

אז צד שמאל של המשוואה האחרונה מזכיר את הקירוב מסדר ראשון של בשיטת ההפרשים הסופיים (finite differences). זהו פתח ראשוני אל עולם הקשרים בין מד״ח (או מישדיפ, אם תתעקשו 😉) ללמידה עמוקה.

כעת נותר לנו רק למצוא אנלוגיה לנגזרות המרחביות, שבלעדיהן אף משוואה דיפרנציאלית אינה מעניינת. לצורך כך, נתחיל מהכיוון השני ונתבונן תחילה במשוואת הדיפוזיה הקלאסית:

כאשר הינה מטריצה סימטרית וחיובית-לחלוטין (positive definite) כלשהי, הוא אופרטור הלפלסיאן ו- הם הדיברגנץ[[2]](#footnote-1) והגרדיאנט[[3]](#footnote-2), בהתאמה. נעניק לכל אחד מאיברי המשוואה את הגרסה הדיסקרטית שלו ונקבל:

וכדי לראות את הקשר למשוואת הרזנט (ResNet) שלנו, נכליל גם את משוואה זו ונגדיר כמקרה פרטי. אם נוסיף את פונקציית האקטיבציה במקום הנכון, קיבלנו בדיוק את משוואת הרזנט.

לא ניכנס כאן לפרטים הטכניים של האנלוגיה הזו ונסתפק בנפנופי ידיים, אבל על בסיס זה מרשה לעצמו המאמר להתייחס לרשת CNN בתור הכללה של משוואת דיפוזיה, כאשר הקונבולוציות משחקות תפקיד של ״אופרטורים דיפרנציאליים״ נלמדים אשר מותאמים לדאטה שעליו מתאמנת הרשת ע״י תהליך האימון.

**יסודות של GCN:**

רשת נוירונים לגרף מגיעה מהצורך לעבד סיגנלים שחיים בעולם לא-סדור (unstructured). ניתן לראות כי גרף הוא הכללה של תמונה, בו הפיקסלים במרווחים לא קבועים ומספר השכנים משתנה גם הוא. כך ניתן לראות רשת נוירונים על גרף בתור הכללה של רשת נוירונים סטנדרטית המותאמת לסוגי דאטה לא-סדורים. הפעם אנחנו מתעניינים ברשת Graph Convolutional Network - GCN, רשת אשר פועלת על גרף באמצעות קונבולוציות.

המטרה של רשת כזו היא ללמוד ייצוגים של הקודקודים והקשתות בגרף באמצעות העברת אינפורמציה מקודקודים וקשתות אחרים בגרף, בדרך כלל תוך התחשבות בקישוריות אשר נתונה לנו בגרף (ועם זאת, יש מודלים שממציאים קשתות או לומדים לחזות חלקים חדשים מהגרף, הכל לפי צורכי המשימה). כלומר, בהינתן ייצוג התחלתי לכל קודקוד וקשת בגרף, מעדכנים את הייצוג שלו על ידי הזרמת מידע מהקודקודים השכנים. העדכונים האלה בדרך כלל מתבססים על פילטרים ואופרטורים נלמדים, כמו ברשת CNN רגילה.

**תקציר מאמר:**

כמו שכבר אמרנו בתחילת הסקירה הגישה המוצעת באה להתמודד עם החלקת יתר של הפיצ'רים בין קודקודים וקשתות בשכבות העמוקות של GCN. המחברים מציעים ארכיטקטורה של GCN המבוססת על דיסקרטיזציה של **משוואה היפרבולית לא-לינארית** (כמובן, עם תנאי התחלה עבור ותנאי שפה שלא נעסוק בהם כאן):

המאמר מוכיח כי פתרון של משוואה זו לא גורם לשחיקה עבור ערכים גבוהים של t (נזכר כי t הוא למעשה מספר השכבה ב-GCN בגרסה הדיסקרטית של המשוואה). המאמר גם מראה כי PDE המתארת GCN סטנדרטי הינה **משוואת דיפוזיה לא-לינארית**:

למי שאינו מנוסה במד״ח והמשוואות נראות לו זהות - שימו לב לצד שמאל של שתי המשוואות. צד ימין אכן זהה. המחברים מראים כי הפתרון של המשוואה האחרונה מכיל מעט מאוד מידע מאחר ודינמיקת הערבוב של משוואת הדיפוזיה מיצעה את כל הפיצ'רים, והתופעה הזו חמורה יותר ככל שמספר השכבות גדל. לטענת המאמר זו הסיבה העיקרית לתופעת החלקת היתר המתרחשת בשכבות העמוקות של GCN סטנדרטיות.

כדי להתגבר על החלקת היתר הזו, המאמר מציע לבנות ארכיטקטורה חדשה של GCN הנקראת PDE-GCN, בהתבסס על **המשוואה ההיפרבולית ולא על משוואת הדיפוזיה כמו GCN סטנדרטי.** בנוסף המחברים מגדירים גירסה דיסקרטית של הגרדיאנט המרחבי של הגרף: עבור שני קודקודים i ו-j שמחוברים בקשת, הגרדיאנט ביניהם מוגדר כהפרש של וקטורי הפיצ'רים (ייצוגי הקודקודים עבור השכבה הנוכחית) ו- המוכפלים במשקולות כלשהם (אשר יכולים להיות נלמדים או מהונדסים). נציין כי הוא למעשה מיפוי ממרחב הקודקודים V למרחב הקשתות של הגרף E.

את אופרטור הדיברגנץ (), המופיע במשוואה ההיפרבולית ניתן לקרב באמצעות (הדיברגנץ על גרף בדרך כלל מוגדר כמיפוי ממרחב הקשתות E למרחב הקודקודים V וכאן אנו פשוט משחלפים את שהיא מיפוי מ- V ל-E) עכשיו רק נותר להפעיל את שני האופרטורים ברצף כדי לקבל את הביטוי עבור האגף הימני של המשוואה ההיפרבולית. אחרי שהגדרנו את כל המשתנים ניתן לבנות את הארכיטקטורה הבסיסית של שכבת PDE-GCN:

כאשר היא מטריצת קונבולוציה 1x1 נלמדת ופונקציית האקטיבציה שנבחרה היא tanh.

אחרי שהגדרנו את המבנה של PDE-GCN נותר לנו להסביר איך כל העסק עובד בפועל. באמת שזה לא מסובך:

* לוקחים את הפיצ'רים של הקודקודים והקשתות,
* מעבירים אותם דרך שכבת קונבולוציות 1x1,
* מחשבים את הפלט של השכבות הבאות בהתבסס על המשוואה האחרונה.

**הערה 1:** המאמר גם מציע דרך לנצל פיצ׳רים על קשתות (אם הם זמינים) כקלט ל-PDE-GCN.

**הערה 2:** המאמר גם מציע שיטה לבניית ארכיטקטורה של PDE-GCN המבוססת על שילוב (צירוף קמור בגדול) של משוואות הדיפוזיה והמשוואה ההיפרבולית.

**הישגי מאמר:**

המאמר השווה את ביצועי PDE-GCN עבור ערכים שונים של מספר שכבות עם מגוון של GCN-ים על כמה משימות ודאטהסטים שונים. המטרה הייתה להראות כי הארכיטקטורה המוצעת מצליחה להתגבר על בעיית החלקת יתר של הפיצ'רים בשכבות העמוקות של GCN. הדרך הטבעית להוכיח זאת היא להראות כי לא נצפית ירידה בביצועי GCN כאשר מוסיפים לה שכבות (כמובן שאם החלקת יתר עדיין קיימת, הוספת שכבות לרשת עלול לפגוע בביצועים). המאמר אכן מראה כי בכל המשימות שנבחנו ביצועי PDE-GCN לא סופגים ירידה (אלא משתפרים קצת ברוב המקרים). בחלק מהמשימות המאמר אפילו מציג SOTA חדש, אך לא בכולן.

כאן צריך לציין שבהשוואה לרוב הארכיטקטורות האחרונות של GCN, הביצועים לא תמיד טובים יותר והרשתות מהשנים האחרונות יודעות להתמודד בהצלחה גם עם הרבה שכבות (במאמר נבדקו עד 64). המאמר למעשה מציג שיטה נוספת (ופשוטה יחסית) לתכנן רשתות GCN עמוקות (לעומת [vanilla GCN של Kipf and Welling מ-2016](https://arxiv.org/abs/1609.02907) ו-[Dropedge](https://arxiv.org/abs/1907.10903)).

**נ.ב.**

המאמר מציע גישה חדשנית ומעניינת לבניית ארכיטקטורה של GCN באמצעות משוואות דיפרנציאליות חלקיות. הגישה מצליחה להתגבר על בעיית החלקת יתר המתרחשת כאשר מוסיפים שכבות לרשת. לדעתנו, זאת גישה מעניינת לתכנון ארכיטקטורות דיפ בכללי ו-GCN בפרט ואנו רואים במאמר זה (ובסקירה זו) פתח לתוך עולם מעניין, אך פחות מוכר. בראיה זו, סקירה היא הראשונה מבין מספר סקירות מאמרים בעולם האנלוגיות בין מד״ח ובין ארכיטקטורות דיפ.

#deepnightlearners

**שיתוף פעולה:** הפוסט נכתב על ידי [מיכאל (מייק) ארליכסון, PhD](https://www.linkedin.com/in/michael-mike-erlihson-8208616/), Michael Erlihson ועדו בן-יאיר.

## Review 64: TRAIN SHORT, TEST LONG: ATTENTION WITH LINEAR BIASES ENABLES INPUT LENGTH EXTRAPOLATION

כמו שאתם יודעים אחת החולשות העיקריות של הטרנספורמר היא הסיבוכיות הריבועית שלו ביחס לאורך סדרת הקלט. יצאו מאות מאמרים המציעים וריאנטים של הטרנספורמר עם סיבוכיות נמוכה יותר.

המאמר הזה לוקח כיוון די מפתיע בהתמודדות עם סוגיית הסיבוכיות הריבועית. הוא שואל את השאלה הבא - אולי נאמן טרנספורמר לאורכי קלט לא גדולים ואז נריץ אותו עם קלטים ארוכים יותר באינפרנס? זה נשמע מגניב אבל

האם זה בכלל יעבוד? באופן לא מפתיע התשובה טמונה בקידודים מיקומיים (positional encodings -PS) שמשתמשים בהם בשביל להעביר לטרנספורמר מידע עם מיקום של טוקנים בסדרה.

מתברר שלאחרונה יצאו מספר מחקרים המציעים PS שונים לשיפור ביצועי טרנספורמר (ולאו דווקא למטרה המתוארת לעיל). אז המאמר הזה מציע שני שינויים ל- PS המאפשרים לטענות ״לאמן קצר ולהריץ ארוך״. במקום לחבר את PS לייצוג הטוקנים כמו שעשו בטרנספורמר האמיתי, המחברים כאן מציעים לעשות את הדבר הבא:

1. מחשבים מכפלה פנימית של וקטורי query ו- key.
2. מחברים למכפלה הזו את המרחק השלילי בין הטוקנים, מוכפל בקבוע m (שהוא שונה בכל head של הטרנספורמר). למשל למכפלה פנימית של טוקן 3 ו- 8 מחברים (5m-).
3. ובונים מכפלה בין וקטורי value עם המקדמים שחישבנו בסעיפי 1-2 (למעשה השינוי הזה הוצע במאמר אחר). ההגיון כאן שכל המידע על מיקום כבר מקודד ב- key ו-query ואין צורך להוסיף אותו ל- value.

**אינטואיציה:** למעשה הקידוד הזה אומר את הדבר הבא: בוא **נקטין** את הקשר (מקדם רלוונטיות) בין טוקנים ככל שמרחק בין הטוקנים גדל. המקדם m השונה בין ראשי הטרנספורמר מאפשר לשלוט במידת ״ההקטנה״ של הרלוונטיות.

וזה כל הסיפור. שימו לב שהמאמר מציע את ה- PS שלהם רק עבור טרנספורמר אוטורגרסיבי (שנקרא decoder) אבל אני לא רואה שום סיבה לא להשתמש בזה בצורה דו-כיוונית.

ה- PS המוצע מאפשר להאט את הירידה בביצועים בתרחיש שבו מאמנים טרנספורמר לאורך קלט נתון (512) ואז עושים אינפרנס לאורכי קלט כמו 1024, 2048 וארוכים יותר (יחסית ל- PS האחרים כולל המקוריים של הטרנספורמר).

מאמר: <https://arxiv.org/pdf/2108.12409.pdf>

קוד: <https://github.com/ofirpress/attention_with_linear_biases>

יניק: <https://www.youtube.com/watch?v=-Kgxv64aG3o>

## Review 66: VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning

מאמר מעניין של Yann LeCunn et al המציע שיטה לבנייה של ייצוג דאטה לא מתויג (self-supervised). המחברים עצמם קוראים לשיטה המוצעת ״ridiculously simple" והיא בהחלט עונה על ההגדרה הזו.

מה התכונה החשובה ביותר שאנחנו רוצים מייצוג של דאטה? אנו רוצים שייצוגים של פיסות דאטה דומות יהיו קרובים במרחב הייצוג (אוגמנטציות של אותה תמונה למשל). בנוסף רצוי מאוד שהייצוגים של פיסות דאטה לא דומות יהיו רחוקים. כדי להשיג את התכונה הראשונה (קרבת הייצוגים של דוגמאות קרובות) ניתן לאמן את הרשת עם פונקציית לוס הממזערת מרחק בין ייצוגים של דוגמאות קרובות. אבל אימון עם פונקציית לוס כזו עלול להוביל לכך שכל הייצוגים יהיו זהים ואז התכונה השניה (ייצוגים רחוקים של דוגמאות לא דומות) לא מתקיימת. אז איך מתגברים על זה?

בגדול יש שתי גישות עיקריות לבניית ייצוגים עבור דאטהסטים לא מתויגים:

* משתמשת בדוגמאות קרובות ורחוקות ודורשת למזער את המרחק בין הייצוגים של דוגמאות קרובות ולמקסם מרחק בין ייצוגים של דוגמאות שליליות (רחוקות). זה נעשה בדרך כלל באמצעות שיטות של Contrastive Loss.
* ממזערת מרחק בין דוגמאות חיוביות (קרובות) ומערבת רגולריזציה לא מפורשת (לא איזה איבר L2), הבאה למנוע התכנסות של כל הייצוגים לאותו וקטור (BYOL, SwAv שאני סקרתי הן דוגמאות קלאסיות של הגישה הזו.

אז מה מציע המאמר של LeCunn? מה השיטה ״ridiculously simple" המוצעת?

למעשה מה אנו רוצים מהייצוגים של הדאטה?

* קרבה של הייצוגים של דוגמאות קרובות
* שונות מספיק גבוהה בכל מימד של וקטורי ייצוג (שלא ייצא 3.2 במימד 68 של כל וקטורי הייצוג)
* קורלציה גבוה בין וקטורי ייצוג של דוגמאות שליליות

פונקציית לוס של VICReg מכילה 3 איברים שמטרת כל אחד מהם להשיג את מה שמופיע לסעיפים 1-3 לעיל. פשוט כך

באופן מפתיע VICReg מצליח להשיג ביצועים קרובים מאוד (הבוחנים את עוצמת הייצוג) לשיטות מתחרות מורכבות הרבה יותר במגוון של משימות.

שתי הערות לסוף:

1. המאמר נדחה NeurlPS 2021 (לא התעמקתי בסיבות)
2. כשקראתי סקירות על בניית ייצוגים לדאטה לא מתויג חשבתי למה לא לעשות את בצורה הכי פשוטה שיש, קרי VICReg. אבל לא טרחתי לממש ולבדוק את זה כי הייתי בטוח שזה לא יעבוד ואחרים בטח בדקו את זה. בדיעבד הייתי מוציא על זה מאמר והוא היה נדחה כמו זה של לקון :)

**לינק למאמר:** <https://arxiv.org/abs/2105.04906>

**פוסט דחיה של לקון:** <https://m.facebook.com/story.php?story_fbid=10157921242067143&id=722677142>

## Review 67: Grokking: Generalization beyond Overfitting on small algorithmic datasets

מאמר די מסקרן של openai שיצא לאחרונה. אסקור אותו קצרות כמו שמקובל ב #shortdeepnightlearners

הטענה העיקרית של המאמר נשמעת לא מסובכת. הרי ידוע שאם נאמן רשת יותר מדי זמן, מלשב מסוים היא תיכנס למוד של overfitting כלומר שגיאת ההכללה שלה תתחיל לעלות. המחברים טוענים שאם נאמן אם נמשיך לאמן את הרשת גם לאחר שהיא נמצאת ב- overfitting מאיזשהו שלב שגיאת ההכללה שלה תתחיל לרדת כלומר הביצועים שלה על טסט (או ולידציה) יתחילו לעלות.

המחברים זיהו תופעות כאלו כאשר הם אימנו רשתות על דאטהסטים ״אלגוריתמיים״ קטנים. מה זה דאטהסט אלגוריתמי ואיזו משימה נותנים לרשת במקרה, אתם שואלים? אחת הדוגמאת הוא מטריצה המתארת פעולות במרחב התמורות (פרמוטציות). כלומר לוקחים תמורות בגודל מסוים (נגיד 120 תמורות עבור n=5) ובונים מטריצה בגודל 120x120 שמשבצת (i,j) מכילה תוצאה של הרכבת תמורה p\_i ו- p\_j. אז משחירים כמה מהמשבmות במטריצה זו ונותנים לרשת ללמוד אותם. זה משהו די מגניב שטרם ראיתי…

עכשיו כמה מילים על התופעה עצמה. התופעה הנצפית היא סוג של double descent, הנקרא epoch-wise (הטרמינולוגיה מ- <https://www.kdnuggets.com/2020/04/double-descent-hypothesis-bigger-models-more-data-hurt-performance.html>). אנו מכירים היטב double descent מהסוג Model-Wise שמשמעותו היא שהגדלה של מספר הפרמטרים לרשת גורמת לתופעה דומה משלב מסוים (שגיאת הכללה עולה ואחר-כך יורדת). מודה שבהתחלה התבלבלתי בעצמי וטענתי התופעה המתוארת במאמר היא לא double descent אך אחרי השיחה עם Liron Itzhaki (תודה רבה על תובנות מאוד מעניינות) הבנתי שיש לי טעות בזיהוי. דרך אגב גם Yannic Kilcher בסרטון שלו טען שזה לא double descent.

למה תופעה כזו מתרחשת. Misha Belkin הגדול חוקר את התופעה המעניינת הזו כבר כמה שנים אבל עדיין אין הסבר מתמטי לכך (תקנו אותי אם אני טועה כאן). לגבי הסברים בנפנופי ידיים יש לי שניים (אחד של לירון ואחד לי):

1. **מרג'ינים**: כאשר ממשיכים לאמן רשת (לסיווג נגיד) כשהיא נכנסה למוד overfitting והלוס ממשיך לרדת לאזור האפס, הרשת עשויה לגלות ״ מודל (המוגדר ע״י הפרמטרים של הרשת) המפריד בצורה רווחת יותר בין הקלאסים״ (עם מרג'ינים גדולים). מודל זה ״חסין״ לרעש בדאטה ומכלילים מספיק טוב לאחר שבהתחלת overfitting היא מצאה פתרונות עם מרג'ינים קטנים שמובילים לשגיאת הכללה גבוהה.
2. **מינימום של פונקצית לוס:** יש הנחה שפתרונות בעלי שגיאת הכללה נמוכה ״נמצאים״ במינימומים בעלי עקמומיות נמוכה (לא חדים) של פונקציית לוס. כלומר כאלו שפונקציית לוס סביבם לא משתנה הרבה וערכיה סביב נקודת מינימום זו היא נמוכים. נקודת מינימום כזו היא יותר טובה ממינימום חד (שאפילו בסביבתו הקטנה פונקציית לוס מקבלת ערכים גבוהים משמעותית מזה בנקודת מינימום). אז יש מצב שמאמנים רשת מספיק איפוקים אז יש סבירות גבוהה להגיע מתישהו לנקודה כזו (לא חדה) שכבר ״קשה״ לצאת ממנה כי פונקציית לוס מקבלת ערכים נמוכים סביבה.

כמובן של הסברים בנפנופי ידיים אבל לדעתי עשויים להיות מועילים לחשיבה על התופעה המעניינת הזו

**לינק:** <https://mathai-iclr.github.io/papers/papers/MATHAI_29_paper.pdf>

**סרטון של יניק:** https://www.youtube.com/watch?v=dND-7llwrpw

## Review 68: PATCHES ARE ALL YOU NEED?

אתם אולי מכירים את חולשתי למאמרים מכילים ביטוי ״all you need". מרגיש מחובתי לסקור קצרות כל מאמר כזה שעיני נתקלת בו. אציין שלא הייתי סוקר את המאמר הזה אחרת 🙂

הפעם במוקד נמצא לא אחר אלא פאץ'. לאור ההצלחה של הטרנספורמרים גם בדומיין הויזואלי (Visual Transformer ודומיו). אזכיר שהקלט לטרנספורמר ויזואלי הינו פאצ'ים של תמונות. כלומר מחלקים תמונה לפאצ'ים, מייצגים כל אחד באמצעות וקטור (בדרך כלל פשוט משטחים את מטריצת הפיקסלים בפאץ' ומכפילים במטריצה).

אז המאמר שואל את השאלה הבאה: האם ההצלחה של הטרנספורמרים הויזואליים נובעת מעוצמתו של ארכיטקטורת הטרנספורמר או שזה פשוט עניין של בנייה נכונה של קלט לרשת, כלומר הפאצ'ים. בשביל לבדוק את העניין הזה המחברים בנו רשת ממש ממש פשוטה (אפילו יותר פשוטה מ- MLP-Mixer המפורסם) כשהקלט שלה הוא הייצוגים של פאצ'ים. הרשת בנויה בצורה מאוד פשוטה מבלוקים residual שכל אחד מהם מכיל קונבולוציה depthwise ו- pointwise אחד אחרי השני. בסוף יש שכבת pooling ו-FC. וזהה זה.

עם ארכיטקטורה מאוד פשוטה זו הם הגיעו לביצועים ברי השוואה עם קונפיגורציה מסוימת של ViT (הם גם השוו ביצועים עם איזה סוג של טרנספורמר, הנקרא DeIT). הם גם עשו השוואה על עם מודלים פשוטים כמו ResNet152 ו- ResMLP.

אבל

הבדיקה נעשתה על שתי משימות בלבד (סיווג על ImageNet ו- CIFAR10). הם לא ביצועי שום אימון self-supervised והוכיחו שהייצוגים שהארכיטקטורה שלהם, שקיבלה שם ConvMixer, יודעת להפיק, חזקים יותר. אלא רק אימון על דאטהסטים מתויגים (תקנו אותי אם פספסתי משהו).

בקיצור מאמר די מאכזב, השם מאוד באזזי אך לצערי לא הולם את התוכן…

אשמח לדעת מה דעתכם….

למאמר: [https://openreview.net/forum?id=TVHS5Y4dNvM](https://openreview.net/forum?id=TVHS5Y4dNvM&fbclid=IwAR12W5NN2amMChyvlYewhtMj72EJNdSc6MW4HJtCk8SZ-kzwhColYuFQ4ZA)

קוד: [https://github.com/tmp-iclr/convmixer](https://github.com/tmp-iclr/convmixer?fbclid=IwAR2pkW7KvIGzzecHEkbp1w-hgiKMQ1iafqWG_yiAIKyQOtIAkfCIqc5m96g).

## Review 69, Short: SIMVLM: SIMPLE VISUAL LANGUAGE MODEL PRE-TRAINING WITH WEAK SUPERVISION

בהתחלה רציתי לכתוב סקירה קצרה בסגנון #shortdeepnightlearners אבל למעשה ניתן לתקצר את המאמר הזה בכמה משפטים בודדים.

אתם בטח זוכרים את CLIP ו- DALL-E שהצליחו להפיק ייצוגים חזקים (ומתואמים!!) של דאטה ויזואלי וטקסטואלי באמצעות אימון משותף באמצעות שימוש בגישת הלוס הניגודי. כל זה כמובן נעשה על תמונות עם כותרות מהאינטרנט. מחברי SimVLM בחרו בגישה פשוטה יותר (לדעתי) - הם מאמנים רשת שבהינתן תמונה והתחלת הכותרת שלה, חוזה את החלק השני של הכותרת.

איך עושים זאת? מעבירים פאצ'ים של תמונה דרך האנקודר של הטרנספורמר יחד עם החלק הראשון של הכותרת. צריך לציין שאת הפאצ'ים מעבירים קודם דרך רשת באקבון קונבולוציונית (שלום inductive bias). לאחר מכן חוזים את החלק השני של הכותרת עם הדקודר של הטרנספורמר.

הייצוגים המופקים בשיטה אימון הצליחו להגיע לביצועים טובים במגוון משימות כגון סגמנטציה ומענה על שאלות לתמונה נתונה (open visual question answering) ב- zero-shot. גם לתת כותרת לתמונה הרשת יודעת לתת בצורה די טובה ואך זה לא מפתיע…

**מאמר**: https://arxiv.org/abs/2108.10904

**פרויקט**: <https://ai.googleblog.com/2021/10/simvlm-simple-visual-language-model-pre.html>

רב טוב חברים,

## Review 70, Short: Typical Decoding for Natural Language Generation

מאמר מסקרן המציע שיטת דגימה חדשה ממודלי שפה מאומנים. כמו שאתם בטח יודעים מקובל לדגום מילים (טוקנים) עם הסתברויות הגבוהות ביותר (בהינתן המילים הקודמות). ככה עובדים למשל nucleus ו- top-k. המחברים טוענים שפרדיגמת דגימה זו מובילה לטקסטים ״משמעמים מדי״ ויותר מדי דומים לדאטהסטים שהמודל אומן עליהם.

המאמר מציע פרדיגמת דגימה אחרת מבוססת על מילה טיפוסית (מבחינת האנטרופיה של המילה בהינתן הטקסט לפני) ולא על מילים הסבירות ביותר.בגדול מחשבים את האנטרופיה של מילה על סמך ההתפלגות שלה (softmax). לאחר מכן מחשבים את המידע של כל מילה (log2 של ההסתברות שלה). בסוף דוגמים מילים כאשר ההסתברות של כל מילה פרופורציונלי למרחק ההפוך של המידע שלה מהאנטרופיה של התפלגות המילה). כלומר ככל שהמידע של מילה קרובה יותר לאנטרופיה היא תדגם בהסתברות גבוהה יותר.

כך אנו דוגמים מילה טיפוסית (אנטרופיה היא למעשה המידע הממוצע של המילה) ולא המילה הסבירה ביותר.

יש במאמר דיון מעניין על אספקטים לינגוויסטיים של הגישה המוצעת.

**לינק:** <https://arxiv.org/pdf/2202.00666.pdf>

סרטון של יאניק: https://www.youtube.com/watch?v=\_EDr3ryrT\_Y&t=2484s

## Review 71: Deep Reinforcement Learning for Cyber System Defense under Dynamic Adversarial Uncertainties

**פינת הסוקר:**

**המלצת קריאה מעדן ומייק**: מומלץ לאנשים העוסקים בתחום ה-Cybersecurity או לאנשים שאוהבים Reinforcement learning

**בהירות כתיבה**: טובה

**ידע מוקדם**:

* הבנה בסיסית בלמידה עמוקה מחיזוקים (DRL)
* ידע בלמידה עמוקה

**יישומים פרקטיים אפשריים**: מערכת לזיהוי התקפות סייבר

**פרטי מאמר:**

**לינק למאמר:** [זמין להורדה](https://arxiv.org/abs/2302.01595).

**לינק לקוד**: לא אותר

**פורסם בתאריך**: 3.2.2023, בארקיב.

**הוצג בכנס**: -

**תחומי מאמר:**

* למידה עמוקה מחיזוקים
* אבטחת סייבר

**כלים מתמטיים, מושגים וסימונים:**

* SDP

**מבוא:**

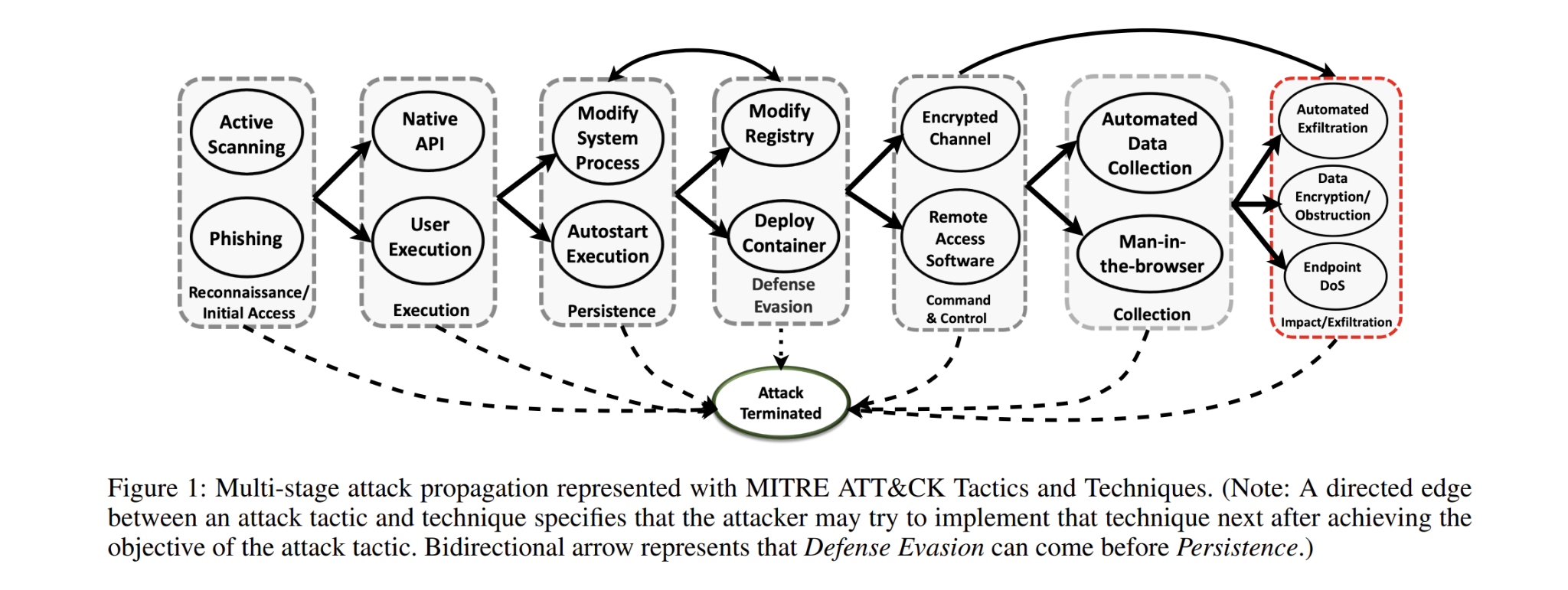
פיתוח אלגוריתמים האחראים על קבלת החלטות עבור מערכת הגנת סייבר איננה משימה פשוטה. הקושי העיקרי נובע מרמת דינמיות גבוהה שקיימת בסביבת תוכנה המכילות גם את כלי האבטחה (נקרא לו גם כלי הגנה בהמשך הסקירה) וגם את התוקף, המשפיעים יחד על הסביבה ומשנים את מאפייניה בכל נקודת זמן. בעיה נוספת הינה מגבלת המשאבים המוקצים לכלי המגן. משאבים אלו כוללים זמן מקסימלי לטיפול בפרצה, משאבי אנוש זמינים וכדומה. בכל פעולה של הכלי המגן הוא צריך להתחשב במגבלות אלו. למשל אם כל כוח העבודה שיכול לטפל בפרצה מסוימת תפוס, המודל לא יוכל להתריע על פרצה חדשה. לאחר שהכלי המגן מבצע פעולה, מספר המשאבים מתעדכן בהתאם, דבר שמשפיע על הסביבה.

בשנים האחרונות עם התקדמות הטכנולוגיה, חוקרים רבים ניסו לשלב למידה עמוקה מחיזוקים (DRL) בתוך כלים שונים תחום הסייבר. הפתרונות הללו ניסו למדל את הבעיה כ-Partially Observable Markov Decision Process ([POMDP](https://en.wikipedia.org/wiki/Partially_observable_Markov_decision_process)). הבעיה בפתרונות הללו הינן שהן התמקדו בסוג התקפות ספציפי, במרחב פעולות מצומצם מאוד של הכלי המגן יכול לבצע ובנוסף הניחו שקיים מודל לסביבה למרות שמודל הסביבה האמיתי מורכב מדי כדי למדל אותו בכלים מתמטיים פשוטים יחסית.

כדי לתת מענה לבעיות אלו, כותבי המאמר מציעים שיטה המתמקדת בפיתוח כלי מגן שיפעל כסוכן DRL שמטרתו להתמודד נגד תוקף מתוחכם ובעל יכולות מגוונות, כיוון שבעולם האמיתי אלו סוגי התוקפים אותם הוא (הסוכן) יפגוש. השיטה גם שכוללת פיתוח מסגרת (Framework) בה יהיה ניתן לבדוק ולהעריך את ביצועי הכלי המגן בצורה אמינה. מטרת הסוכן כפי שהגדירו אותה במאמר הינה ללמוד את אסטרטגיית ההגנה הטובה ביותר אשר תשפיע כמה שפחות על התהליכים הקיימים בסביבה שבה הוא מותקן (למשל להימנע ככל האפשר מפעולות כגון ניתוק מחשב מהרשת) ותדע להתחשב בכל שלבי ההתקפה האפשריים של התוקף.

לכן, החוקרים השתמשו ב-[MITRE ATT&CK Framework](https://attack.mitre.org/) כדי למדל את כלל הפעולות אותן יכול לבצע התוקף ובאותה נשימה לבדוק את ביצועי הכלי המגן. החוקרים מדגישים שהתרומות העיקריות במאמר זה הינן:

1. ההשוואה של ביצועי אלגוריתמי DRL שונים תחת תנאי סביבה אמיתיים.
2. פיתוחה של סביבת סימולציה חדשה המדמה את הסביבה האמיתית שאותה יראה המגן בחוץ באופן מדויק. הסביבה כוללת תוקף מתוחכם, המאופיינת בחוסר ודאות גבוה, תהליכים אמיתיים ואילוצים אמיתיים.



**התוקף:**

מטרת התוקף אותו מימשו במאמר הינה לנוע מתחילת גרף (תזרים) ההתקפה (אותו ניתן לראות באיור 1) לסוף המסלול. כדי לעשות זאת מומש מודל מבוסס אסטרטגיה. לפי האסטרטגיה אותה מיישם התוקף מגדירה הצלחתו בשלב נתון (במאמר זה נקרא טקטיקה) במסלול אם הוא הצליח להריץ את אחת ההתקפות (נקרא טכניקה במאמר) באותו שלב. למשל, אם התוקף הצליח להריץ את ההתקפה (טכניקה) הנקראת Active Scanning, הוא הצליח להגיע לשלב הנקרא Reconnaissance. כדי לבצע התקפה התוקף מריץ תהליך התקפה (Procedure) אשר בפועל מממש את ההתקפה. כלומר התקפה מוגדרת בתור סדרת מהלכים שעל התוקף לבצע ותהליך יכול להיות קוד בשפת תכנות כלשהי אשר מאפשרת לתוקף להגשים את מטרתו. כדי לעשות זאת התוקף מנצל חולשות הקיימות במערכת ולא טופלו כהוגן בכל נקודת זמן, מיקום התוקף הינו בשלב האחרון בו התוקף הצליח לבצע בהצלחה התקפה כלשהיא. התוקף יפסיד במשחק אם פעולת התקיפה שלו כשלה בשל פעולה של המגן (כלומר הוא נחשף). ההנחה היא שהתוקף רשאי לשנות את אסטרטגיית התקיפה שלו בכל נקודת זמן בהתאם למצב המערכת ותוצאת ההתקפה הקודמת שלו. לכן פתרונות הגנה סטטיים לא יספיקו שכן התוקף יוכל למצוא את המסלול (הרצת התקפה) שאותו כלי הגנה לא יצליח למצוא (אם קיים אחד).

**:MITRE ATT&CK**

ה-ATT&CK הינו שם למודל שארגון MITRE יצר לצורך הפשטה של סוגים שונים של התקפות הסייבר. ההפשטה נועדה כדי לעזור לקהילת הסייבר לסווג התקפות שהם רואים בשטח בהתאם למסגרת המוגדרת מראש ידי ATT&CK. המודל מציג גם פתרונות עבור סוגי התקפה שונים ולכן יש ערך בניסיון מיפוי התקפות לפי מודל זה. בנוסף אנשים שירצו לנסות לסמלץ התקפות בעצמם יוכלו לעשות זאת בקלות רבה יותר שכן המודל מכיר תיאור של מגוון סוגי ההתקפות ומה הן מנסות להשיג. תהליך ההפשטה של המודל מכיל שלושה רמות שונות של אבסטרקציה:

* טקטיקה - שם כללי להתקפה. השם מעיד על מטרת ההתקפה.
* טכניקה - תיאור טכני על ההתקפה. התיאור נשאר ברמת אבסטרקציה גבוה.
* תהליך - דרך המימוש של טכניקה מסוימת. כאן ממש נכתב קוד כדי לנסות לממש את הלוגיקה המוצגת בטכניקה.

שלושת מונחים אלו מהווים את עמודי התווך עליהם בנוי המודל של MITRE.

**המגן:**

מטרת המגן הינה למנוע מהתוקף להגיע לשלב האחרון המסלול התקיפה. כדי לבצע זאת, על המגן להבין איפה התוקף נמצא כעת ולחזות את הצעד הבא שלו. כדי לבצע זאת על המגן להתמודד עם אתגר מורכב הטומן בו חוסר ודאות הנובע מהשיקולים הבאים:

* התוקף יכול לנוע במסלולים שונים על גרף התקיפה. למשל, לאחר ביצוע ההתקפה User Execution הוא יוכל לנסות שתי פעולות אפשריות שכל אחת שייכת לשלבים השונים של גרף התקיפה למשל: Modify System Process או Modify Registry. זאת יחד עם העובדה שהמגן לא מכיר את גרף התקיפה אותו מיישם התוקף מקשים על המגן לחזות את הצעד הבא (טכניקה) של התוקף בצורה טובה.
* המגן לא יודע איזה תהליך (procedure) ביצע התוקף כדי לבצע טכניקה (התקפה) מסוימת בגרף התקיפה.
* המידע על מצב המערכת העומד לרשות המגן עלול להיות לא מלא. המגן נשען על כלי גילוי והתרעה שמטרתם לספק לו מידע על התרחשות חריגה במערכת. אותם כלים יתקשו לספק מידע על התקפה שבוצעה בשל ידע מוגבל על המערכת עליהן הן רצות או בשל העובדה שאין ברשותן מיפוי מפעילות חריגה להתקפה (טכניקה) ידועה.

המשימה של המגן נוסחה במאמר כבעיית (Sequential Decision Process ([SDP](https://en.wikipedia.org/wiki/Sequential_decision_making). בבעיה זו, הפעולה הבאה של התוקף תלויה רק במצב (S) האחרון והפעולה (A) האחרונה אותה ביצע. כעת נגדיר את המאפיינים העיקריים של SDP.

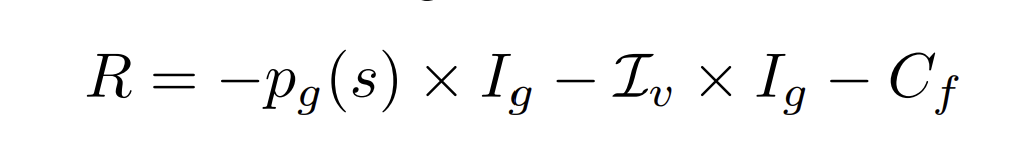
* מרחב המצבים (S): מכיל 17 מצבים שונים שכל אחד מהם הוא וקטור המייצג מיקום ייחודי של התוקף. המיקום מיוצג על ידי:

1. **הטכניקה** (סוג ההתקפה) בה השתמשו.
2. **המצב ההתחלתי** ממנו התחיל התוקף - Attack Initiated.
3. האם התוקף עבר למצב הסופי של התקפה זו- Attack Terminated.

* מרחב הפעולות (A): למגן שלוש פעולות אפשריות אותן יכול לבצע:

1. **Inactive**: המגן לא עושה כלום
2. Reactive: המגן מסיר את כל התהליכים במערכת הקשורים להתקפה האחרונה אותה ביצע התוקף.
3. Proactive: המגן חוסם סט ספציפי של קריאות API של המערכת כדי למנוע התקדמות של התוקף. דוגמא: חסימה של קריאת API לשינוי מפתחות Registry.

* פרס (Reward) - כותבי המאמר משתמשים בפונקציית הפרס הבאה:



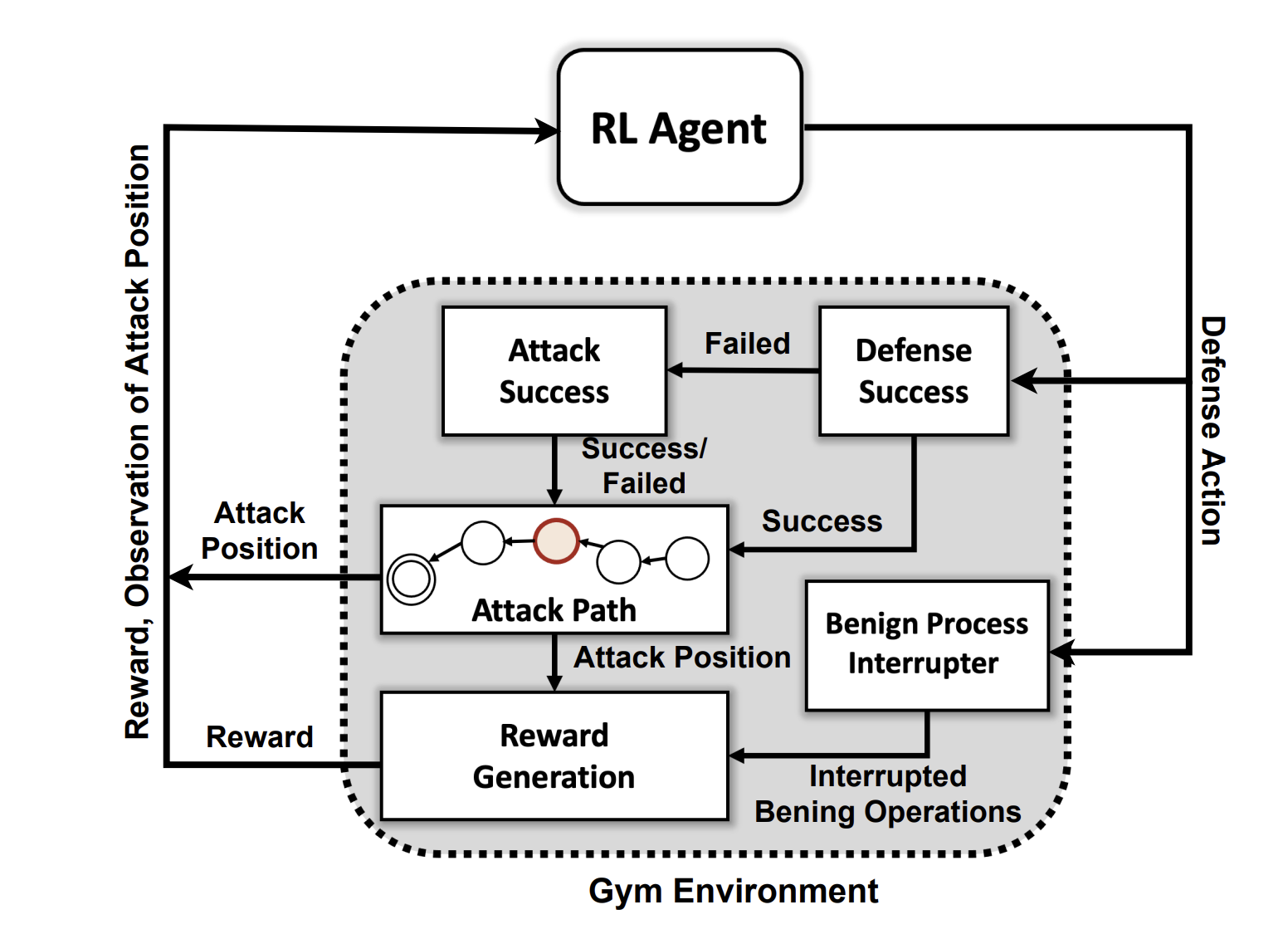
כאשר:

* שווה ל-1- אם המגן ניצח, אחרת 0.
* - ההסתברות של התוקף להגיע לשלב האחרון במסלול / גרף ההתקפה.
* - ההשפעה (=נזק) של התקפה מוצלחת על המערכת.
* - המחיר של ביצוע פעולת הגנה. המחיר תלוי במידת ההפרעה של פעולת המגן על פעילותה התקינה של המערכת.
* שילוב ו- מכמת את הסיכון (Risk) במצב s.
* שילוב ו- מכמת את התמריץ של המגן לנצח את התוקף שכן אם הוא ינצח ( שווה ל-1-) הפרס יגדל.

נשים לב שבפעולות 2 ו-3 המגן עלול להפריע/לגרום נזק לפעילות התקינה של הרשת / מערכת שעליה הסוכן מגן.

**הסביבה (Framework):**

בתוך המסגרת אותה מציעים חוקרי המאמר מתבצע התהליך הבא:



בכל נקודת זמן, המגן (סוכן DRL) מבצע צעד הגנה ומקבל כמשוב על הפעולה את מיקום התוקף על גבי מסלול ההתקפה ופרס המתבסס על מיקום התוקף ועל ההשפעה על הסביבה.

כדי לאתר את מיקום התוקף, קיימות מערכות גילוי והתרעה אשר בוחנות קריאות API אשר מתבצעות במערכת ומתריאות על קריאות חריגות. בנוסף ברקע רצים תהליכים נוספים שגרתיים שבמקרה שהמגן משבש את פעילותם הסדירה עלולים לגרום לתקלות שונות במערכת. כפי שצוין בחלק הקודם, לא תמיד קיים מיפוי בין התראה לבין טכניקה (סוג ההתקפה) וגם אם קיים הוא לא בהכרח נכון. לכן כדי למדל את תופעה זו כותבי המאמר מניחים שקיים מיפוי בין התראה לבין טכניקה אך דיוקו אינו מושלם ויכול לעמוד על 85%, 75% ו-65%. (אלו הדיוקים שנבדק על ידי הכותבים) כלומר כאשר התראה ממופת לטכניקה מסוימת קיים סיכוי של 65% למשל שמיפוי זה הוא נכון.

כעת נתאר את החלקים הנוספים בסביבה שבה המגן רץ:

* מגנון ה-Defense Success הינו חלק בסביבה האחראי לקבל את פעולת המגן ולהחליט האם היא עצרה את פעולת התוקף (ולכן הוא הפסיד).
* מנגנון ה-Attack Success אחראי לקבוע האם פעולת התוקף הצליחה וההתקפה בוצעה בהצלחה. אם אכן כך, התוקף מקבל אינדיקציה שהוא יכול לעבור לשלב הבא, אחרת הוא נשאר במקום או מפסיד.
* מנגנון ה-Benign Process Interrupter יודע לעקוב אחרי אילו תהליכים תמימים הופסקו כתוצאה מפעולת המגן. הוא שולח את הרשימה הזו למגנון ה-Reward Generation בשביל חישוב הפרס.

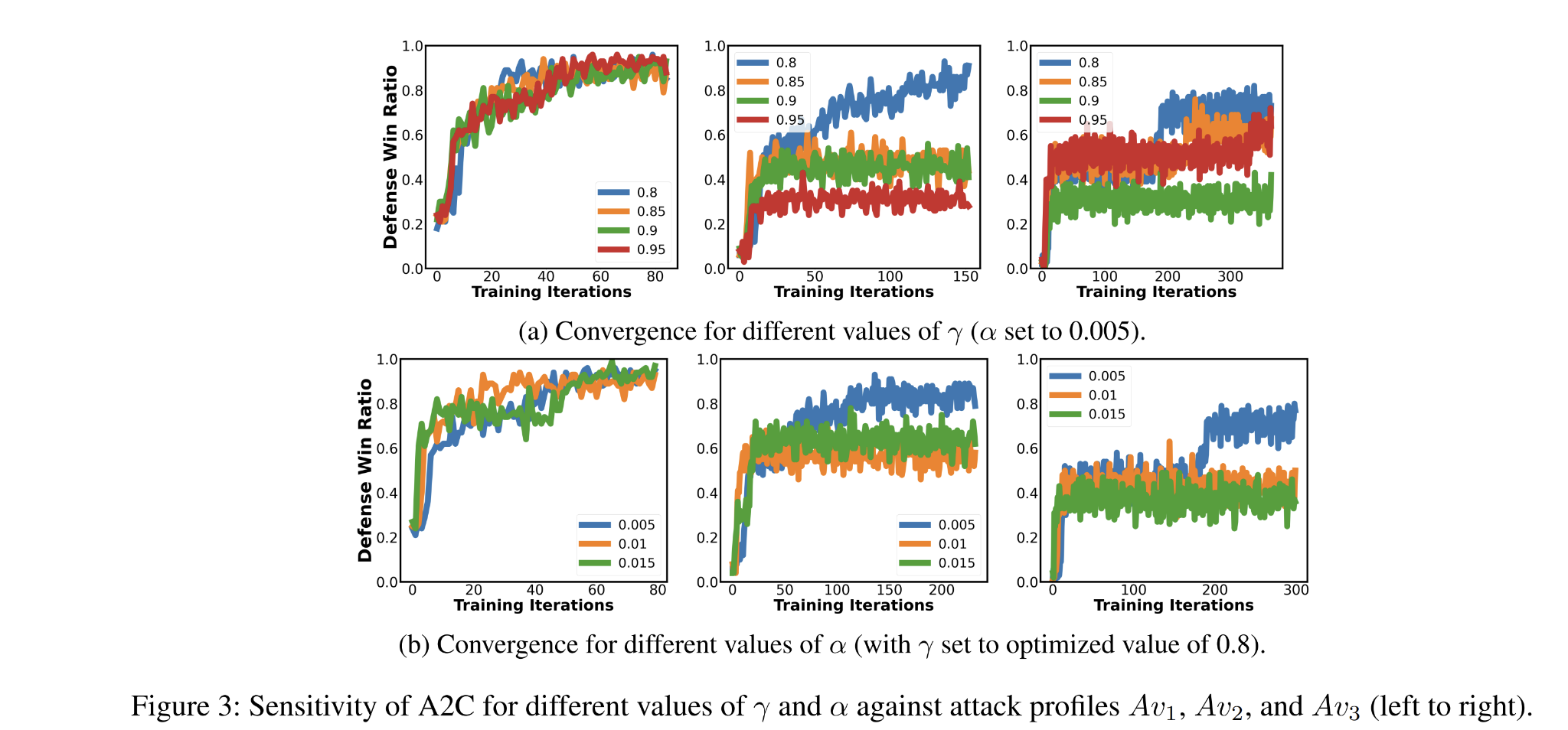
**ניסויים:**

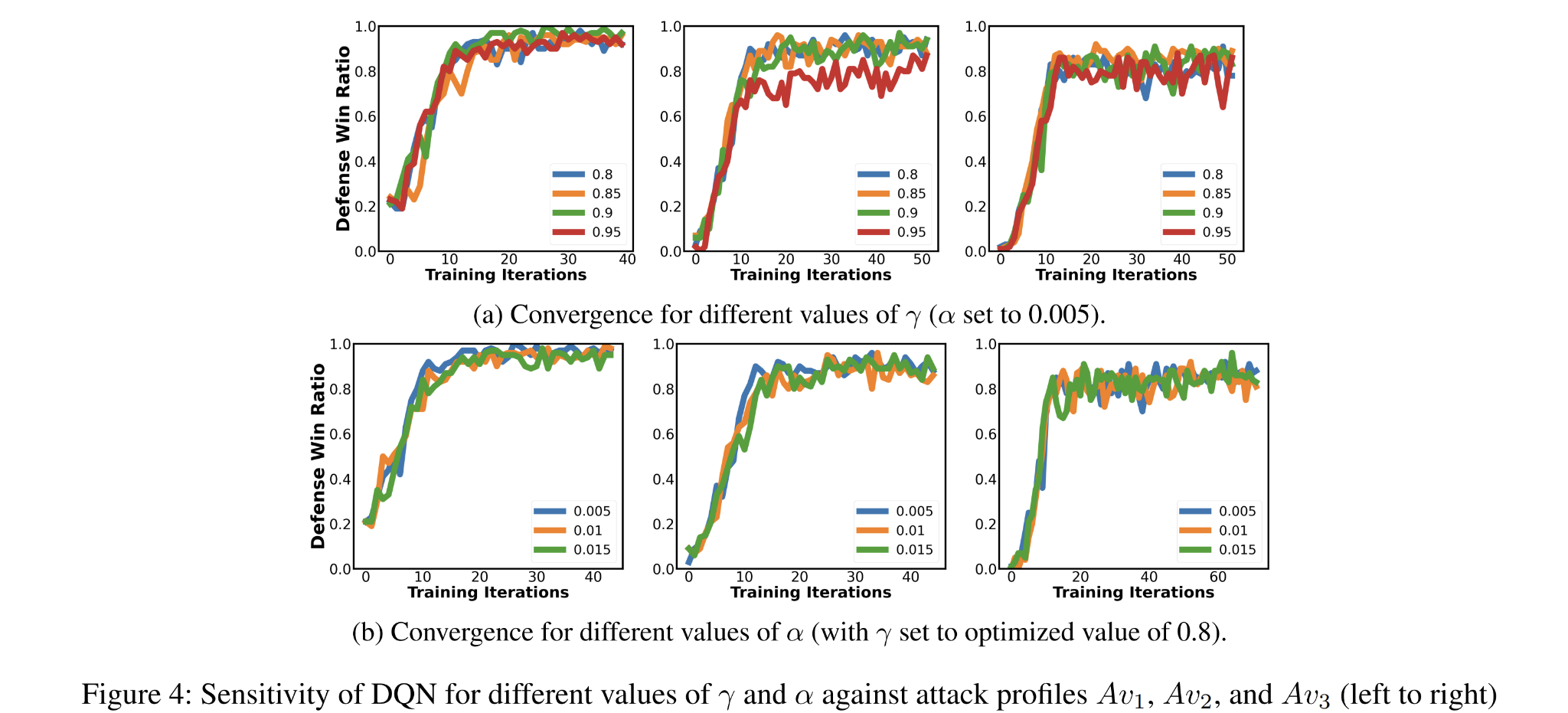
בסביבת הניסוי שלהם, החוקרים לא השתמשו במלוא הטקטיקות והטכניקות המופיעות ב-MITRE אלא ב-7 מתוך 11 הטקטיקות. כתוצאה מכך, מספר הטכניקות נמוך אף הוא מהמקור. מספר פעולות המגן שמומשו הינן 23 כאשר 21 מתוכן הן מסוג Proactive (לא מוזכר מה הן אותן פעולות).

בכדי לאמן את המגן, החוקרים יוצרים קודם את כלל מסלולי ההתקפה האפשריים בגרף ההתקפה מנקודת ההתחלה לנקודת הסיום. מתוכם 80% מסך המסלולים שומשו לאימון ו-20% לטסט. בזמן האימון כל איטרציה כוללת ניסיון של התוקף לנוע על מסלול אחד מתוך גרף התקיפה מההתחלה עד הסוף. בכך נותנים למגן לראות המון סימולציות שונות ותנאי סביבה שונים.

החוקרים מציגים את הטענה שמידול התוקף מתבצע בשני מישורים: כישרון ועקשנות. כאשר כישורי התוקף גבוהים יותר כך גדל הסיכוי שההתקפה שהוא בוחר לבצע בשלב הבא תצליח. תוקף עקשן הוא תוקף שלא יוותר וימשיך לנסות לתקוף גם אם לא צלח בניסיון הראשון. פירוש אחר הוא שתוקף עיקש הוא אחד כזה שלא ניתן לזהות בקלות גם אם התקפה מסוימת שלו כשלה. במקרה זה עקשנות הינה מספר הפעמים שהתוקף יבצע התקפה שנכשלה לפני שיפסיד (כלומר יתגלה על ידי המגן). החוקרים ניסו רמות שונות של כישרון ועקשנות כדי לאמן את המגן תחת דרגות שונות של קושי. בתור המגן נוסו אלגוריתמי DRL שונים, ביניהם: [DQN](https://paperswithcode.com/method/dqn) ו-[A2C](https://stable-baselines3.readthedocs.io/en/master/modules/a2c.html#notes).

**תוצאות:**

בתרשימים הבאים ניתן לראות את ביצועי כל אחד מהמגנים נגד שלושה סוגים של פרופילים של תוקף כאשר ההבדל בין כל פרופיל הינה רמת הכישרון והעקשנות של התוקף. המדד בו השתמשו להשוואה הינו אחוז הפעמים שהמגן ניצח את התוקף מסה"כ המשחקים ששוחקו.



מהתרשימים ניתן לראות ש-DQN הציג את הביצועים הטובים ביותר נגד סוגי התוקף השונים. הוא לא רק הצליח לנצח ביותר משחקים מאשר A2C אלא הוא הצליח לעצור את התוקף בשלב מוקדם יותר כפי שניתן לראות לפי מספר האיטרציות בציר ה-X, שם מספר האיטרציות נמוך יותר בתרשימים של DQN.

קיים פוטנציאל גדול לסוג כזה של מגן וסביבה בעולם האמיתי. מגן אשר ראה מספר רב ומגוון של התקפות ושל פרופילים שונים של התקפות. ביצועי ה-DRL מראים כיוון מחקר עם ישומים אמיתיים ועזרה אמיתית לארגונים להתמודד עם האיומים השונים המחכים להם.

**סיכום:**

מאמר זה מציג את ההיתכנות של פריסת סוכנים אוטומטיים לצורך הגנה מפני התקפות סייבר. כותבי המאמר הראו אשר ניתן לאמן סוכנים כאלו באמצעות אלגוריתם DRL אשר יוכל להגן בהצלחה מפני התקפות סייבר שונות. בנוסף הסוכן מסוגל להתמודד עם סוגים שונים של תוקפים מתוחכמים.

**שיתוף פעולה:** הסקירה נכתבה בשיתוף פעולה עם עדן יבין

## Review 72, Short: Unifying Large Language Models and Knowledge Graphs: A Roadmap

מה יקרה אם נשלב מודלי שפה גדולים (LLMs), כגון ChatGPT ו-GPT4 עם גרפי ידע (Knowledge Graphs)?

הרי KGs ו- LLMs משלימים אחד לשני באופן מאוד טבעי. KGs יכולים לעזור לטפל בבעית hallucination של LLMs וגם לחזק את ״התשתית העובדתית״ שלהם. LLMs יכולים לשפר את היכולת של KGs ל-reasoning.

המאמר מסכם את המחקרים האחרונים בנושא המעניין הזה:

1. איך לשדרג LLMs עם KGs?
2. איך לשדרג KGs עם LLMs?
3. דרכים לבנות מודלים המשלבים LLM ו-KG?

המחברים גם דנים בכיווני מחקר עתידיים ומשרטטים ״מפת דרכים״ עבורם.

**Paper:** <https://arxiv.org/abs/2306.08302>

**Github:** <https://github.com/RManLuo/Awesome-LLM-KG>

## Review 73, Short: Diffusion Models for Zero-Shot Open-Vocabulary Segmentation

המאמר מציע שיטת סגמנטציה zero-shot בעלת מילון פתוח, כלומר אין לנו מכלול סגור קטגוריות אלא כל תמונה מקבלת סגמנטציה עם הקטגוריות משלה. כמובן שהשיטה לא דורשת שום אימון מקדים (zero-shot). זה מתבצע באמצעות מודל Text2Image דיפוזיה (DDPM) גנרטיבי

השיטה המוצעת מכילה שני שלבים:

1. בשלב הראשון (אימון מקדים) המחברים מגנרטים מספר רב של תמונות לקטגוריה(פרומפט) מסוימת (כמו תמונה טובה של חתול או כלב) בעזרת מודל דיפוזיה מאומן. לאחר מכן הם בונים ״מסד טיפוסים של פיצרים״ של כל קטגוריה שכולל הסגמנטציה של האובייקטים (וגם רקע) באמצעות מודל סגמנטציה מאומן.
2. בשלב השני מפיקים את הפיצ'רים של תמונה באמצעות CLIP ומשווים אותה עם אב הטיפוסים של הפיצ'רים שישי במסד הנתונים. הקטגוריה והסגמנטציה הדומה ביותר ממסד הפיצ'רים נבחרת.

HuggingFace: https://huggingface.co/papers/2306.09316

Arxiv: https://arxiv.org/pdf/2306.09316.pdf

## Review 74, Short: Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture

המאמר של יאן לקון (Yann LeCun) מציג גישה חדשה ומעניינת I-JEPA לבניית ייצוג (representation or embedding) של דאטה ויזואלי (תמונות) לא מתויג כלומר )SSL) self-supervised. שיטות SSL הקודמות אימנו מודל האוכף דמיון גבוה בין ייצוגי אותה פיסת דאטה אחרי אוגמנטציות שונות, השומרות על התוכן.

במקום זאת המחברים מאמנים מודל המנסה לחזות ייצוג של חלק אחד של הדאטה מהחלק האחר. למשל בתמונות המודל חוזה ייצוג של פאץ' מייצוג של פאץ אחר. השיטה הצליחה להפיק ייצוגים בעלי ביצועים חזקים מאוד על מגוון משימות כסיווג, ספירת אובייקטים ובניית מפות עומק.

אחד היתרונות המשמעותיים של המאמר הוא שהוא מייתר במידה רבה את הצורך לבנות אוגמנטציות השומרות על התוכן הסמנטי של התמונות בצורה ידנית.

HuggingFace: https://huggingface.co/papers/2306.09316

Arxiv: <https://arxiv.org/pdf/2306.09316.pdf>

## Review 75, Short: Recurrent Memory Decision Transformer

המאמר מציע לשלב את המנגנון של הזכרון recurrent בטרנספורמרים לפתרון בעיות של למידה באמצעות חיזוקים (reinforcement learning - RL). המאמר מנסה לתת מענה לאחת הסוגיות המרכזיות העולות כאשר משתמשים בטרנספורמרים לעיבוד של דאטה בבעיות RL: אי היכולות שלהם להחזיק רצפים ארוכים בזכרון בעקבות הסיבוכיות הריבועית של מנגנון תשומת הלב שלהם.

המאמר מציע שיטה לעדכון של הזכרון כתלות באופיינים השונים כמו המצב המערכת, אובזרבציה, פעולה והתגמול. כך מאתפשרת שמירה בזכרון המודל רצפי דאטה יותר ארוכים שמשפר את התורם חיובית ליכולות הלמידה של מודל. השיטה המוצעת הציגה ביצועים במשחקים כמו Atari ו -MoJoCo.

OpenReview: https://openreview.net/forum?id=Uynr3iPhksa

Arxiv: <https://arxiv.org/abs/2306.09459>

## Review 76, Short: Gradient is All You Need?

המאמר מנתח שיטות אופטימיזציה מסוג של consensus-based(CBO) ומשווה אותן עם השיטות ממשפחת GD שאנו מכירים היטב ומשתמשים בהם רבות. CBO היא משפחת שיטות שמריצות מספר ״סוכנים״ (לפעמים בלתי תלוים ולפעמים לא) הבוחנים את מרחב האופטימיזציה של הבעיה. המילה consensus מופיע בשם השיטה כי עם מספר רב של סוכנים חושבים שאיזור במרחב האופטימיזציה הוא ״טוב״ (הפונקציה מקבלת בו ערכים נמוכים אם מדובר בבעיית המזעור) אז כנראה כדי למקד את המיפוש באיזור הזה.

המאמר מראה שלקטוגוריות רבות של פונקציות לא קמורות ההתנהגות של CBO היא די דומה לשיטות GD למרות ששיטות CBO לא משתמשות בגרדיאנטים כלל? ולמה זה חשוב? כי מהשקילות הזו ניתן להסיק שבתנאים מסוימים GD מתכנסות למינימום עבור קטגוריות רחבות של פונקציות לא קמורות כי CBO מתכנס אליו.

Arxiv: <https://arxiv.org/abs/2306.09778>

## Review 77, Short: Diffusion with Forward Models: Solving Stochastic Inverse Problems Without Direct Supervision

המאמר מציע מודל דיפוזיה גנרטיבי (DDPM)למקרה שבו הדאטה לאימון לא זמין בצורה ישירה אלא רק אחרי טרנספורמציה שאחריה חלק מהמידע על הדאטה הולך לאיבוד. בדרך כלל אנחנו מאמנים מודל דיפוזיה על דאטהסט המכיל , נגיד, תמונות בעלי תיאור מילולי, והמודל לומד ליצור תמונה מתיאור כאשר יש לנה תמונה אמיתית עם התיאור הזה - אז די ברור איך לאמן את המודל.

המאמר מטפל במקרה מורכב יותר שניתן לראותו למשל במשימת inverse graphics. כאן המודל מתבקש ליצור דגימות של סצנת 3D כאשר יש לנו ביד רק תמונה אחת או כמה תמונות אך אין לנו את מודל ה-3D של הסצנה עצמה. המחברים מציעים לאמן מודל דיפוזיה המשלב יצירה של מודל הסצנה יחד עם הדגימה מהסצנה הזו. המודל מאומן ליצור דגימות שמצד אחד מתאימות למודל הסצנה הנבנה ומצד שני אוכף את מודל הסצנה להיות תואם לתמונה הנתונה.

Arxiv: <https://arxiv.org/abs/2306.09778>

## Review 78, Short: Fast Segment Anything

המודל Segment Anything - SAM שפורסם לאחרונה הפך להיות מודל בסיס למשימות הראייה הממוחשבת כמו סגמנטציה, גנרוט כותרת של תמונה וגם לעריכה של תמונה. אולם שימוש במודל זה מצריך כוח חישוב משמעותי הנובע מהחישובים של מודל הטרנספורמר עבור תמונות ברזולוציה גבוה.

המאמר מנסח את בעיית הסגמנטציה בתור שילוב של שתי בעיות: גנרוט סגמנטים ו-prompting (יצירת כותרת לסגמנט). רפורמולציה זו מאפשרת לנצל מודל ל-instance segmentation מבוסס רשתות קונבולוציה שמאפשר לחסוך חלק גדול מהחישובים.

Arxiv: <https://arxiv.org/abs/2306.09778>

## Review 79, Short: SqueezeLLM: Dense-and-Sparse Quantization

מה הדרך ההגיונית לשמור ולהריץ מודלי שפה במחשבים אישיים לא חזקים במיוחד? כמובן לשמור את המשקלים שלה לא בדיוק המלא (32 ביט) אלא בדיוק חלקי (נגיד 4 ביט). אך איך לעשות זאת כדי לא לספוג ירידה קשה בביצועים?

הקוונטיזציה היוניפורמית לא תספק לנו כאן ביצועים טובים כי המשקלים לא מפולגים יוניפורמית. המאמר מציע לקלסטר משקלים עם k-means כאשר המרחק מבוסס על החשיבות של המשקלים למודל (כמה הם משפיעים על הלוס). זה מאשפר לשמור משקולות שחשובים יותר בדיוק גבוה כאשר הפחות חשובים נשמרים בדיוק נמוך.

המחברים גם שמו לב כי ה-outlierים משפיעים לרעה על איכות הקלסטרים (מורחים אותם) והחליטו לשמור אותם בקלסטרים נפרדים מהשאר

<https://arxiv.org/abs/2306.07629>

## Review 80, Short: SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis

זהו השדרוג של מודל SD2.0 המגנרט תמונות באיכות מטריפה מטקסט. המאמר הציע כמה שפצורים לארכיטקטורה של SD המקורי, שכלל את האימון והשתמש ב-SDEdit לשיפור איכות התמונות

המודל מבוסס על אותם העקרונות כמו SD המקורי אך מוספים refiner באמצע המבוסס על SDEdit (דגימה ממודל דיפוזיה דרך פתרון נומרי של מישדיפ סטוכסטי :).

כלומר תהליך הדגימה נראה כך:

1. שיבוץ (embedding) הטקסט במרחב הלטנטי עם ה-encoder ודגימה מרעש גאוסי
2. יצירה של שיבוץ התמונה עם מודלי דיפוזיה לטנטי LDM (כמו ב-SD)
3. SDEdit לשיפור וקטור לטנטי מסעיף 2
4. יצירת תמונה מהשיבוץ של באמצעות ה-decoder

כאמור יש שינוים בארכיטקטורה (עדיין מבוססת על UNet ומנגנון attention). הם גם "העשירו״ את תהליך האימון:

1. הכניסו תמונות בגדלים שונים אך העבירו לרשת את הגודל שלהם
2. עשו crop לתמונות והזינו לרשת את גודל הקיצוץ.

<https://arxiv.org/abs/2307.01952>

## Review 81, Short: Segment Anything Meets Point Tracking

המאמר מציע שיטת SAM-PT לביצוע סגמנטציה ומעקב של אובייקטים בוידאו (VOS). המודל ממנף את מודל הסגמנטציה SAM עוצמתי שעלה לאוויר לא מזמן.

השיטה מכילה 4 שלבים עיקריים:

1. מגדירים את האובייקט שאנו עוקבים אחריו בפריים הראשון באמצעות כמה נקודות (או שמגדירים אובייקטים ורקע לא קשורים לאובייקט המעקב
2. משתמש בשיטות מאומנות של מעקב אחרי הנקודות בוידאו כמו PIPS כדי לעקוב אחרי הנקודות המסומנות.
3. משתמשים ב-SAM בשביל בשביל לבנות (סגמנטציה) מנקודות אלו את האובייקטים המתאימים לכל פריים של וידאו. בשלב הראשון מכניסים ל-SAM רק את הנקודות החיוביות (השייכות לאובייקט) ובשלב השני מכניסים גם את הנקודות השליליות (כמו רקע) יחד עם התוצאת הסגמנטציה הראשונה כדי לקבל את תוצר הסגמנטציה הסופי.

רובוט יקר, תעשה לי קפה ותשים על השולחן! 😶‍🌫️

רוצים שיהיה לכם רובוט משלכם שתוכלו לתת לו משימות בבית? 😶‍🌫️

## Review 82, Short: SayPlan: Grounding Large Language Models using 3D Scene Graphs for Scalable Task Planning

המאמר ממנף את כוח של #llms כדי לעזור לרובוט לתכננן משימות ב-indoor.

איך עושים זאת? בונים את גרף הייצוג התלת מימד (3D) של המקום (scene) הנקרא 3DSG . בגדול 3DSG מכיל את את כל האובייקטים שיש בסצנה, את המיקומים היחסים שלהם. אחר כך משתמשים ב- #llm (מזינים בה פרומפט והוראה עצמה) בשביל למצוא את הקודקודים הרלוונטים למשימה ולתכנן תכנית ביצוע המשימה. אם לא מצליחים לבצע את המשימה לוקחים את המשוב ומעבירים אותו שוב ל-llm עם הפרומפט ואת ההוראה כדי לבנות תכנית פעולה חדשה. מסמלצים את תכנית, יוצרים את המשוב עד המצליחים לסיים את המשימה. די מגניב האמת.

## Review 83, Short: HyperDreamBooth: HyperNetworks for Fast Personalization of Text-to-Image Models

זה למעשה שילוב של DreamBooth ו- (Low Rank Adaptation (LoRA

מה DreamBooth עשה? ביהנתן תמונה או מספר תמונות (לא הרבה, נגיד 3-4) הוא מכייל מודל SD כך שהוא יודע ליצור תמונות של הכלב הספציפי שלך, נגיד, בכל מיני סצנות (המתוארות עם תיאור טקסטואלי). הכיול מתבצע על ידי מתן שם (מורכב מטוקנים נדירים) לאובייקט ואז מכיילים SD עם תמונות האובייקט תוך שמירה של יכולתו ״בהבנת העולם הויזואלי״.

מה זה LoRA? נניח אתם רוצים לכייל מודל שפה ענקי של 50B פרמטרים אבל אין ברשותכם כוח חישוב אינסופי. אז אתה עושה שני דברים:

1. לכל מטריצת המשקולות אתה לומד את ההפרש (delta) מהיעד (המשקלים אחרי הכיול) ולא משקל היעד הסופי.
2. אתה מניח שמטריצת ההפרש הזו כי low-rank וניתן לתאר אותה כמכפלה של 2 מטריצות לא מאוד גדולות שאתה לומד אותן. ככה חוסכים בכמות המשקלים הנלמדים.

אז מה זה HyperDreamBooth? זה פשוט LoRA ו- DreamBooth המשולבים בצורה חכמה

## Review 84, Short: Learning to Retrieve In-Context Examples for Large Language Models

הידעתם ש-llms יודעים להתאים את עצמם למשימות חדשות ללא שום כיול (=שינוי המשקלים) אחרי כמה דוגמאות? לפעמים זה עובד ללא דוגמאות. זה נקרא למידה in-context או ICL

מה קורה עם משלבים ICL עם אוגמנטצית אחזור: קרי נותנים ל-LLM להשתמש בדאטה חיצוני (RAG)?

Learning to Retrieve In-Context Examples for Large Language Models

המודל מאומן בכמה שלבים:

1. לשאילתה נתונה מאחזרים כמה דוגמאות מהדאטה החיצוני, זוגות של (שאלה, תשובה) עם אלגוריתם BM25 מריצים llm כדי לדרג את הרלוונטיות של לשאילתה. בגדול ככל שהזוג ״מוביל״ אותנה לתשובה מכון הוא מדורג גבוה יותר
2. מאמנים מודל תגמול (reward) עם למידה ניגודית (דוגמאות מדורגות גבוה מול אלו שמדורגים נמוך). מאמנים אנדוקר לשכן (embed) **יחד** את הזוג ה-ground truth של השאילתה (x, y) יחד עם הזוג המאוחר (x\_i, y\_i). מודל זה משמש רק בתור מודל מורה לשלב הבא כי משתמש בתשובה y שלא ידועה כמובן בזמן הטסטינג
3. אימון מאחזר llm עם RAG: מאמנים מודל שיכון עבור זוג (x\_i, y\_i) ממאגר חיצוני ועבור דוגמא x. בוחרים זוגות עם דמיון הכי גבוה בין ייצוגו לבין הייצוג של שאילתה x (מרחק cosine). מאמנים את הייצוגים אלו כדי עם שילוב של 2 פונקציות יעד:
   1. קרבה בין ההתפלגויות של מודל המורה מהשלב הקודם לבין המודל מודל המאחזר את הזוג הקרוב ביותר (עם הייצוגים שחושבו לפני)
   2. מקסום מרחק בין ייצוגים של הדוגמאות המדורגות גבוה לבין אלו שמדורגות נמוך (לוס ניגודי ממאמר InfoNCE)

מעניין שעושים כמה איטרציות של אחזור כאשר משתמשים בזוג שנמצא באיטרציה הקודמת לאיטרציה הבא שמשפר את איכות האחזור.

## Review 85, Short: Anticorrelated Noise Injection for Improved Generalization

עבודה מעניינת המציע שיטה מאוד פשוטה לשיפור ביצועים של (stochastic gradient descent (SGD

הוספת רעש אקראי לעדכוני משקלים (perturbed GD) המתבצעים במהלך SGD נחקרה בכמה עבודות בשנים האחרונות. אחת המסקנות של עבודות אלו היא שהזרקה של רעש (בדרך כלל גאוסי) בעל שונות נמוכה יחסית ל- SGD עשויה לשפר את יכולת ההכללה של הרשת המאומנת.

אבל למה זה עוזר בעצם? מקובל להסביר את ההשפעה החיובית של הזרקת רעש ל-SGD בכך שזה עוזר לפונקציית לוס להתכנס לנקודת מינימום ״רחבה״ כלומר כזו שברוב הנקודות בסביבה הקרובה שלה ערך של פונקצית לוס נותר נמוך. מינימום רחב של פונקציית לוס נחשב לטוב ליכולת הכללה של המודל לעומת מינימום חד כלומר כזה שאפילו בסביבתו הקרובה יש עלייה ניכרת בערכי פונקציית לוס. הסיבה לכך היא טמונה בהנהה (יש גם תוצאות תיאורטיות חלקיות המוכיחות זאת עבור מודלים פשוטים יחסית) שנקודת מינימום רחבה ״מבטאת את הסיגנל האמיתי״ מהדאטה ולא נוצרת כתוצאה של הרעש שנמצא בדאטה.

אוקיי, אז מה מציע המאמר? בדרך כלל מוספים רעש גאוסי בלתי תלוי לעדכוני משקלים של SGD והמאמר שואל שאלה לגיטימית: האם זה אופטימלי לביצועים. מתברר שלא כל כך….

המאמר מציע להוסיף רעש בעל קורלציה שווה ל -1. איך עושים זאת? דוגמים סדרה ארוכה של **וקטורים** גאוסית **x\_i** ויוצרים סדרת הפרשים בין איבר סדרה הסמוכים (**x\_2 -x\_1, x\_3 - x\_2**…, **x\_n+1 - x\_n...**) ומוסיפים אותה לעדכוני משקלים של SGD באיטרציה **n**. מתברר שזה עוזר בלא מקרים ליכולת ההכללה של הרשת. הם גם מראים שהערכים העצמיים של ההסיאן בנקודת מינימום עם השיטה שלהן (השיטה נקראת Anti-PGD) יותר נמוכים מאשר ל-PGD עם רעש חסר קורלציה.

המאמר גם דן (לא התעמקתי) בקשר בין הטכניקה המוצעת להוספה של רעש אקראי ללייבלים וגם לשיטות החלקה שונות (smoothing).

המאמר כתוב היטב ודי קל להבנה.

**מאמר:** <https://arxiv.org/abs/2202.02831>

## Review 86, Short: BuboGPT: Enabling Visual Grounding in Multi-Modal LLMs

רוצים מודל העונה על השאלות לגבי האובייקטים בתמונה וגם מראה לכם איפה האובייקט נמצא בתמונה ולתת הסבר לגביו? בנוסף לאודיו נתון אתו המודל יותר לסמן אותכם מי בתמונה שלכם הוא מקור האודיו. וכל זה מלווה בהסברים

היום ב-#shorthebrewpapereviews:

BuboGPT: Enabling Visual Grounding in Multi-Modal LLMs

המאמר משלב כמה מודלים חזקים שאומנו למשימות הקשורות ל-grounding ויזואלי כמו Recognize Anything Model שמפיק את הקטגוריות של האובייקטים. לאחר מכן מזינים הקטגוריות אלו למודל Ground-DINO שמזהה את מיקום האובייקטים. בסוף מכניסים את התוצאה ל-SAM ועושים סגמנטציה עדינה של האובייקטים

בשלב האחרון צריך למצוא את הזוג (קטגוריה, מסכה) המתאים לשאילתה שלכם. בשביל כך מעבירים את השאילתה דרך GPT4 ומחפשים את הקטגוריה הקרובה ביותר מהרשימה שהוצאה על יד SAM. אוקיי סיימנו עם grounding ויזואלי אבל הבטחתי לכם 3 מודים - אז איפה נעלם האודיו?

קודם כל ״מיישרים״ את הייצוגי אודיו ותמונות על ידי אימון שני אנקודרים (לכל אחד) על דאטהסט של מכיל זוגות של אודיו ותמונה מתאימה. גם כאן מכיילים מודלים קיימים: BLIP2 לתמונה ו-ImageBind לאודיו. בסוף מכיילים את כל המפלצת הזו על שילובים של אודיו-טקסט, תמונה-טקסט, ותמונה-טקסט-אודיו.

<https://huggingface.co/papers/2307.08581>

## Review 87, Short: TokenFlow: Consistent Diffusion Features for Consistent Video Editing

סוקרים המאמר **כחול-לבן** היום קצרות ב #shorthebrewpapereviews

**חוסר רציפות בין הפריימים**: מה הבעיה הגדולה ביותר בעריכה של וידאו עם באמצעות מודלי דיפוזיה גנרטיביים? מודלי דיפוזיה מסתדרים די יפה עם עריכה של תמונות לפי תאור טקסטואלי אבל עם הוידאו הסיפור הוא יותר מסובך כי נדרשת רציפות בין הפריימים.

הדרך הנאיבית לבצע עריכת וידאו בהתאם לתיאור טקסטואלי היא לערוך כל פריים (תמונה). אבל איך נשמור על קוהרנטיות בין הפריימים הערוכים? המחברים לוקחים פיצרים של הפריימים הסמוכים ולהשתמש בהם ולהחליק את הוידאו ערוך בעזרת אינטרפולציה של הפיצ'רים של הפריימים הקרובים לו.

אבל מה הם הפיצ'רים של הפריימים שכדאי לקחת? קודם כל המחברים לוקחים את את השאילתות, מפתחות וערכים (queries, keys, values) ממנגנוני ה-attention מכמה פריימים ערוכים **סמוכים** של הוידאו המקורי. לאחר מכן עבור פריים i מפעילים מנגונן -attention על השאילתה שלו ועל המפתחות והערכים של הפריימים.

ככה למעשה מחושב ״צוג הרציפות״ של וידאו (המורכב מייצוג של כל פריים ביחס לפריימים האחרים). לכל פריים מחפשים את הפריים הבא לפניו וזה שבא אחריו עם ה-attention הקרוב ביותר לפי מרחק הקוסינוס מבחינת ייצוג הרציפות. ואז עבור כל פריים שאנחנו עורכים אנו משפרים את רציפות בין הפריימים תוך שמירה על אותו ״ייצוג רציפות״ כמו בוידאו המקורי על ידי אינטרפולציה שלו על ידי שני ייצוגי הרציפות של הפריימים שמצאנו.

<https://arxiv.org/abs/2307.10373>

## Review 88: Secure Machine Learning in the Cloud Using One Way Scrambling by Deconvolution

**פינת הסוקר:**

**המלצת קריאה מעדן ומייק**: מומלץ לאנשים העוסקים בתחום ה-Cybersecurity או/ו לאנשים שמתעניינים בטכניקות לסיווג תמונות (בלי לראות אותן!!)

**בהירות כתיבה**: טובה

**ידע מוקדם**:

* הצפנה
* Deconvolution

**יישומים פרקטיים אפשריים**: מערכת להצפנת מידע

**פרטי מאמר:**

**לינק למאמר:** [זמין להורדה.](https://arxiv.org/abs/2111.03125)

**לינק לקוד**: לא אותר

**פורסם בתאריך**: 4.11.2021, בארקיב.

**הוצג בכנס**: - KDD

**תחומי מאמר:**

* פרטיות המידע
* מודלים בענן

**כלים מתמטיים, מושגים וסימונים:**

* אנקודר(להפקת ייצוג לטנטי של תמונה) ודקודר

**מבוא:**

הרצת מודלים גדולים של למידה עמוקה איננה משימה קלה. נדרש מאמץ רב ויכולות מחשוב כבירות (בד״כ GPU-ים או TPU-ים) סביב אימון המודל והפעלתו (Inference) שלו. חסם זה מנע משחקנים רבים את היכולת להשתמש במודלים ענקיים אלו . אך כיום זה משתנה עקב שימוש הולך וגדל בשירותי למידת מכונה בענן בו חברות פורסות מודלים גדולים ומאפשרים גישה לכלל הציבור דרך קריאות API שנעשות על גבי האינטרנט. גישה זו אמנם הנגישה מודלים רבים ואפשרה למספר רב של משתמשים להריץ אותם שלא היה ניתן קודם אך גם הביאה איתה מספר סוגיות רציניות. בעיה אחת מרכזית הינה פרטיות המידע: בשביל להעביר למודל מידע הנחוץ להרצתו יש צורך לשלוח אותו על גבי האינטרנט - - דבר שעלול לגרום לזליגת מידע רגיש. הבעיה השניה הינה היכולת של גורמים זרים להסיק מסקנות לגבי המידע של הארגון שלך בצורה עקיפה. ניתן לעשות זאת על ידי ניתוח הפלט של המודל בענן וזאת בשל העובדה שלכולם יש גישה לאותו מודל.

כדי להתגבר על הבעיות המוזכרות לארגונים היום קיימות שתי אופציות:

1. להצפין את המידע לפני שליחתו, פענוח בצד השני, הצפנת התוצאה ושליחתו חזרה לפענוח אצל הלקוח.
2. שימוש בהצפנה הומומורפית ([Homomorphic Encryption - HE](https://en.wikipedia.org/wiki/Homomorphic_encryption)). הצפנה זו מאפשרת לצד המקבל את המידע המוצפן לא לפענח אותו, אלא לנתח אותו ולהגיע למסקנות דומות באותה צורה שבה היה מגיע אילולא המידע לא היה מוצפן מלכתחילה.

אך לפתרונות אלו יש כמובן בעיות. הבעיה עם הפתרון הראשון הינו שלאחר פענוח המידע בצד של ספק הענן, לאותו ספק תהיה גישה למידע הלא מוצפן. כלומר, יש פה מקרה של הפרת פרטיות המידע. הבעיה עם הפתרון השני הינו שהוא איטי שכן עיבוד המידע המוצפן קשה יותר.

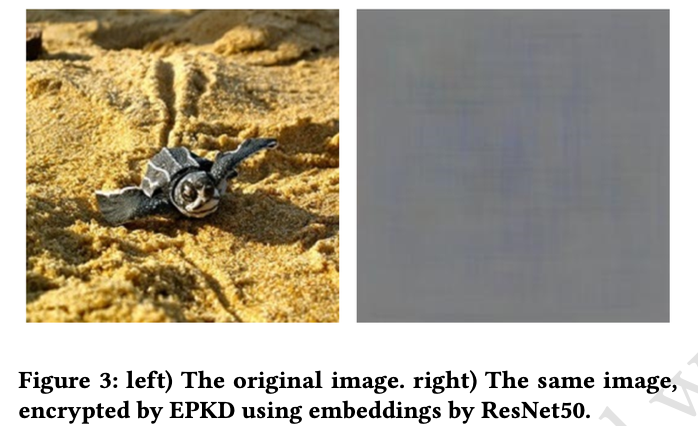
כדי להתגבר על הבעיות של פתרונות אלה בבת אחת, כותבי המאמר מציעים שיטה הנקראת Encoding via Private-Key Deconvolution (EPKD).

**השיטה:**

השיטה המוצעת דומה ל-HE בכך שהיא מאפשרת למודל בענן לקבל את המידע מוצפן ולעבד אותו כאילו לא היה מוצפן אך עם מהירות דומה לעיבוד המידע הרגיל. כלומר החיסרון של מהירות העיבוד שמציג HE לא משחקת פה תפקיד. השיטה משתמשת בשלושה רכיבים:

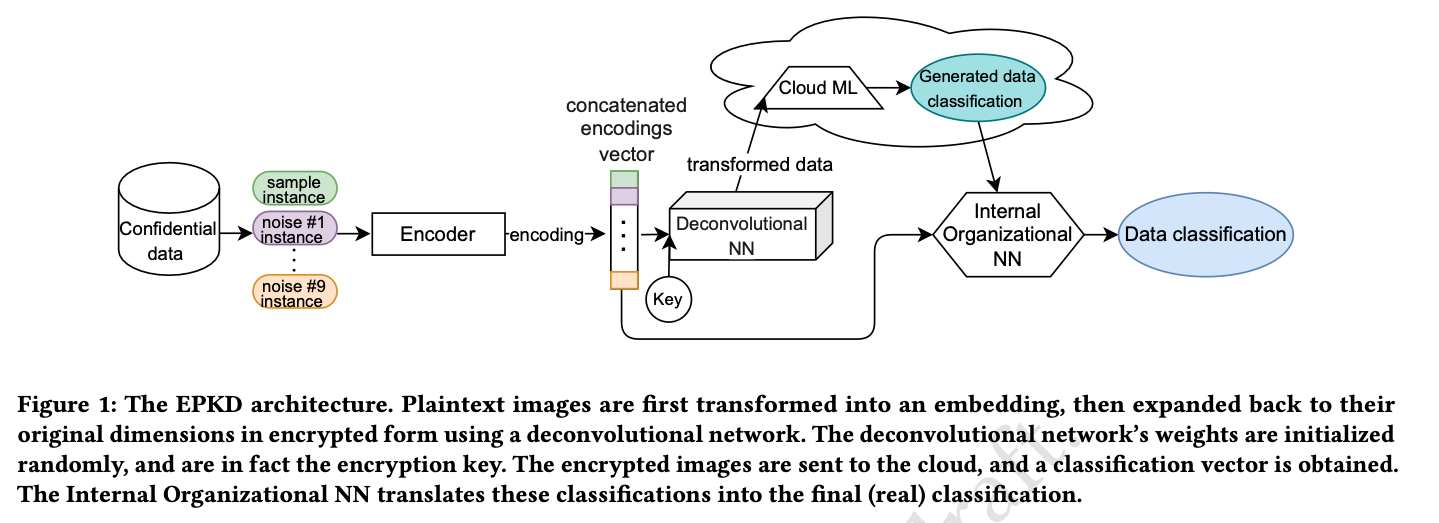
1. Encoder(מקודד): נועד כדי ליצור ייצוג דחוס של המידע. בעבודה השתמשו החוקרים בכמה מודלים מסוג Resnet כגון ResNet5, ResNet101, ResNet50v2. כולם היו מאומנים מראש ולא כוילו בנוסף. השכבה האחרונה של ה-softmax הוסרה כדי להשתמש בוקטור הייצוג(Embedding) שהרשת יוצרת.
2. Generative Model(מודל גנרטיבי): המודל המשמש בתור שיטת ההצפנה. כאן משתמשים ברשת מסוג Deconvolution אשר המשקולות שלה מאותחלות באופן רנדומלי. המשקולות הרנדומליות משתמשות כמפתח ההצפנה. הרשת מקבל את וקטור הייצוג מהמקודד ויוצרת ממנו תמונה חדשה שלעין האנושית תראה כמו רעש מוחלט. התמונה החדשה נשלחת למודל בענן שמעבד אותה כמו תמונה רגילה ושולח את וקטור התחזיות שלו חזרה.
3. The Internal Inference Network (INN) - רשת feed forward קטנה המקבל כקלט את הייצוג של התמונה המקורית יחד עם וקטור התחזיות שהמודל בענן החזיר על התמונה המוצפנת. הרשת מוציאה כפלט את התיוג האמיתי של תמונת המקור.

חשוב לשים לב שהמודל בענן אינו מקבל את התמונה המקורית אלא את התמונה אחרי תהליך הגנרוט. דבר נוסף לשים לב אליו הינו שהמודל המגנרט אינו מאומן אלא מאותחל בסט משקולות חדש ורנדומלי בכל פעם. למה כך? ובכן בתחום ההצפנה יש את מושג מפתח ההצפנה שבעזרתו מצפינים את המידע. אם מישהו יידע את המפתח הוא יוכל להשתמש בו כדי לשחזר את המידע המוצפן. לכן, חשוב שיהיה קשה להשיג את המפתח. במקרה של המודל המגנרט, מפתח ההצפנה הינו משקולת המודל. אם מישהו ישיג את המשקולות של המודל הוא יוכל להשתמש בהן כדי להעתיק את המודל ולהשתמש בו לתהליך ההפוך - מעבר ממרחב האמבדינג חזרה לתמונה המקורית. כדי למנוע זאת, מאתחלים רנדומלית את המשקולות בכל פעם כדי להקטין את הסיכוי שמישהו ישיג בנקודת זמן מסוימת בדיוק את המשקולות שיצרו תמונה מסוימת. התמונה המוצפנת שנוצרת מהמודל הגנרטיבי תראה כמו רעש מוחלט לעין האנושית כפי שניתן לראות בתמונה 1:



**תהליך האימון:**

הרכיב היחיד שצריך אימון הינו מודל ה-INN. כותבי המאמר טוענים שבשביל אימון המודל יש צורך בדאטהסט קטן בלבד של תמונות מתויגות או לחילופין דאטהסט ציבורי שזמין באינטרנט. תמונות אלו נשלחות למודל בענן שמחזיר את וקטור ההסתברויות (של הקטגוריות) עבור משימת סיווג. הקידוד של התמונה יחד עם וקטור ההסתברויות (כלומר שני וקטורים) משמשים כדגימה אחת לאימון מודל ה-INN. תהליך האימון מוצג בתמונה 1.



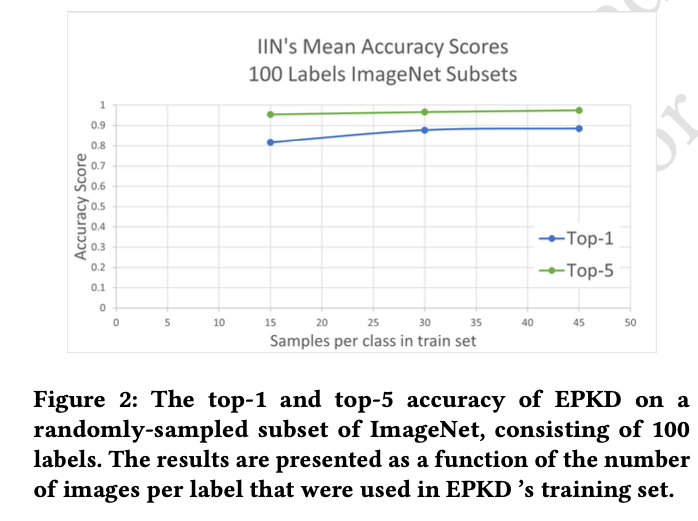
איך אבל מודל ה-INN יודע לקבל מידע מוקטור ההסתבוריות של מודל אותו הוא לא מכיר? אובחן מדובר בשילוב של שתי שיטות משני תחומים.

הראשונה שייכת לתחום ה-[Adversarial Machine Learning](https://en.wikipedia.org/wiki/Adversarial_machine_learning) ונקראת Membership Inference. בשיטה זו קיים משתמש הרוצה לנסות לגנוב מידע על אופן אימון המודל. במקרה זה מדובר ניצול של ידיעה של האם דגימה מסוימת היא חלק מסט האימון של המודל או לא. על ידי ניתוח וקטור ההסתברויות של המודל, ניתן לראות דפוסים בהם המודל מתנהג אחרת עבור דגימות מסט האימון וכאלו שלא. כדי לבצע את הניתוח מאמנים מודל שקובע עבור כל קלט (וקטור הסתברויות ותמונה) האם התמונה היא חלק מסט האימון. עבודה זו לוקחת השראה מתחום זה. הסיבה לכך הינה שגם בשיטה המוצעת כאן, הקלט הינו וקטור הסתברויות והדגימה שאליה שייך וקטור זה. השוני הוא הפלט, שכן בשיטה המקורית משתמשים במודל כדי לחזות האם דגימה שייכת לסט האימון או לא וכאן מנסים לחזות את הסיווג האמיתי של הדגימה מוקטור ההסתברויות. בעבודה זו, ניתן לראות שיש חפיפה שכן מדובר באותו קלט אך פלט שונה.

השיטה השניה היא מתחום ה-Distillation. השיטה מדברת על אימון מודל קטן המחקה מודל גדול יותר (ensemble של מודלים במקרה זה). המודל המאומן יהיה בסדרי גודל קטן יותר מהמודל אותו הוא מנסה ללמוד ובנוסף יהיה בעל ביצועים דומים. מודל ה-INN מנסה באמצעות אותה שיטה ללמוד את מודל הענן. הדגש של כותבי המאמר כפי שמוזכר הינו שמודל ה-INN הינו מודל קטן (ההנחה היא שקטן בסדרי גודל מהמודל בענן) וכולל רק שכבה חבויה אחת בלבד.

כל תמונה רגישה מועברת דרך המקודד ומוצפנת על ידי המודל הגנרטיבי. בנוסף כדי להקשות עוד יותר על פענוח המידע על ידי גורם זר, נוספו לתמונה הרגישה 9 תמונות הנבחרות רנדומליות כל פעם אשר הוצפנו גם הם. התמונות נועדו כדי להוסיף רעש, כך לא ניתן יהיה לדעת מה מהמידע הוא רגיש ומה הוא לא. בכל שליחה של מידע התמונה הרגישה והתמונות הלא רגישות עורבבו כך שמיקום התמונה הרגישה במערך שנשלח היה נבחר באופן רנדומלי כדי להקשות על זיהוי שלה. מודל ה-INN נמדד על היכולת לסווג נכון כל תמונה בין אם היא מוצפנת ובין אם לא. המדדים אשר כותבי המאמר השתמשו בהם הינם: [Top1, Top5](https://towardsdatascience.com/accuracy-and-loss-things-to-know-about-the-top-1-and-top-5-accuracy-1d6beb8f6df3) שבקיצור אומרים האם הסיווג האמיתי הוא בעל ההסתברות הגבוה ביותר או בין החמש הכי גבוהות.

את התוצאות ניתן לראות בקצרה בגרף המוצג:



**בדיקת אבטחה:**

כיוון שקיים תהליך הצפנה בכל השיטה חשוב לבדוק את איכות ההצפנה והאם ניתנת לפיצוח על ידי גורם זר. החוקרים בדקו קודם את המקרה הטריוויאלי שבו התוקף ינסה את כל הקומבינציות של פיקסל בתמונה המוצפנת כדי לשחזר את התמונה המקורית. הם ציינו שאפילו במקרה הפשוט של תמונה 2x2 ומפתח הצפנה (משקולות המודל הגנרטיבי) בגודל 2x2 עם טווח ערכים של מרחב האפשרויות הינו גדול מדי בשביל לפצח את ההצפנה ולשחזר את התמונה המקורית. לכן הם התמקדו בהתקפת סייבר יותר מתוחכמת הנקראת ([Distinguishing attacks](https://en.wikipedia.org/wiki/Distinguishing_attack) (DA. בהתקפה זו מטרת התוקף היא לא לגנוב מידע כל שהוא אלא לבצע אנליזה למידע ולראות האם הוא יכול למצוא קשרים בין חלקים שונים, למשל בין המידע המוצפן למידע שהצפינו או בין שני פריטי מידע שהוצפנו. מציאת קשר שכזה יוכל לשרת את התוקף לאחר מכן בשביל לפצח חלקים מההצפנה. במקרה שלנו נרצה להימנע מהמקרה בו התוקף יוכל ללמוד על הקשר של שתי תמונות מוצפנות. אם אכן קיים כזה התוקף יוכל להסיק שמקורן באותו סיווג (למשל שתי תמונות של כלב).

התקפה נוספת אותה בחנו הכותבים הינה Model Extraction. בהתקפה זו התוקך מנסה לשחזר את משקולות המודל (משמע לגנוב את המודל). המחברים ניסו את השיטות המתקדמות כיום בתחום תחת ההנחה שלתוקף יש גישה למודל בצורה של שליחת קלט וקבלת פלט. תחת שיטות אלו צריך מספר קטן של תמונות והגרסה המוצפנות שלהן. את זה ניתן על ידי שליחת תמונות למודל וקבלת הפלט שלו (ההצפנה במקרה זה). התוצאות האמפיריות מראות שבשביל שההתקפה תעבוד על התוקף להחזיק ב-10% דוגמאות מגודל המודל. כלומר אם גודל המודל הינו 10 מיליון פרמטרים, התוקף יצטרך בערך מיליון דוגמאות של קלט ופלט של המודל. הטענה היא שהמודל חסין לסוג זה של התקפה בשל העובדה בשל העובדה שמפתח ההצפנה משתנה לעיתים תכופות תוך כדי שימוש המודל. כלומר אם התוקף ירצה לייצר מיליון דגימות הוא לא יוכל לעשות זאת תחת ההנחה שמפתח הצפנה אחד (משקולות המודל) הוא זה שיצר את התמונות המוצפנות ולכן ההתקפה נשברת.

**סיכום:**

בעבודה זו הראנו שניתן להשתמש במודלים שונים כדי להצפין את המידע בצורה יעילה, כדי לשלוח מידע בצורה מאובטחת על גבי האינטרנט למודל גדול הנמצא בענן. אותה שיטה לא סובלת מבעיות של איטיות בהצפנת HE או בבעיות אבטחה שונות. השיטה משתמשת בשני רשתות ניורונים אשר מאותחלות באופן רנדומלי וללא אימון מקודדים תמונות. הקידוד משמשת כהצפנה של התמונה אשר נשלחת מוצפנת לענן. המודל בענן שולח לנו חזרה את וקטור ההסתברויות של התמונה המוצפנת. נשתמש ברשת ניורונים נוספת, קטנה משמעותית מהמודל בענן כדי ללמוד לשחזר את הסיווג האמיתי של תמונה בהינתן וקטור ההסתברויות והקידוד של התמונה. שיטה זו מראה את השימוש של רשתות ניורונים לא רק כמשחזרי מידע טובה אלא גם כמצפיני מידע טובים ופותחת דלת לשימוש השיטה על סוגי דאטה נוספים כגון טקסט או דאטה טבלאי.

**שיתוף פעולה:** הסקירה נכתבה בשיתוף פעולה עם עדן יבין.

## Review 89: Faster Convergence for Transformer Fine-tuning with Line Search Methods

מזמן לא כתבתי סקירה ונתקלתי במאמר חמוד הדן במשפחת גישות לאימון רשתות נוירונים שלא הייתי מודע לקיומה. אתם בטח יודעים איך היום מאמנים רשתות נוירונים ומודלי ML אחרים (=ממזערים את פונקציית הלוס שלהן). כמובן עם כל מיני שכלולים של גישת מורד הגרדיאנט הסטוכסטי (או Stochastic Gradient Descent או SGD בקצרה).

בגדול SGD היא שיטה איטרטיבית בכל איטרציה המשקלים של המודל מוזזים בכיוון של הגרדיאנט השלילי שהוא הכיוון (הלינארי) שבו פונקצית לוס קטנה ״הכי הרבה״. כאמור קיימים לא מעט שפצורים של SGD כמו ADAM, RMSProp ושיטות רבות נוספות המערבות צבירה גרדיאנט (מומנטום) שמטרתם היא להאיץ את קצב ההתכנסות של SGD ולהפוך אותו ליותר יציב. נזכיר שבכל השיטות האלו בכל איטרציה מעדכנים את המשקלים על סמך מיני-באץ' ולא דוגמא אחת כמו ב-SGD קלאסי.

המאמר שנסקור היום מציע גישה אחרת (מהירה יותר לטענת המחברים) למזעור של פונקציית לוס עבור מודלי מבוססי טרנספורמרים. קודם כל נציין כי ערך של פונקצית לוס לא בהכרח יורד (על מיני-באטץ' של איטרציה זו) אחרי עדכון של פרמטרי המודל באיטרציה של כל שיטה מבוססת מורד הגרדיאנט.

לפעמים הלוס על מיני-באטץ' עשוי לעלות אחרי העדכון גם אם אתם משתמשים בשיטות מתקדמות כמו ADAM או RMSProp. עבור SGD (כלומר MiniBatch GD) זה נובע בגדול מכך שקצב למידה (learning rate) גדול מדי. בשיטות עם מומנטום כמו ADAM המצב מורכב יותר (כי הכיוון שבו מזיזים את המשקולות הוא לא הגרדיאנט הממוצע של מיני-באטץ') אך הבעיה עדיין קיימת.

חשוב להבין שעלייה זמנית של פונקציית לוס עבור מיני-באצ'ים פה ושם היא ״לא אסון״ אם המגמה הכללית של ירידת לוס נשמרת במהלך האימון. אבל נשאלת השאלה האם שיטת אימון שתבטיח אי עלייה של פונקציית לוס בכל איטרציה תוביל לאימון יעיל ומהיר יותר בלי לפגוע באיכות המודל המתקבל בסוף האימון. זו השאלה שמחברים המאמר מנסים לענות עליה.

השיטה הנדונה במאמר מציעה דרך מאוד אינטואיטיבית לעקוף את הסוגיה הזו. כאמור אי ירידת של ערך פונקצית לוס ב-SGD נובעת מקצב למידה גדול מדי. המאמר מציע לבחור את קצב הלמידה כך שיבטיח ירידה של פונקצית לוס עבור כל המיני-באץ' בכל איטרציה של אימון.

בגדול בכל איטרצית אימון מתחילים בקצב למידה אקראי ומקטינים אותו (נגיד מחלקים ב 2) עד שהלוס אחרי העדכון יירד (על המיני-באץ'). ניתן לעשות את זה גם עם שיטות מתקדמות שנזכרו לעיל השלב האחרון הוא הזזה של משקלים בכיוון מסוים עם מקדם מסוים (קצב למידה). השיטה הזו נקראת Armijo Line Search או ALS.

המאמר מציע להפעיל את ALS על כל שכבה (בלוק של טרנספורמר) בנפרד. כלומר מחלקים משקלים של המודל ל-L קבוצות כאשר L הוא מספר השכבות ברשת. לאחר מכן מבצעים עדכון של המשקלים לכל שכבה בנפרד עם שיטת אופטימיזציה שבחרתם (SGD, ADAM etc) משולבת עם ALS. כלומר מורידים קצב למידה לכל שכבה בנפרד עד שהערך של פונקצית לוס אחרי העדכון יקטן כאשר שאר המשקלות מוקפאות.

לדעתי עם השיטה המוצעת האימון ייקח יותר זמן (כי מעדכנים כל שכבה בנפרד) אבל ניתן לעבוד עם באצ'ים גדולים יותר שזה תורם ליציבות תהליך האימון. השיטה מראה תוצאות לא רעות על פיין טיון של הטרנספורמרים.

<https://arxiv.org/abs/2403.18506v1>

היום ב [#shorthebrewpapereviews](https://twitter.com/hashtag/shorthebrewpapereviews?src=hashtag_click) סוקרים קצרות מאמר:

## Review 90, Short: Tree-Ring Watermarks: Fingerprints for Diffusion Images that are Invisible and Robust.

המאמר מציע שיטה לגילוי של תמונות נוצרות באמצעות מודלי דיפוזיה גנרטיביים.

השיטה מאפשרת למשל להגן על קניין רוחני כי באמצעותו ניתן לזהות שימוש במודלים גנרטייבים ״פרטיים״. המאמר מוסיף וקטור watermark י בעל תכונות מיוחדות (ring) לוקטור גאוס שממנו יוצרים תמונות על ידי מודלי דיפוזיה. וקטור זה ניתן לזיהוי עבור תמונות שגונרטו איתו.

וקטור watermark ניתן לזיהוי דרך הפעלת מודל ״ההופך״ מודל דיפוזיה מאומן כדי לקבל דגימה של רעש שממנו נוצרה התמונה (diffusion model inversion). וקטור watermark חסין (ניתן לזיהוי) גם אם התמונה הנוצרת סובבה, או כל הפקסלים מטושטשים או מוזזים בגורם/שיפט קבוע.

**HuggingFace:** https://huggingface.co/papers/2305.20030

**Paper:** <https://arxiv.org/abs/2305.20030>

1. ב-ResNet אמיתי, על xt בצד ימין יופעל בדרך כלל אופרטור downsample כלשהו כדי להתאים את המימדים שלו למימדי השכבה, אבל לצורך הדיון זה פרט שולי. כמו כן נתעלם כאן מאיברי bias שמופיעים לעיתים קרובות. [↑](#footnote-ref-0)
2. תזכורת: אופרטור [הדיברגנץ](https://he.wikipedia.org/wiki/%D7%93%D7%99%D7%91%D7%A8%D7%92%D7%A0%D7%A5) מוגדר במרחב אוקלידי כסכום הנגזרות החלקיות הכיווניות של שדה וקטורי או סקלרי. [↑](#footnote-ref-1)
3. חשוב לשים לב: מדובר בגרדיאנט **מרחבי** ולא בגרדיאנט של פונקציית הלוס של הרשת, כפי שהתרגלנו. בהקבלה לרשת CNN שפועלת על תמונות - הגרדיאנט הזה יהיה הגרדיאנט (הדמיוני) של התמונה, כמו שמקובל לחשוב עליו בעיבוד תמונה קלאסי. דוגמאות ניתן למצוא בשפע בוויקיפדיה ובאינטרנט. [↑](#footnote-ref-2)