סקירה זו היא חלק מפינה קבועה בה אני סוקר מאמרים חשובים בתחום ה-ML/DL, וכותב גרסה פשוטה וברורה יותר שלהם בעברית. במידה ותרצו לקרוא את המאמרים הנוספים שסיכמתי, אתם מוזמנים לבדוק את העמוד שמרכז אותם תחת השם deepnightlearners.

לילה טוב חברים, היום אנחנו שוב בפינתנו deepnightlearners עם סקירה של מאמר בתחום הלמידה העמוקה. היום בחרתי לסקירה את המאמר שנקרא:

Perceiver: General Perception with Iterative Attention

פינת הסוקר:

המלצת קריאה ממייק: חובה (!!) לאוהבי הטרנספורמרים, לאחרים מומלץ מאוד (הרעיון ממש מגניב).

בהירות כתיבה: בינונית פלוס.

רמת היכרות עם כלים מתמטיים וטכניקות של ML/DL הנדרשים להבנת מאמר: היכרות בסיסית עם ארכיטקטורת הטרנספורמר וידע בסיסי בסיבוכיות.

יישומים פרקטיים אפשריים: טרנספורמרים בעלי סיבוכיות נמוכה המותאמים לעיבוד סדרות ארוכות של דאטה (פאטצ'ים של תמונה, פריימים של וידאו, טקסט ארוך וכדומה).

פרטי מאמר:

לינק למאמר: זמין להורדה.

לינק לקוד: כאן, כאן וכאן (לא רשמיים).

פורסם בתאריך: 04.03.21, בארקיב.

הוצג בכנס: טרם ידוע

תחום מאמר:

- טרנספורמרים בעלי סיבוביות חישוב ואחסון נמוכות.

כלים מתמטיים, מושגים וסימונים:

- יסודות ארכיטקטורת הטרנספורמרים.

מבוא:

הטרנספורמר הוא ארכיטקטורה של רשתות נוירונים המיועדת לעיבוד של דאטה סדרתי. הטרנספורמרים הוצעו במאמר משנת 2017 הנקרא Attention is All You Need. מאז השתלטו הטרנספורמרים על עולם ה NLP והפכו לארכיטקטורת ברירת המחדל שם. הטרנספורמרים משמשים לבניית ייצוגי דאטה חזקים (pretraining) שלאחר מכן ניתן לכייל אותם (fine tuning) למגוון משימות downstream.

בתקופה האחרונה, התחילו הטרנספורמרים את פלישתם גם לתחום של הראייה הממוחשבת. בין המאמרים שהשתמשו בטרנספורמרים למשימות שונות בדומיין התמונות ניתן למנות (An image is worth 16×16 words), ושלושה מאמרים שסקרנו לאחרונה (TransGAN, DETR ו- Pretrained Image Transformer). לאחרונה אנחנו רואים שימוש בטרנספורמרים גם למשימות עיבוד וידאו Knowledge Vision Transformers. נזכיר שבדרך כלל הקלט לטרנספורמרים במשימות הראייה הממוחשבת הינם הפאטצ'ים של תמונת הקלט.

עם זאת קיימים מספר אתגרים המונעים שימוש נרחב יותר בטרנספורמרים בדומיין הויזואלי.

- **התלויות הלוקאליות האינהרנטיות שקיימות בתמונות.**

רשתות קונבולוציה, "המכבות" כמעט בכל משימה של הראייה הממוחשבת, מנצלות את התלויות (קשרים) הלוקאליות הקיימות בתמונות על ידי שימוש בפיקסלים סמוכים בלבד לחישוב פיצ'רים בשכבות הנמוכות. לעומת זאת, מבנה הטרנספורמרים אינו מאפשר לבנות ייצוגים לוקאליים כאלו מאחר וייצוג הדאטה בטרנספורמר הקלאסי נבנה באמצעות ניתוח קשרים בין כל חלקי הדאטה בו זמנית (להסבר מפורט על הטרנספורמר ראו TransGAN). על קושי זה ניתן להתגבר על ידי מנגנון אתחול משקלים מתוחכם (ראה TransGAN). יש עבודות שמשתמשות בשכבות קונבולוציה כשלב מקדים לבניית ייצוגים של פאטצ'ים לפני הזנתם לטרנספורמר).

- **סיבוכיות חישובית ריבועית של הטרנספורמרים במונחי אורך הקלט.**

כאמור, הטרנספורמר בונה ייצוג של דאטה באמצעות ניתוח של קשרים בין כל חלקי הקלט המבוצע באמצעות מנגנון הנקרא Self-Attention(SA) - הלב של הטרנספורמר. זאת אומרת, אנו צריכים לבצע חישוב עבור $O(M^2)$ זוגות של איברי הקלט עבור קלט באורך M. זה עלול להיות מאוד בעייתי מבחינת משאבי אחסון וזמן עיבוד הנדרשים לכך עבור תמונות ברזולוציה גבוהה (עקב מספר הפאטצ'ים הגבוה). דרך אגב, בשנתיים האחרונות יצאו מספר עבודות המציעות וריאנטים זולים יותר חישובית של הטרנספורמר כמו Linformer

Reformer, ומאמר שסקרתי לאחרונה Performer אולם למיטב ידיעתי, גרסאות אלה טרם הצליחו להשוות לרמת הביצועים של הטרנספורמר הקלאסי במגוון משימות.

תמצית מאמר:

כמו שהוסבר ב-TransGAN הסיבוכיות הריבועית של הטרנספורמר (למעשה של מנגנון Self Attention) היא התוצאה של מכפלה (נסמן אותה ב-**L**) של מטריצות **Q=Q'X** ומטריצת **K=K'X** המשוחלפת כאשר **Q', K'** הם מטריצות Query ו-Key ו-**X** היא מטריצה המייצגת קלט לטרנספורמר. הגודל של מטריצות **Q** ו-**K** הוא MxD כאשר M הוא אורך סדרת הקלט ו-D הוא מימד ייצוג הדאטה. מכאן קל לראות בבירור מאיפה צצה הסיבוכיות של O(M^2) של SA. נזכיר שהפלט של SA מחושב כ- **LV**, כאשר **V=V'X** ו-**V'** היא מטריצת Value.

להבדיל מרוב המאמרים המציעים גרסאות זולות חישובית של הטרנספורמר על ידי קירובים שונים לתוצאה של מנגנון SA, המאמר הנסקר מציע לתקוף את הבעיה מכיוון שונה לגמרי. המאמר מציע ללמוד (!!) את מטריצת **Q** במקום לחשב אותה מהקלט. זה מאפשר לקבוע את הגודל של **Q** להיות הרבה יותר קטן מאורך הקלט M, כך שסיבוכיות חישוב המכפלה של **Q** ב-**K** לא תהיה ריבועית ב-M אלא O(MN).

רעיון בסיסי:

המאמר מציע לחשב את **Q** בצורה **Q'A**, כאשר **A** היא מטריצה נלמדת, הנקראת מערך לטנטי (latent array). מטריצות **V** ו- **K** מחושבות בצורה מאוד דומה למנגנון SA המקורי. לאחר מכן במקום לחשב את הביטוי עבור Self-Attention הקלט **X**, המאמר מחשב את מה שנקרא Cross-Attention בין הקלט **X** לבין המערך הלטנטי **A**. גודל המערך הלטנטי **A** הרבה יותר קטן מגודל הקלט - וכך נמנעת הסיבוכיות הריבועית במונחי אורך הקלט.

הערה: מנגנון Cross-Attention (CA) הוצג לראשונה במאמר BERT ושימש לחישוב קשרים בין הפלט של האנקודר של BERT לבין פלטי ביניים של הדקודר במשימות כמו תרגום אוטומטי או Text Summarization.

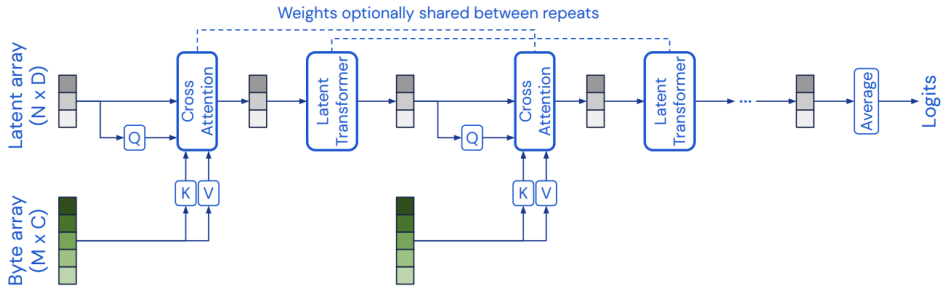

Figure 1. The Perceiver is an architecture based on attentional principles that scales to high-dimensional inputs such as images, videos, audio, point-clouds (and multimodal combinations) without making any domain-specific assumptions. The Perceiver uses a cross-attention module to project an input high-dimensional byte array to a fixed-dimensional latent bottleneck ($M \gg N$) before processing it using a stack of transformers in the low-d latent space. The Perceiver iteratively attends to the input byte array by alternating cross-attention and latent transformer blocks.

תקציר מאמר:

כעת נסביר את מבנה הקלטים למנגנון CA במאמר הנסקר. מטריצות **K** ו-**V** נבנות בצורה זהה למנגנון SA המקורי - כלומר באמצעות הכפלת הקלט במטריצות **V'** ו-**K'** הנלמדות, בהתאמה. מכיוון שאנו כבר לא מוגבלים עם הסיבוכיות הריבועית (במונחי אורך הקלט) ניתן לקחת סדרת קלט ארוכה יותר מאשר בטרנספורמר הרגיל. למשל כאשר הקלט לטרנספורמר הוא תמונה ברזולוציה גבוהה, נהוג לחלק אותה לפאטצ'ים בגודל 16x16, בגלל מגבלת הסיבוכיות של הטרנספורמר המקורי. שימוש במערך לטנטי **A**, שניתן לבחור את גודלו לפי משאבי חישוב העומדים לרשותנו, מסיר מגבלה זו, המונעת מאיתנו להכניס לטרנספורמר סדרות קלט ארוכות. למעשה, המאמר מציע "לשטח" את הקלט ולהפוך אותו ל"מערך בתים" (byte-array) לפני שמכפילים אותו במטריצות Key ו-Value. אם הקלט הוא תמונה, כל איבר במערך הבתים מכיל את ערכו של הפיקסל (!!).

כמובן ניתן להכניס ל-Perceiver גם סדרות אודיו ארוכות או קטעי וידאו. יתרה מזו, המאמר טוען שניתן ל-Perceiver גם סדרות וידאו יחד(!!) עם אודיו במקשה אחת, דבר שלא היה אפשרי בגרסאות הקודמות של הטרנספורמרים (שדרשו התאמות לארכיטקטורה של הטרנספורמר בהתאם לסוג הקלט). כלומר, הארכיטקטורה שהוצעה במאמר היא אגנוסטית (!!) לסוגים רבים של קלט וזה דבר חזק מאוד בעצמו.

ארכיטקטורה של Perceiver: פרטים

לאחר שהבנו את העקרונות הבסיסיים של ארכיטקטורת Perceiver, ניתן לתאר את שאר הפרטים לגביה. לאחר חישוב של Cross-Attention בין המערך הלטנטי לבין הקלט, הפלט (של CA) מוזן לטרנספורמר רגיל, הנקרא במאמר הטרנספורמר הלטנטי (LTr -latent transformer). חשוב לציין כי הגודל של הפלט של מנגנון CA אינו תלוי בגודל המקורי של הקלט אלא בגודל של המערך הלטנטי (הנקבע כאמור בהתאם למשאבי חישוב זמינים). מכיוון שהגודל של המערך הלטנטי בדרך

כלל הרבה יותר קטן מגודל הקלט המקורי, ניתן "להעביר" אותו דרך LTr בסיבוכיות סבירה. ארכיטקטורה של LTr דומה לארכיטקטורה של GPT-2 ומורכבת מהדקודר של מהמאמר המקורי.

הפלט של LTr שוב מוזן למנגנון CA בדומה למה שעשינו לפני כן (לשם כך משתמשים שוב במטריצות **K** ו-**V** המחושבים מהקלט המקורי המשוטח). הפלט של CA מוזן ל-LTr נוסף כאשר השילוב הזה (CA+LTr) יכול לחזור על עצמו פעמים רבות במטרה ליצור ארכיטקטורה עמוקה ועוצמתית המסוגלת לבנות ייצוגים חזקים לקלטים במספר דומיינים. נציין כי כל ה-LTrs יכולים להשתמש באותם משקלים (shared weights), משקלים שונים לכל אחד LTr, או כל אופציית ביניים שהיא (למשל 3 סטים של משקלים לכולם). ניתן לחשוב על Perceiver כרשת נוירונים רב שכבתית כאשר כל השכבה מורכבת מ-CA ו-LTr.

פינת האינטואיציה:

ניתן להסתכל על מערך הלטנטי כסט של "שאילתות נלמדות" לגבי הקלט. דוגמא של "שאילתה" אפשרית יכולה להיות: תמדוד את הקשרים בין פאטץ' p שבמרכז התמונה לכל הפאטצ'ים בתוך פאטץ' גדול יותר, המכיל את p .דוגמא של "שאילתה" אפשרית יכולה להיות: תמדוד את הקשרים בין פאטץ' p שבמרכז התמונה לכל הפאטצ'ים בתוך פאטץ' גדול יותר, המכיל את p (בשכבת CA הראשונה). בשכבות עמוקות יותר של Perceiver המערך הלטנטי (השאילתות) כבר תלוי בערכים המחושבים בשכבות הנמוכות, ובדומה לרשתות קונבולוציה, מנסות לשערך את הפיצ'רים היותר סמנטיים של התמונה. ניתן גם לחשוב על Perceiver כ-RNN רב שכבתית (כאשר כל שכבה מקבלת את הקלט כולו).

קידוד מיקומי (positional encoding):

כמו שכבר ציינתי בסקירותיי הקודמות של מאמרים בנושא הטרנספורמרים, מנגנוני SA ו-CA הם אגנוסטיים לסדר איבריו בסדרות הקלט. כלומר ייצוג איבר סדרת קלט, המופק באמצעות CA ו-SA, יישאר ללא שינוי גם לאחר הפעלת פרמוטציה כלשהי על סדרה/ות הקלט. כמובן שמצב זה אינו סביר עבור תרחישים שיש בהם סדר אינהרנטי בין איברי סדרת הקלט (למשל שפה טבעית, תמונה, וידאו, אודיו ועוד).

כדי להעביר למנגנונים של CA ו-SA את המידע לגבי מיקום של כל איבר בסדרה, מוסיפים לסדרת הקלט את מה שנקרא הקידוד המיקומי (PE). שמטרתו של PE היא לקודד מיקומו (היחסי) של כל איבר בסדרת הקלט. עבור CA המאמר משתמש ב-PE דומה לזה שהוצע ב-BERT (המבוסס על פיצ'רי פוריה). לעומת זאת עבור מנגנון SA ב-LTr, המאמר משתמש ב-PE נלמדים.

הנושא של הקידוד המקומי נדון בהרחבה במאמר (נעשו בן כמה שינויים מעניינים והמחברים ניסו לתת אינטואיציה לסיבת שיפור הביצועים).

הישגי מאמר: המאמר השווה את הייצוגים המופקים באמצעות Perceiver עם מספר שיטות אימון self-supervised (מוסיפים שכבה לינארית לרשת המפיקה את הייצוג (המאומנת), מאמנים את המשקלים של שכבה זו ובודקים ביצועים) וגם עם שיטות SOTA supervised במספר דומיינים:

- תמונות
- וידאו
- אודיו
- וידאו עם אודיו
- ענני נקודות

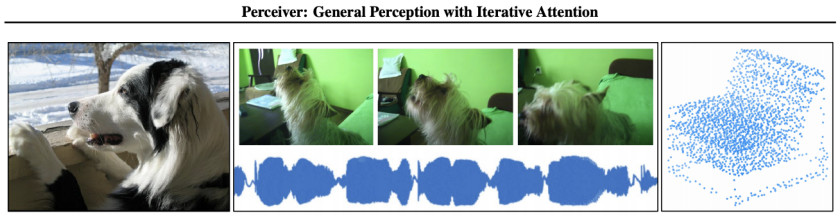

Figure 2. We train the Perceiver architecture on images from ImageNet (Deng et al., 2009) (left), video and audio from AudioSet (Gemmeke et al., 2017) (considered both multi- and uni-modally) (center), and 3D point clouds from ModelNet40 (Wu et al., 2015) (right). Essentially no architectural changes are required to use the model on a diverse range of input data.

עבור כל הדומיינים Perceiver הצליח להפגין ביצועים יותר טובים מכל שיטות unsupervised שהם בדקו (כולל אלו שמבוססים על הטרנספורמרים). נציין כי חלק משיטות, ש-Perceiver "התגבר עליהן", נבנו עבור דאטה מדומיין ספציפי תוך ניצול התכונות האינהרנטיות של הדאטה בדומיינים אלו (כמו ResNet בדומיין של תמונות). עם זאת הביצועים של Perceiver בכל דומיין היו טיפה פחות טובים מהשיטות supervised המנצלים את התכונות של דאטה בדומיינים אלו.

ResNet-50 (He et al., 2016)	76.9
ViT-B-16 (Dosovitskiy et al., 2021)	**77.9**
ResNet-50 (RGB+FF)	73.5
ViT-B-16 (RGB+FF)	76.7
Transformer (64x64)	57.0
Perceiver	76.4

Table 1. Top-1 validation accuracy (in %) on ImageNet. Methods shown in red exploit domain-specific grid structure, while methods in blue do not. The first block reports standard performance from pixels – these numbers are taken from the literature. The second block shows performance when the inputs are RGB values concatenated with Fourier features (FF) of the xy positions – the same that the Perceiver receives. This block uses our implementation of the baselines. The Perceiver is competitive with standard baselines on ImageNet while not relying on domain-specific architectural assumptions.

	Fixed	Random	Rec. Field
ResNet-50 (RGB+FF)	39.4	14.3	49
ViT-B-16 (RGB+FF)	61.7	16.1	256
Transformer (64x64)	57.0	57.0	4,096
Perceiver	**76.4**	**76.4**	50,176

Table 2. Top-1 validation accuracy (in %) on **permuted** ImageNet. "Fixed" = permuted with a constant permutation for all images over the dataset. "Random" = random, per-example permutation. Methods that make strong assumptions about the structure of 2D data fare poorly when this structure is removed. All methods receive identical input features (RGB+FF). We also show the receptive field of the input units for each model on the right, in pixels. Note that both Transformer and Perceiver have a global view of all inputs in each first layer unit. ResNet-50 starts with a 7x7 convolution, hence each unit sees 49 pixels, and ViT-B-16 inputs 16x16 patches, hence 256 pixels are seen by each first layer unit.

נ.ב. מאמר מאוד מעניין, מציע שיטה מגניבה להתגבר על הסיבוכיות הריבועיות של הטרנספורמר. הארכיטקטורה המוצעת במאמר אגנוסטית למבנה של קלט, ויכולה לשמש כמו שהיא לבניית ייצוגי דאטה בדומיינים מגוונים.

הפוסט נכתב על ידי מיכאל (מייק) ארליכסון, PhD, Michael Erlihson.

מיכאל עובד בחברת סייבר Salt Security בתור Principal Data Scientist. מיכאל חוקר ופועל בתחום הלמידה העמוקה, ולצד זאת מרצה ומנגיש את החומרים המדעיים לקהל הרחב.