⚡️🚀המאמר היומי של מייק -03.11.24: ⚡️🚀  
TOKENFORMER: RETHINKING TRANSFORMER SCALING WITH TOKENIZED MODEL PARAMETERS

אוקיי, זה מאמר די לא צפוי עם רעיון פשוט להבנה ובאופן די מפתיע (לפחות אותי) גם עובד (לפי מחברי המאמר כמובן). מכירים את הטרנספורמרים או שאיך שניה אופנתי לקרוא שנאים בעברית. בלוק הטרנספורמר (אבן הבניין של ארכיטקטורה זו) מורכב ממנגנון attention (אמרו לא לקרוא לזה ״תשומת לב״ כי זה לא נשמע טוב) יחד עם 2 שכבות feedforward או FF (יש אקטיבציה לא לינארית רק בשכבה הראשונה מהן). בנוסף יש כמה שכבות נרמול (לבחירתכם) וזה כל הקסם.

אז המחברים של המאמר מציע שינוי מעניין בארכיטקטורה זו (שמשגשגת לנגד עיננו כבר 7 שנים) שינוי די לא צפוי. מה שהוביל אותם לשינוי הזה זה קושי של השינוי המימדים של שכבות הקלט ופלט לבלוק טרנספורמר שמחייב אימון מחדש של כל המודל (המורכב ממספר בלוקי הטרנספורמר). אני לא משוכנע שזה נכון ד״א.

אז כדי להתמודד עם הסוגיה הזו המחברים הציעו להחליף את שכבות FF במנגנון שקיבל שם PAttention שמחשב משהו שקצת דומה ל-attention. אמנם לא באמת דומה כי אין שם השוואה בין הייצוגים השונים של טוקנים (המופקים באמצעות מטריצות Q ו- V כאשר ההשוואה מחושבת דרך מכפלה פנימית שלהם ונרמול עם softmax). מה ש-PAttention באמת הוא חישוב המשקלות של FF - כאן צריך להזכיר כי בלוק השנאי הרגיל הוא גם שכבה fully-connected כאשר משקלותיה תלויות בקלט (דרך מנגנון ה-attention המקורי של השנאי).

מה ש-PAttention עושה הוא חישוב של המשקלים האלו באופן הבא:

מכפלה של ייצוגי הטוקנים במטריצה K\_P נלמדת

נרמול רגיל של הוקטור המתקבל (מחלקים בשורש של הנורמה הריבועית)

הפעלת פונקציית אקטיבציה לא לינארית (זה GeLU שמוגדר עם erf למי שמתעניין)

הכפלה במטריצת V\_P נלמדת

אז מה יש לנו בסוף? שכבת fully connected עם משקלים מחושבים בדרך טיפה שונה מה-attention הרגיל במקום שכבת FF שיש לנו בשנאי. מפעילים את ה-PAttenttion אחרי בלוק attention הרגיל.

וכן זה מאפשר לשנות את מספר מימדים של המטריצות הפנימיות של השנאי ללא retraining מלא של המודל (על ידי שרשור המטריצות החדשות הנלמדות של PAttention עם הישנות שכבר אומנו)..

וכל הסיפור הזה עובד…

https://arxiv.org/abs/2410.23168