המאמר היומי של מייק - 18.12.24:  
Byte Latent Transformer: Patches Scale Better Than Tokens

כמובן לא יכולתי לפספס את המאמר הזה שהתפרסם לפני כמה ימים וגרם ללא מעט תהודה בקהילת AI. המאמר מציע להחליף את הטוקנייזר הסטטי שיש בכל מודל השפה במנגנון דינאמי שבונה את הטוקנים החדשים (שקיבלו שם פאצ'ים) כלומר כזה שבונה אותם בתלות בהקשר (contextualized).

הרציונל כאן הוא די ברור הרי לפעמים יש מקרים שחיזוי של כמה טוקנים הבאים הוא די ברור וניתן לעשות אותה כמקשה אחת (כלומר לאחד את כל הטוקנים לטוקנים אחד ארוך או פאץ' לפי שמו במאמר). ולפעמים המצב הוא הפוך והיינו רוצים לחזות בצורה בגרנולריות קטנה יותר. וכמובן שזה בלתי אפשרי במודל שיש בהם מילון טוקנים קבוע.

כאמור המאמר מציע להכניס דינמיות בבניית פאצ'ים (הטוקנים החדשים). איך הוא עושה את זה. לדאטהסט נתון המאמר מאמן מודל רדוד יחסית ברמה של בטים (bytes) כאשר המטרה של המודל היא לחזות את הבייט הבא. ואז במודל הגדול שלנו הם קובעים את גבולות הפאץ על סמך אנטרופיה של הבטים. כלומר אם האנטרופיה של הבייט או גדולה מסף מסוים או חוותה עליה מעל סף מסוים מעל האנטרופיה של הבייט הבא, פותחים פאץ' חדש. אחרת ממשיכים את הפאץ' הנוכחי.

אבל איך כל הסיפור הזה עובד - כמו שאמרתי המודל הוא byte-level כלומר הוא מאומן לחזות את הבייט הבא בטקסט. אבל במקום להסתכל על הקונקסט בתור מערך של טוקנים המחברים מציעים להחליף אותו בפאצים דינמיים נקבעים על סמך האנטרופיה כמו שהסברתי קודם.

בנוסף לפאצים המאמר משתמש גם בייצוג של בטים באמצעות n-grams (לוקחים n-grams לבייט נתון מ n=3 עד n=8, מפעלים איזה פונקציית האש, סוכמים ומנרמלים). את התוצאה הופכים לווקטור (המאמר לא מפרש איך- רק מזכיר שיש איזו שכבה לינארית המעורבת בזה) ומזין אותו למה שקרוי במאמר Encoder Multi-Headed Cross-Attention (נקרא לזה לפשטות EMHCA).

מטרתו של EMHCA היא לשלב את ייצוגי הפאצ'ים עם ייצוגי הבטים שלהם(כל פאץ מתחשב רק בייצוגי הבטים שלו ולא של האחרים). הייצוג ההתחלתי של כל פאץ מחושב כ-pooling (כלומר ממוצע) של ייצוגי הבטים שלו (נזכיר זה כל פאץ הינו מערך של הבטים). כלומר אנו בונים ככה ייצוג של כל פאץ' המתחשב רק במה שיש בתוכו (internal representation).

אז ייצוג הבטים וייצוגי הפאצ'ים מוזנים ל-EMHCA שזה למעשה טרנספורמר די רדוד (עם מעט שכבות) שמטרות לבנות ייצוג תלוי הקשר שפאצ'ים כתלות בבטים שלו. כלומר גם ייצוגי הבטים הם keys and values כאן כאשר ה-queries הם ייצוגי הפאצים. כאמור מה שיוצא מהטרנספורמר הרדוד הזה הוא ייצוגי הפאצ'ים. נציין ש- EMHCA פולט גם ייצוגי הביטים בסוף (לא הצלחתי להבין איך זה נבנה).

כל אלו מוכנסים לטרנספורמר יותר עמוק וכבד חישובית היוצר ייצוג יותר ״עמוק״ של הפאצים. בשלב האחרון יש את ה-Local Decoder שהופך את ייצוגי הפאצ'ים יחד עם ייצוגי הבטים לייצוגי הבטים הסופיים שמהם נחזה הבייט הבא. זה גם טרנספורמר רדוד אבל הפעם ייצוגי הפאצ'ים הם keys and values וייצוגי הבטים הם ה-queries.

המאמר טוען לכל מיני יתרונות של השיטה המוצעת כמו יכולת לחזות יותר טוקנים לעלות אינפרנס קבועה, ומציגה דיוק משופר באימון המודלים.

אוקיי, חייב להגיד שהמאמר לא כתוב כזה טוב - יש דברים שלא הוסברו בצורה ברורה (למיטב ידיעתי כמובן). אני רק מקווה שהצלחתי להבין אותו נכון….

https://arxiv.org/abs/2412.09871