# Review 184: Mamba Series, An Intro

**Paper: https://arxiv.org/abs/2408.11451v4**

כמו שהבטחתי היום אנחנו מתחילים את סדרת הסקירות הקצרות שתוביל אותנו בסופו של דבר לדובדבן שבקצפת שזה תיאור של ארכיטקטורה בשם ממבה (Mamba) שעשתה הרבה רעש לאחרונה. היום אספק לכם מבוא כללי על מה שאנחנו הולכים לדבר בשבוע הקרוב.

אז מה זה ממבה? כאמור ממבה הינה ארכיטקטורת רשת נוירונים שבאה לתת מענה לחיסרון הבולט של הטרנספורמרים והוא הסיבוכיות הריבועית במונחי אורך הקלט (עבור קלט טקסטואלי אורכו הינו מספר הטוקנים בחלון ההקשר). נכון שלאחרונה הוצעו מספר שיפורים למנגנון תשומת הלב של הטרנספורמר (שהוא לב הבעיה) כמו FlashAttention2 אבל עדיין הטרנספורמרים מתקשים לעבוד בצורה המיטבית עם דאטה בעל אורכי הקשר בסדר גדול של מיליוני טוקנים.

אז איך אנחנו נתמודד עם הסיבוכיות הריבועים של מנגנון תשומת הלב של הטרנספורמרים? הצעירים שבינינו זוכרים שפעם היתה לנו ארכיטקטורה הנקראת (RNN (Recurrent Neural Networks שבהם לא היתה לנו בעיה של הסיבוכיות הריבועית (היו שם חסרונות אחרים שנדון בהם בהמשך). ה-RNN והשכלולים שלו כגון GRU ו-LSTM לא היו צריכים לקחת בחשבון את ייצוג של כל פיסות דאטה (נגיד טוקנים בטקסט) בצורה מפורשת (כפי שמנגנון תשומת הלב של הטרנספורמים עושה). במקום זאת הוא היה דוחס את המידע על הטוקנים (הייצוגים והקשרים ביניהם) באמצעות וקטור המצב (ב-LSTM יש בנוסף עוד כמה וקטורים האחראים על דחיסה של זכרון).

אז אם הצלחנו לדחוס את כל המידע הטמון בטוקנים הקודמים בצורה טובה אז לא צריך להתחשב שום מידע באופן מפורש בזמן אימון ובזמן היסק (inference) של טוקן הבא. אמנם אם כל הזכרון שלנו נמצא בוקטור הדחוס הזה אז נשתמש בו במקום להתחשב כל הטוקנים הקודמים. אולם יש בגישה הזו שתי בעיות עיקריות:

ארכיטקטורות RNN לא הצליחו לדחוס בצורה טובה את הטוקנים הקודמים כאשר אורך חלון הקשר ארוך וזו הייתה הסיבה העיקרית שארכיטקטורות אלו נכשלו במשימות הכרוכות בעיבוד קטעי טקסט ארוכים. הרי אם המודל לא מסוגל לזכור את המידע מהטוקנים הקודמים, לא ניתן לצפות ממנו ביצועים טובים בחיזוי טוקן הבא.

בטח כבר שמעתם שארכיטקטורות RNN הן לא scalable. מה זה בעצם אומר? כאשר אנו מבצעים אימון של מודל שפה המשימה היא לחזות חלקי קלט שאנו מסתירים (ממסכים) מהמודל. עם טרנספורמרים יש לנו יכולת לחזות את כל הטוקנים הנסתרים בצורה מקבילית עלי ידי שימוש בו זמני במסיכות שונות (כל פעם ממסכים רק את מה שצריך). ב-RNNs זה בלתי אפשרי כי עבור חיזוי של כל טוקן אנו צריכים לחשב את ייצוג הזכרון שלוקח בחשבון את כל הטוקנים שהיו לפניו. כלומר עבור טוקן מספר 1000 אנו צריכים לבצע חישוב סקוונציאלי (אחד אחרי השני) עבור 999 טוקנים שהיו לפניו. פעולה זו לא ניתנת למקבול עקב נוכחות של פונקציות לא לינאריות בחישוב ייצוג הזכרון (ממבה עוקף את המכשול הזה באלגנטיות). כמובן שזה לא יעיל ומהווה מכשול משמעותי בניצול יעיל של משאבי חישוב (GPUs).

עכשיו נשאלת השאלה האם אנחנו יכולים להתגבר על הסיבוכיות הריבועית של הטרנספורמרים ובאותו זמן להיפטר משני חסרונות שתיארנו בפסקה הקודמת? זו בדיוק השאלה המחקרית העיקרית שנדון בה בסקירות הבאות שיובילו אותנו לארכיטקטורת ממבה הנחשקת.

לבסוף אתן לכם כמה טיזרים קטנים בנוגע למה שאתם הולכים לראות בימים הקרובים:

ארכיטקטורות שאנו הולכים לדבר עליהן מאפשרים שני משטרי הפעלה:

אימון מקבילי כמו עם טרנספורמרים במהלך אימון המודל

היסק מהיר כמו עם ה-RNN-ים

באופן די מפתיע ארכיטקטורה זו היא בעלת מבנה דומה לרשת קונבולוציה רק שהקרנלים של קונבולוציות אלו הן מאוד ארוכות

ארכיטקטורות אלו שואבות השראה ממערכות דינמיות לינאריות וקשרות לשערוך של פונקציות על ידי פולינומים אורתוגונליים.