⚡🚀 המאמר היומי של מייק 30.05.2024 🚀⚡
2BP: 2-Stage Backpropagation

אנו יודעים שהמודלים העמוקים גדולים היום מדי כדי להיכנס לזכרון ram של gpu אחד. עקב כך מחלקים את משקלי המודל בין הgpus השונים (sharding). זה פותר צוואר בקבוק אחד (זכרון) אבל כתוצאה מכך נוצר צוור בקבוק אחר בחישוב של backprop, המאמר הנסקר פיתח שיטה למקבל את חישוב הגרדיאנטים במהלך backprop ובכך מקל על צוואר הבקבוק הזה.

מאמר: https://arxiv.org/pdf/2405.18047

⚡🚀 המאמר היומי של מייק 31.05.24: ⚡🚀
Transformers Can Do Arithmetic with the Right Embeddings
אנו יודעים שמודלי שפה גדולים לא מצטיינים בלחשב ביטויים מתמטיים בטח כאלו המכילים מספרים עם הרבה ספרות. גם אם מאמנים אותם על מיליוני דוגמאות עדיין מסתבכים להכליל אותם למספרים גדולים. המאמר מציע להוסיף positional encoding למספרים שמטרתם לספק למודל שפה מרחק של כל ספרה מתחילת המספר. וזה עובד לא רע.

רפו: https://github.com/mcleish7/arithmetic
מאמר: https://arxiv.org/abs/2405.17399

⚡🚀המאמר היומי של מייק 01.06.24: ⚡🚀
The Evolution of Multimodal Model Architectures

אתם יודעים שאני אוהב לכתוב סקירות אבל בד״כ אני סוקר מאמר אחד. כאן יש לכם סקירה של תחום שלם שהוא מודלים מולטי-מודליים כלומר כאלו שיודעים ״לטפל״ בסוגי דאטה שונים (שפה, תמונות, אודיו וכדומה). המאמר נותן סקירה היסטורית על ארכיטקטורות של מודלים מולטי-מודליים ומחלק אותם ל 4 קטגוריות רחבות שמתחלקות לתת-קטגוריות כמובן. מאמר שיכול לעשות לכם קצת סדר בנוגע לתחום המגניב הזה.

טלגרם: https://t.me/MathyAIwithMike/60
טוויטר: https://x.com/MikeE_3_14/status/1796823310459666491
מאמר: https://arxiv.org/abs/2405.17927

⚡🚀המאמר היומי של מייק 02.06.24: ⚡🚀
LLaMA-NAS: Efficient Neural Architecture Search for Large Language Models

פעם הנושא של Neural Architecture Search או NAS בקצרה שעסק בחיפוש לאחר ארכיטקטורה אופטימלית של רשת נוירונים עבור משימה/משימות/דומיין היה די פופולרי אך בשנים האחרונות התחום נמצא בדעיכה. אני שמח שנתקלתי במאמר הזה שמנסה לפתח NAS עבור מודלי שפה. אני זוכר מאמרים די מגניבים שמשתמשים בשיטות RL די מגניבות לכך. אולי בעתיד NAS תהפוך למתחרה רציניות של שיטות פרונינג וקוונטיזציה.
-

מאמר: https://arxiv.org/abs/2405.18377
טלגרם: https://t.me/MathyAIwithMike/69

⚡🚀המאמר היומי של מייק 03.06.24: ⚡🚀
Better & Faster Large Language Models via Multi-token Prediction

אתם בטח שיודעים אנו רגילים לאמן מודל שפה גנרטיביים באמצעות חיזוי טוקן הבא בהינתם הטוקנים הקודמים (הקשר או קונטקסט). המאמר הזה (שקיבל די הרבה pr כשיצא) מציע לחזות כמה טוקנים עוקבים בו זמנית בהינתן הקשר. המחברים הראו שזה יכול לשפר את ביצועי המודל - זה לא מפתיע(לתחושתי) כי משימת חיזוי טוקנים מרובים דורשת מהמודל הבנה יותר מעמיקה של השפה. השיטה גם עשויה לתרום להאצת זמן ריצה והרווחים גדלים עם גודל המודל.
מאמר: https://arxiv.org/pdf/2404.19737
טלגרם: https://t.me/MathyAIwithMike/69

⚡🚀המאמר היומי של מייק 04.06.24: ⚡🚀
Are Emergent Abilities of Large Language Models a Mirage?

היום המאמר היומי הוא מלפני שנה בערך והוא משך את תשומת ליבי בגלל שהוא חוקר מה שנקרא emergent capabilities של מודלי שפה - כלומר יכולתם ללמוד משימות חדשות. המאמר בוחן האם למודלי שפה אכן יש יכולת ללמוד משימות שהם אומנו עליהם בצורה מפורשת (פחות או יותר) או שזו אשליה הנובעת מאיך שאנו מודדים את היכולות האלו.
מאמר: https://arxiv.org/abs/2304.15004
טלגרם: https://t.me/MathyAIwithMike/76

⚡🚀המאמר היומי של מייק 05.06.24: ⚡🚀
GraphAny: A Foundation Model for Node Classification on Any Graph

כיצד לפתח מודלים foundational בתחום הגרפים?

מודלי שפה foundational שינו בצורה משמעותית את האופן שאנו בונים מודלים בתחום nlp: בהרבה מקרים הם מאפשרים פיתוח מהיר הרבה יותר (פיינטיון וכאלו). מרחב קלט משותף לכל המשימות (טוקנים) הוא מרכיב חיוני שדרכו foundational LLMs מגלמים יכולת הכללה שמאפשרת התאמתם היחסית לא מורכבת למגוון מגוון משימות NLP.

לצערנו לגרפים אין תכונה משותפת כמו טוקנים, כי כל גרף לרוב מאופיין על ידי סמנטיקה משלו מבחינת מאפיינים לייבלים, דבר שמונע את פיתוח המודלים foundational של הגרפים. האם ניתן להתגבר על זה? יש לנו התחלה: המחברים מציעים GraphAny, ארכיטקטורה foundational לביצוע משימת סיווג קודקודים בגרף. המודל יכול להכליל לגרף חדש כלשהו עם מרחבי מאפיינים ולייבלים שרירותיים, שונים בדרך כלל מאלה של הגרף שאימנו עליו.
מאמר: https://arxiv.org/abs/2405.2044
טלגרם: https://t.me/MathyAIwithMike/78

⚡🚀המאמר היומי של מייק 07.06.24: ⚡🚀
Scaling and evaluating sparse autoencoders?

המאמר הזה של openai ממשיך את הקו המחקרי של antropic (https://www.anthropic.com/news/mapping-mind-language-model) המנסה לראות איך ניתן למצוא של קונספטים (מסלולים וויזואלים) בתוך נוירונים של מודלי שפה מאומנים. המאמר של אנטרופיק בגדול טוען שיש נוירונים הנדלקים על קונספטים (נגיד גשר הזהב(מסוימים ויש כאלו שמהווים ערבוב של קונספטים.

אבל איך ניתן להציג את קונספט באמצעות וקטור? מתברר שניתן להציג כל קונספט באמצעות וקטור ארוך אך מאוד דליל(sparse). אז נוירונים המהווים ערבוב של קונספטים ניתן להציג בתור סכום משוקלל של וקטורים דלילים אלו אחרי שמטילים את הסכום על מרחב האמבדינג המקורי של הטרנספורמר.

הוקטורים הדלילים המתאימים לקונספטים ניתן להפיק באמצעות אימון של sparse autoencoder של שכבה אחת לכל כיוון כאשר הייצוג באמצע (אחרי האנקודר) הוא וקטור דליל: במהלך האימון לוקחים ממנו את k הרכיבים הגדולים ביותר - אחרי ReLu

ויש כמובן חוקי Scaling מעניינים לגבי הייצוגים האלו. מאמר מעניין.
מאמר: https://cdn.openai.com/papers/sparse-autoencoders.pdf

⚡🚀המאמר היומי של מייק 08.06.24:🚀⚡
Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality

למאמר הזה יש עוד שם והוא 🤔mamba-2. המאמר הזה מתמקד בשכלול הארכיטקטורה של ממבה המקורית שעשתה הרבה כותרות בחצי השנה האחרונה ואני הצטרפתי לחגיגה וסקרתי בערך 20 מאמרים בנושא המרתק הזה.

המאמר הזה של Albert Gu התותח ממשיך להעשיר את עולם הממבה והפעם הוא הגיע לכמה תובנות די מעניינות. הוא לראשונה מגדיר SSM בעל תכונה N-semi-separable שלמעשה מגדיר את צורתו של קרנל קונבולוציה המופעל על סדרת הקלט במוד הקונבולוציוני של SSM (כאשר משתמשים ב-SSM לאימון ממוקבל). אלחש לכם בסוד שבסופו של דבר זה מתנקז לצורתו של מטריצה A.

שנית מאמר חוקר מנגנוני ה-attention בפרט השונים למשל הקלאסי הלינארי ,כלומר ללא סופטמקס, ועם סדר שונה בביצוע פעולות בין מטריצות Q, K, ו-V. המאמר מפרק את החישוב ל- 3 שלבים "אטומיים" (שכל אחת מהם הוא מכפלות מטריצות, אך לפעמים מאוד גדולות) השלב השני והחשוב ביותר הוא מיסוך (masking) שניתן לתאר אותו גם עלי ידי מכפלות מטריצות (Kernel trick). המיסוך הקוזלי (causal) הוא חלק ממנגנון ה masking. הבחנה זו אפשרה למחברים להוכיח סוג של שקילות בין מנגנוני attention מסוימים ל-SSMs .

בנוסף הם מפתח שיטה לחישוב יעיל של קונבולוציה ארוכה (שזה הלב של SSM) בחומרה עם מטריצות הנקראת semi-separable-1 (עבור מטריצה A מצורה מסוימת).

מה יוצא לנו מכל הסיפור הזה? האצת אימון של ממבה (שזה למעשה ממבה 2) וגם פריימוורק תיאורטי למידול ארכיטקטורה העוצמתית הזו משותפת גם למנגנוני ה-attention השונים.

קריאה מהנה!
https://arxiv.org/abs/2405.21060

🚀⚡המאמר היומי של מייק 09.06.24:⚡🚀
What Do Language Models Learn in Context? The Structured Task Hypothesis.

המאמר הזה תפס את עיניי כי הוא מנסה לפתור את תעלומת in context Learning או ICL. היכולת של מודלי שפה לבצע משימות שלא אומנו עליהם באופן מפורש על לאחר הצגה של כמה דוגמאות(שאלה, תשובה) היא לא פחות ממדהימה ועדיין אין תשובה חד משמעות המסבירה מה אכן קורה שם.

המאמר בוחן 3 הסברים אפשריים ל ICL:

1. מודל שפה אשכרה "מזהה" את המשימה מכמה דוגמאות ומבצע אותה לפרומפט נתון
2. המודל לומד במהלך אימון מקדים (pre-training) לעשות meta-learning כלומר ללמוד את המשימה מכמה דוגמאות שניתנו לו
3. המודל לומד לייצג משימה חדשה כ"שילוב" של כמה משימות שלמד במהלך אימון מקדים

המחברים מוכיחים ש ההשערות 1 וגם 2 לא מתקיימות שלא משאיר הרבה אפשרויות...
https://arxiv.org/abs/2406.04216

⚡🚀המאמר היומי של מייק 10.06.24:⚡🚀
Learning to grok: Emergence of in-context learning and skill composition in modular arithmetic tasks

אחד התופעות המרתקות בלמידה עמוקה היא גרוקינג - שהיא מעבר "פתאומי" של רשתות עמוקות למצב של הכללה מהמצב של overfitting למשל אחרי אימון מאוד ארוך. הרי ידוע שאם עבור דאטהסט נתון ורשת עמוקה בעלת יכולת ייצוג גבוהה מספיק (representativeness) אחרי שלב מסוים באימון אנו נגיע ל-overfitting כלומר למצב שבו ביצועי המודל יילכו וישתפרו עבור סט האימון אולם הביצועים על סט הולידציה יספגו ירידה בביצועים.

מה שמגניב ומפתיע בגרוקינג שעבור אימון ארוך מספיק מגיע המצב שביצועי המודל על סט הוולידציה מתחילים לעלות יחד עם אלה על סט האימון כלומר המודל מגיע לשלב של הכללה אמיתית. מעניין שתופעה דומה מתרחשת בתנאים מסוימים אם אנו מגדילים את קיבולת המודל (מס' הפרמטרים) כאשר גודל הדאטהסט ומשך האימון נותרים קבועים) וגם כאשר אנו מגדילים את גודל הדאטהסט תוך שמירה של משך האימון קיבולת המודל קבועים.

למעשה תופעות אלו שייכות למשפחת double descent (יש גם multiple descent) שנחקרה רבות על ידי חוקר דגול מישה בלקין. התופעה עצמה נתגלתה לפני יותר מ 30 שנה (מי שרוצה להתעמק בנושא תעקבו אחרי https://www.linkedin.com/in/charlesmartin14/ - הוא אחד המומחים הגדולים).

אוקיי, אז מה עשה המאמר הנסקר? הוא חקר תופעת גרוקינג כאשר מתרחשת אם מגדילים את מספר המשימות (כל משימה היא סוג של רגרסיה לינארית בשדה המודולו(שארית)) שעבורן אנו מאמנים את המודל (כמובן לקחו מודל שפה). מתברר כי יש כמה משטרים (מודים) של יכולת הכללה של המודל כאשר משחקים עם היחס של מספר הדוגמאות פר משימה ועם מספר המשימה. בגדול מאוד אם נותנים מספיק משימות גדול מספיק ומספר דוגמאות פר משימה גדול מספיק אז מגיעים להכללה אמיתית כאשר המודל אכן לומד את המשימה במלואה).

https://arxiv.org/abs/2406.02550
קריאה מהנה!

⚡🚀המאמר היומי של מייק 11.06.24:⚡🚀
The Geometry of Categorical and Hierarchical Concepts in Large Language Models

המאמר חוקר כיצד קונספטים ומושגים מקודדים במרחבי הייצוג (embeddings)של מודלים של שפה גדולה. הכותבים חוקרים 2 שאלות מרכזיות: הייצוג של קונספטים קטגוריים והקידוד של יחסים היררכיים בין קונספטים.

הם מרחיבים את ההסתכלות הלינארית הרגילה על הקונספטים כדי להראות שהקונספטים קטגוריים מיוצגים כסימפלקסים, קונספטים היררכיים הם אורתוגונליים, וקונספטים מורכבים מיוצגים כפוליטופים שנבנים מסכומים ישירים של סימפלקסים.

המחקר בוחן 957 קונספטים היררכיים עם נתונים מ- WordNet באמצעות מודל ג'מה. הכותבים מראים שקונספטים סמנטיים high-level יכולים להיות מנוטרים ומנוהלים על ידי מדידה ועריכה ישירה של הייצוגים הווקטוריים הפנימיים של ה-LLMs. התוצאות התיאורטיות מגלות מבנה פשוט שבו קונספטים קטגוריים מיוצגים גיאומטרית כסימפלקסים ומושגים היררכיים מקודדים כאורתוגונליות.

https://arxiv.org/pdf/2406.01506

⚡🚀המאמר היומי של מייק 12.06.24:⚡🚀
Accelerating Feedforward Computation via Parallel Nonlinear Equation Solving

היום סוקרים קצרות מאמר עתיק (מלפני 3 שנים) אבל יש למאמר הזה אימפקט גדול (רק תמשיכו לעקוב אחרי הסקירות היומיות). כשמסתכלים על שם המאמר הזה לא קל לקשר אותו ללמידה עמוקה. הרי מה לפתרון משוואות לא לינאריות וללמידה עמוקה? אולי מילה Parallel עשויה לרמוז לנו קלות על איזשהו קשר ללמידה עמוקה כי אנחנו מאוד אוהבים לחשב דברים במקביל במהלך אימון ואינפרנס של המודלים העמוקים שלנו.

אוקיי, זה כן קשור ותיכף נבין למה. קודם כל נרענן טיפה את זכרוננו על שיטות איטרטיביות לפתרון של מערכות משוואות כמו שיטת Jacobi או שיטת Gauss-Seidel(GS). שיטות אלו ניתן להפעיל גם במערכות משוואות לינאריות ולא לינאריות כאחד. בכל שיטה מתחילים מניחוש אקראי לפתרון ומעדכנים אותו על ידי חישוב איטרטיבי עד ההתכנסות (שצריך כמובן להגדיר) על יד עדכון וקטור הפתרון רכיב-רכיב. ד״א בשיטת יעקובי ניתן לעדכן את כל הרכיבים בצורה מקבילית ולעומת זאת GS פחות ניתן למקבול.

אבל איך כל זה קשור למודלים עמוקים? מתברר שתהליך האינפרנס במודלי שפה (נתמקד בהם למרות שהמאמר לא מגביל את עצמו אליהם אלא מדבר על מודלים אוטורגרסיביים כלליים) ניתן להציג על ידי מערכת משוואות כאשר כל משוואה בעצם ״בוחרת״ את הטוקן בעל נראות הגבוהה ביותר בהינתן הטוקנים הקודמים. כלומר כל משוואה מכילה פונקציית argmax על מרחב הטוקנים.

בד״כ האינפרנס מתבצע בצורה אוטורגרסיבית כלומר טוקן אחרי טוקן שזה כמובן מאט את מהירות האינפרנס. אנו מתחילים בסדרת טוקנים אקראית וממשיכים לעדכן אותה בצורה איטרטיבית עד ההתכנסות. מתברר שבאמצעות שילוב של שיטת יאקובי ו- GS ניתן לזרז את החיזוי.
https://www.arxiv.org/pdf/2002.03629

⚡🚀המאמר היומי של מייק 13.06.24:⚡🚀
Break the Sequential Dependency of LLM Inference Using LOOKAHEAD DECODING

זוכרים את המאמר שסקרנו קצרות אתמול שהציע גישה איטרטיבית לפתרון מקבילי של מערכות משוואות לא ליניאריות. אחת הדוגמאות של פתרון מערכות משוואות כאלו היא גנרוט טקסט ממודלי שפה כאשר כל טוקן נבחר בתור argmax של התפלגות הטוקן בהינתן הטוקנים הקודמים (המופק באמצעות השכבה האחרונה של מודל השפה).

יש בגדול שתי שיטות איטרטיביות שניתן לרתום אותן לדגימה יעילה יותר ממודלי שפה: יעקובי וגאוס-סיידל. שתי השיטות מתחילות מניחוש אקראי של כמה טוקנים בהינתן ההקשר ואז מאפטמים אותם על פתרון איטרטיבי של מערכת המשוואות עם argmax (ששקול לגנרוט). אפשר די בקלות לראות שבגלל שהמשוואות הן אוטורגרסיביות שיטות אלו לא יכולות להתכנס ביותר מ n איטרציות (מספר הטוקנים הנדגמים עם שיטה) ולפעמים אפשר להספיק פחות (נציין כי כל איטרציה דורשת קצת יותר משאבי החישוב).

הבעיה עם השימוש הנאיבי בשיטה הוא שהרווח הממוצע על פני דגימה אוטורגרסיבית סטנדרטית ממודלי שפה הוא לא גדול ועומד על פחות מ 1.1 האצת קצב גנרוט.

המאמר מציע שכלול לשיטה הנאיבית ומציע לשמור בזכרון את הטוקנים של כמה איטרציות האחרונות. במקרה אם והיא מוצאת בזכרון זה תת-סדרת טוקנים שבה הטוקן הראשון זהה לטוקן הראשון "הנכון" של האיטרציה(באיטרציה i טוקן i וקודמיו נחזים נכון) אנו לוקחים תת סדרה זו ומציבים אותו במקום מה שנחזה באיטרציה האחרונה.

זה מאפשר להקטין את כמות האיטרציות עוד טיפה
https://arxiv.org/pdf/2402.02057

⚡🚀המאמר היומי של מייק 14.06.24:🚀⚡
CLLMs: Consistency Large Language Models

בשתי הסקירות הקודמות(כדאי שתעברו עליהם כי נתתי שם קצת הסברים) דיברנו על שיטות איטרטיביות מקבילות לדגימה ממודלי שפה. השיטות האלו מבוססות על שיטות יאקובי או Gauss-Seidel (GS). השיטות האלו מתחילות מכמות מסוימת n של טוקנים שנדגמים באקראי (או בצורה קצת יותר מושכלת) ואז מעדכנים טוקנים אלו בבת אחת באיטרציות עד שתנאי עצירה מתקיים(התכנסות). תנאי העצירה כאן הוא בד"כ שוויון בין הפלטים של איטרציות עוקבות.

מובן שאנו מעוניינים לסיים את התהליך במשמעות פחות איטרציות ממספר הטוקנים שאנו חוזים בו זמנית (ד"א ניתן להראות נדרשות לכל היותר n איטרציות עד ההתכנסות).

שימו לב שמהלך האימון של מודלי שפה מותאם לשיטת הדגימה האוטו-רגרסיביות כאשר בוחרים טוקן בעל הסתסברות הגבוה ביותר ביהנתן הטוקנים הקודמים. אולם עכשיו אנו דוגמים בצורה אחרת ואולי ניתן להתחשב בזה במהלך האימון. כלומר במהלך האימון אשכרה דוגמים עם השיטה הזו (השילוב של יאקובי ו- GS).
וזה בדיוק מה שנסקור אותו היום עושה. המחברים מוסיפים עוד איבר ללוס הרגיל של מודלי שפה (הממקסם את הנראות המירבית של הדאטה). מטרת האיבר הזה היא לגרום למזעור של מספר האיטרציות עד להתכנסות של הדגימה האיטרטיבית.

המחברים בחנו שתי אופציות לאיבר הזה:

1. מזעור של מרחק (KL הפוך לדעתי אך לא צללתי לעומק) בין התפלגויות הטוקנים בנקודת ההתכנסות לבין התפלגויות טוקנים במהלך הדגימה האיטרטיבית (דוגמים האיטרציות באקראי).
2. מזעור מרחק בין התפלגויות הטוקנים באיטרציות עוקבות.

ואם חשבתם שיש דמיון בין השיטה הזו לבין המאמר של איליה סלוצקבר ושותפיו "Consistency Models" - אכן הוא קיים ואני אצלול בו בקרוב.

https://arxiv.org/abs/2403.00835

⚡🚀המאמר היומי של מייק 15.06.24:🚀⚡
MEDUSA: Simple LLM Inference Acceleration Framework with Multiple Decoding Heads

ב 3 הסקירות האחרונות ראינו כמה שיטות איטרטיביות מקבילות, מבוססות על שיטות יאקובי ו- Gauss-Seidel, המנסות להאיץ את מהירות גנרוט הטקסט (decoding) של מודלי שפה. היום נסקור קצרות מאמר המציע גישה אחרת לאותה הבעיה, שגם מבצעת גנרוט מקבילי של טקסט אבל בשיטה 'טיפה' אחרת.

בגדול המאמר מציע להוסיף ולאמן כמה "ראשים" (שכבה לינארית עם סופטמקס) למודל שפה מאומן. מטרתה של כל ראש כזה היא לחזות טקסט לא החל מהטוקן הבא אלא להתחיל לחזות מהטוקן ה-k אחרי הפרומפט (או הטוקן האחרון שנחזה). כלומר בהינתן פרומפט באורך 10 טוקנים הראש מסדר 3 מגנרט טוקנים החל מהטוקן ה-14 בזמן שמודל שפה רגיל חוזה(מגנרט) החל מהטוקן ה-11. הראשים האלו מחוברים לשכבה האחרונה (לפני שכבת החיזוי) של מודל שפה. כלומר הם מפעילים טרנספורמציה לינארית על ייצוג(תלוי קונטקסט) הטוקן המופק על ידי מודל שפה.

המחברים מציעים שתי דרכים לאמן מודל שפה עם הראשים האלו. הדרך הראשונה היא לאמן רק את הראשים כאשר מודל השפה עצמו נותר מוקפא. הדרך השנייה היא לעשות פיין טיון של מודל שפה מאומן (עם LoRa כמובן). במקרה השני הם משלבים את הלוס הסטנדרטי של מודלי שפה עם זה של הראשים האחרים.

באינפרנס המחברים לוקחים את החיזויים מהראשים השונים (כמה טוקנים החל מטוקן k לכל ראש) של הראשים השונים ומשלבים אותם בצורה דומה ל- beam search (כאן זה קצת יותר מורכב ונקרא tree-search) כדי לקבל את כמה סדרות של טוקנים (המועמדות) שמהן נבנה החיזוי הסופי של מודל שפה. כדי לבחור את התת-סדרות של טוקנים "הטובות ביותר" ביותר הם עושים משהו דומה למה שנעשה ב-speculative decoding קלאסי (טיפה יותר מורכב משם ו-rejection sampling בעניין).

אז מה הרווח כאן אתם שואלים? שהראשים מופעלים באופן מקבילי ולפעמים בהפעלה אחת שלהם אנו חוזים כמה טוקנים ולא אחד כמו בגנרוט אוטורגרסיבי רגיל.

https://arxiv.org/pdf/2401.10774

⚡🚀המאמר היומי של מייק 16.06.24:🚀⚡
STATISTICAL REJECTION SAMPLING IMPROVES PREFERENCE OPTIMIZATION

המאמר הזה וכמה הבאים שאסקור בימים הקרובים מציעים שכלולים שונים לשיטה Direct Preference Optimization או בקיצור DPO. למעשה DPO בעצמה היא שדרוג של Proximal Policy Optimization או PPO שהפכה להיות מאוד פופולרית אחרי שמכמה חברות השתמשו בה ליישור מודלי שפה (alignment או instruction tuning) בתור השלב האחרון של אימון מודל שפה foundational. השיטה שייכת למשפחת RLHF

כי היא דורשת דאטה (שאלות ותשובות) המדורגות על ידי בני אדם - עבור כל שאלה הם (המתייגים) בוחרים מה התשובה איזה תשובה טובה יותר.

למעשה DPO בא לייתר את מודל התגמול (reward) גם חוסך גם משאבים לאימונו וגם מאפשר לא להחזיק מודל נוסף בשלב RLHF. למעשה DPO מנצל את המבנה של פונקצית לוס של PPO, שהיא מקסום פונקציית תגמול עם איבר רגולריזציה שבא לשמור את המודל המיושר קרוב למודל התחלתי, כדי להיפטר מפונקציית התגמול בפונקציית לוס. זה מתאפשר עקב העובדה שקיים ביטוי מפורש לפוליסי האופטימלי (מודל שפה "מושלם אחרי היישור") דרך הפוליסי אחרי ה-SFT (מודל שפה שאנו מתחילים ממנו את אימון היישור) ופונקציית התגמול.

אחרי שמשתמשים במודל לוס המושרה על ידי מודל Bradley-Terry (BT) המגדיר מהי הסתברות העדפה של תשובה חיובית על תשובה שלילית (על אותה השאלה) מה- rewards שלהם, ואנו מגיעים לביטוי עבור לוס של RLHF שמכיל רק את הפוליסי התחלתי. זה למעשה DPO והוא ממזער את פונקציית הלוס שלו על סט המכיל זוגות של תשובות טובות וגרועות.

המאמר שנסקור היום שואל את השאלה האם הדגימה האחידה מהסט הזה היא אופטימלית (מבחינת איכות התוצאה שהיא הופליסי הסופי או מודל שפה אחרי היישור). אולי אם היה לנו פונקציית תגמול היינו מעדיפים זוגות עם יחס מקסימלי בין ה-reward של התשובה החיובית לשלולית? אולי צריך לתעדף זוגות עם reward שלילי הנמוך ביותר?

המאמר מציע את הגישה הבאה:
- מאמנים מודל text2text שבהינתן שאלה ושתי תשובות מוציא את התשובה המועדפת.
- בעזרת המודל הזה בונים את פונקציית התגמול דרך סמלוץ (על ידי דגימה של שאלה וזוג תשובות) של הסתברות העדפה של תשובה טובה על תשובה גרועה.
- בעזרת פונקציית תגמול זו בונים פוליסי pi_r שלמעשה זה מודל שפה (המאפשר לחשב הסתברות של תשובה בהינתן שאלה)
- משתמשים בדגימת rejection כדי לדגום pi_r באמצעות הפוליסי ההתחלתי (= מודל שפה) כדי למזער את הלוס בדרך לפוליסי "המיושר".

הם גם משחקים עם כמה פונקציות לוס כמו hinge loss (בטח כבר שכחתם אבל אוהבים להשתמש בו ב -SVM).

⚡🚀המאמר היומי של מייק 17.06.24:⚡🚀

SSAMBA: SELF-SUPERVISED AUDIO REPRESENTATION LEARNING WITH MAMBA STATE SPACE MODEL

הסקירה נמצאת כאן:
https://docs.google.com/document/d/1zmMPssJsuvb_3zyXZf4uoehuR5GCuWXhuzrhDiCt3UE/edit

⚡🚀המאמר היומי של מייק 18.06.24:⚡🚀

Helping or Herding? REWARD MODEL ENSEMBLES MITIGATE BUT DO NOT ELIMINATE REWARD HACKING

הסקירה הזו ממשיכה את קו הסקירות האחרונות שכתבתי בנושא RLHF. כמו שאתם זוכרים פונקציית לוס ב-RLHF מכילה שני איברים: האיבר שמנסה למקסם את פונקציית התגמול (reward) והאיבר השני מנסה לשמור את מודל השפה אחרי טיוב (פוליסי סופי) קרוב למודל שמתחילים את ה-RLHF ממנו. בעבר יצאו מאמרים

שהציעו לאמן כמה מודלי reward ואז למצע (או לקחת מקסימום) של כל ה-rewards של מודלים אלו עבור שאלה ותשובה נתונות של מודל שפה. זה לטענתם מקטין את הסיכוי שהמודל שפה ב-RLHF יבצע reward hacking כלומר יתכנס לפוליסי הממקסם תגמול אך בפעול מגנרט תשובות באיכות גרועה.

המאמר שנסקור היום טוען שגישה זו אינה אופטימלית כי פונקציית לוס שאיתה מאומנים מודלי reward (כלומר Bradley-Terry) גורמת לכך שכל שני מודלי reward שונים רק בקבוע שתלוי רק בשאילתה x יקבלו את אותו הערך של פונקציית לוס. בפועל זה אומר כי לכל לערכי ה-reward, המופקים על ידי המודלי, יכול להיות ממוצעים ובפועל הבחירה של המקסימלי או הממוצע מכמה מודלי כאלו עשויה להיות לא אופטימלית (כמו ממוצע של תפוזים ועגבניה). אז המאמר מציע לאמן פונקציית תגמול עם רגולריזציה שבאה "לרסן" את הקבוע זה שתלוי רק בשאילתה ובכך "לסכנרן" מודלי reward שונים.

⚡🚀המאמר היומי של מייק 19.06.24:🚀⚡

INTRINSIC DIMENSIONALITY EXPLAINS THE EFFECTIVENESS OF LANGUAGE MODEL FINE-TUNING

כולכם מכירים את LoRa נכון? בטח גם שמעתם על עשרות השכלולים השונים שלה כמו DoRa, MoRa, GaloRe וכדומה. מתברר כי היה מאמר שבצורה מסוימת הניח יסודות של משפחת הגישות הזו.

למעשה מה זה LoRa? זה אופן שבו אנחנו עושים פיינטיון של מודלים מאומנים גדולים למשימה ספציפית בלי לעדכן את כל משקלי המודל. במקרה של LoRa אנו מאמנים מטריצת תוספות למשקלים של כל שכבה כאשר תוספת זו היא בעלת ראנק נמוך הרבה יותר ממטריצת המשקלים המקורית. כלומר ניתן לייצג אותה על ידי מכפלה שתי מטריצות בעלות רנק נמוך (בגדלים מסוימים במקרה של LoRa).

מתברר שגישה זו היתה ידוע כבר ב 2020 ואפילו היו מאמרים שדיברו עליה ב 2018. אז המאמרים הציעו מספר דרכים לבניית מטריצת תוספת זו וביניהם הטלה ספארסית של וקטור במימד נמוך למרחב בעל מספר מימדים גבוה דרך Fastfood algorithm (צורה של מטריצת ההטלה הזו - תקראו עליו, זה חמוד).

בקיצור מאמר "היסטורי" מעניין וקל לקריאה.

https://arxiv.org/abs/2012.13255

⚡🚀המאמר היומי של מייק 20.06.24:🚀⚡

WARM: On the Benefits of Weight Averaged Reward Models

הסקירה הזו ממשיכה את קו הסקירות בנושא שיפור ביצועי RLHF לטיוב מודלי שפה. כבר דיברנו בסקירות הקודמות על כך שבמהלך RLHF המודל יכול לבצע reward hacking כלומר להתכנס לפוליסי (משקלי המודל) שממקסם את ה-reward ובאותו הזמן יוצר תשובות באיכות ירודה לפרומפטים.

המאמר שנסקור קצרות היום מציע לאמן כמה מודלי reward שונים ולהשתמש בממוצע שלהם כ-reward יותר "יציב" שעשוי למנוע מהמודל לעשות reward hacking. הבעיה העיקרית בגישה הזאת נובעת מכך שהיא מצריכה להחזיק בזמן אימון RLHF כמה מודלי reward שכמובן דורש יותר משאבי חישוב (ומייקר את חשבון החשמל).

המחברים מציע לשלב את התוצאה של המודלים אלא הביצועים שלהם. בשפה פשוטה הם מאמנים כמה מודלי reward וממצעים את המשקלים שלהם. זה מסתמך על איזושהי תופעה שלא ידעתי עליה שנקראת "Linear

mode connectivity״ או LMC הטוענת שהביצועים של מודל עם סכום ממושקל של המשקלים של כמה מודלים אחרים הוא יותר טוב מסכום ממושקל (עם אותם משקלים) של ביצועי המודלים (אולי אתעמק בזה בהמשך).

עכשיו כדי לבצע את הפעולה הזו הרשתות צריכות להיות בעלי אותה ארכיטקטורה ומה שונה בין מודלי reward כאן הם פרמטרי אימון כמו קצב למידה ודרופאאוט, סדר שונה של הכנסת דאטה לאימון (סיד שונה כנראה) וגם איתחולים שונים (לוקחים מודלים אחרי צ'קפוינטים שונים ב-SFT).

כתוצאה מקבלים מודל reward אחד טוב יותר שמשמש אותם לאימון RLHF.

⚡🚀המאמר היומי של מייק 21.06.24:⚡🚀
Named Entity Recognition as Structured Span Prediction

היום נסקור מאמר בנושא שלא סקרתי הרבה מאוד זמן והוא Named Entity Recognition או NER. מטרת משימה זו היא לזהות בטקסט עצמים(מילים וקבוצות מילים רצופות) מסוגים מסוימים כמו שמות פרטיים, כתובות מגורים, מספרי ת״ז וכדומה. קיימים מודלי NER המתמחים בזיהוי שמות חברות, רשומות רפואיות וכדומה.

מחד גיסא משימת NER היא משימה דיסקרימינטיבית והתוצאה שלה היא סיווג של כל טוקן במשפט לקטגוריה שהוא שייך או לקטגוריה "O" אם הוא לא שייך לאף קטגוריית יעד(נציין כי הזיהוי מתבצע פר מילה ולא פר טוקן מכיוון שמילה עשויה להיות מורכבת מכמה טוקנים). מאידך גיסא ניתן בקלות להפוך אותה לבעייה גנרטיבית כאשר המודל יגנרט את העצמים השייכים לקטגוריות יעד.

בעבר פותחו מגוון שיטות למשימה הזו, חלקם rule-based, חלקם סטטיסטיים אך לאחרונה רשתות השתלטו לנו על NLP וגם המשימה הזו לא הצליחה לברוח מהן. הוצאו לא מעט רשתות שהגיעו לביצועים די יפים במשימה הזו.

המאמר שנסקור היום מציע גישה מעניינת לבעיית NER. כמו שאמרתי ניתן לפתור את הבעיה הזו באופן דיסקרימינטיבי וגנרטיבי אך המאמר הזה לוקח גישה בין אלו וקורא לה Structured Span Prediction.

בגדול הגישה עובדת באופן הבא. מעבירים את כל שמות הקטגוריות יחד (מופרדים עם טוקן מיוחד) דרך טוקנייזר משלהם. לאחר מכן מעבירים את הטקסט דרך טוקנייזר משלו ומכניסים את שניהם דרך מודל שפה דו-כיווני (bidirectional או encoder) כמו BERT או DeBerta. המודל מפיק ייצוגי הטוקנים תלוי הקשר (גם עבור קטגוריות וגם עבור הטקסט) בתור פלט.

החידוש האמיתי בא לאחר מכן. הרי המטרה של NER היא לזהות כמה מילים רצופות השייכים לאותה קטגוריה. נגיד אנו לוקחים את המילים מ 1 עד 4 ומנסים לזהות מה הסיכוי שהם שייכים לקטגוריה c. המאמר מציע לקחת את הייצוגים תלויי הקשר של מילה 1, מילה 4 (המחברים מציעים להשתמש בייצוג של הטוקן הראשון של כל מילה לייצוג המילה) וגם קטגוריה c ובונים (מאמנים) מודל קטן לשערוך הסתברות זו. יש כמובן הרבה גישות לארכיטקטורה של מודל דליל זה. אפשר לעשות את עם רשת קונבולוציות פשוט, ניתן לקנקט את הייצוגים ולהוסיף שכבה לינארית ואני יכול לחשוב על כמה אופציות נוספות.

עכשיו השאלה האחרונה היא איך לבחור קטגוריות לכל המילים. השיטה הנאיבית היא לחשב את ההסתברויות האלו עבור כל תת סדרה של מילים רצופות החל מהמילה הראשונה ועבור כל תת סדרה לבחור את הקטגוריה בעלת הסתברות הגבוהה ביותר אם היא עולה על סף מסוים או קטגוריה ריקה אם זה לא. הבעיה עם הגישה הזו שכך נוכל לפספס spans ארוכים אחרי שסימנו את ה-span קצר יותר שיש לו חיתוך עם ה-span הארוך.

עקב כך המאמר מציע כמה גישות שונות לבעיה הלא פשוטה זו וביניהם Conditional Random FIelds וגם Maximum Weight Independent Set Method. גם Exhaustive Search יכול לעבוד עבור טקסטים קצרים. יש כאן טעימה נחמדה של שיטות מעניינות הלא קשורות לרשתות.

ואיך מאמנים את זה? האמת זה די פשוט - לוקחים את כל הקטגוריות המסומנות בטקסט ומריצים cross-entropy loss על כולם.

מאמר מאוד מעניין ומאיר עיניים - מחר ההמשך...

⚡🚀המאמר היומי של מייק 22.06.24:⚡🚀

GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer

המאמר הזה הוא שפצור קל של המאמר שסקרנו אתמול 21.06.24. המאמר מציע גישה לאימון והיסק של מודל לזיהוי NER המורכב משלבים הבאים:

1. מעברים כל קטגוריה שברצוננו לזהות דרך טוקנייזר - הקטגוריות מופרדות על ידי טוקן מיוחד הנקרא "ENT"
2. מעבירים דרך הטוקנייזר את כל הטוקנים של הטקסט. ד״א הטוקנים של הקטגוריות מופרדות מהטוקנים של טקסט על ידי טוקן מיוחד "SEP"
3. מכניסים את הטוקנים מהשלבים הקודמים לטרנספומר דו-כיווני (encoder) כמו BERT או ROBERTA
4. מעבירים את הייצוגים תלויי הקשר של הקטגוריות דרך FFN דו שכבתי (יש כזה בטרנספורמר) כדי לקבל ייצוג של כל קטגוריה.
5. מפעילים את מה שנקרא במאמר הקודם: Structured Span Prediction כלומר כדי לזהות את הקטגוריה של הטוקנים i עד i+n: לוקחים את הייצוג של טוקן ה-i ואת זה של טוקן i+n ומעבירים את השרשור שלהם דרך FFN דו שכבתי (מבנה דומה לסעיף הקודם) וכך מפיקים ייצוגו של ה-span הזה
6. כדי לשערך הסתברות ש- span (תת-סדרה של טוקנים רצופים) שייך לקטגוריה j מחשבים סיגמואיד של המכפלה פנימית של ייצוג הקטגוריה j מסעיף 4 עם ייצוג ה-span מהסעיף הקודם.
7. מפעילים אלגוריתמיסן גרידים כדי לזהות spans השייכים לכל קטגוריה (המאמר לא מרחיב על כך, צריך להביט בקוד)

⚡🚀המאמר היומי של מייק 23.06.24:⚡🚀

TextGrad: Automatic “Differentiation” via Text

אני קצת שיכור אחרי כמה שוטים ובירות באירוע המגניב של one-shot אבל התמדה בסקירות יומיות גברה על כך. הסקירה של היום מדברת גישה ש״מטילה״(project) את שיטת מורד גרדיאנט (gradient descent או פשוט GD) למקרה שהמשתנה שאנו מפטמים לפיו זה הפרומפט ולא משקלי המודל (שנותרות קבועים). כמו שאתם זוכרים GD הסטנדרטי מזיזים בכיוון הגרדיאנט השלילי של פונקציית לוס (מחסירים ממשקלי המודל את הגדיאנט מוכפל בקצב למידה).

ב-GD הגרדיאנט מחושב בצורה ברורה (לפחות מתמטית) כי פונקציית לוס הינה גזירה ביחס למשקלי המודל. ד״א בראייה ממוחשבת ניתן לגזור את פונקציית לוס לפי הקלט (תמונה) מאותה הסיבה - לפעמים עושים זאת כדי לבנות תמונה הממזערת את הלוס עבור קטגוריה מסוימת.

אבל איך לגזור את המודל ביחס לטקסט? הכוונה כאן לא לגזור את פונקציית לוס לפי הייצוגים של טוקני הקלט (זה דווקא אפשרי כמו במקרה של תמונה ונקרא soft prompting). אך כאן מדובר ב"גזירה" אשכרה לפי הטקסט עצמו. כמובן שמבחינה מתמטית זה די בעייתי כי טקסט הוא משתנה דיסקרטי.

המאמר הופך את ומחליף גזירה מתמטית על ידי מה "פידבק של שכבה ח לשכבה ח-1" וכאן לא מדובר בשכבות של מודלי שפה אלא בשכבות של כלים שונים המפעילים ומופעלים על ידי מודלי שפה (נגיד rag או כמה אג'נטים). אז בכל שלב אנו שואלים מודל שפה (אם הוא מופעל) איך היה ניתן לשפר את הפרומפט בשלב שלהם כדי לשפר את התוצאה ומעבירים את הפידבק לשכבה הקודמת. כמובן שהאגרגציה של פידבקים מתחילה ה-llm בשכבה האחרונה של המערכת ושיש לו סוג של פונקציית לוס בתור "שערוך של איכות התשובה". ומכאן מתחילה האגרגציה.

אז textgrad זה בגדול פרופגציה של פידבק טקסטואלי ופחות טקסט אבל עדיין המאמר חמוד כי מאפשר מערכות מורכבות מלא מעט כלים המערבים llms.

⚡🚀המאמר היומי של מייק 24.06.24:🚀⚡
Are you still on track!? Catching LLM Task Drift with Activations

הסקירה הזו הולכת להיות קצרה כי הרעיון העיקרי של המאמר הוא די פשוט ואינטואיטיבי. אתם מדברים עם מודלי שפה שלכם באמצעות שאילתות שבד"כ נקראות פרומפטים שהמודל עונה לכם. אבל מה קורה אם מודל השפה שלכם מחובר לעוד כלי שמגנרט בשבילו פרומפטים למשל בהתבסס על תוצאה של איזשהו חישוב על הפלט של מודל אחר או מתבסס על RAG או אולי אפילו תלוי בתוצאות חיפוש באינטרנט.

כמובן שגנרוט אוטומטי של פרומפט יכול להתפקשש (באגים, אולי פעילות זדונית) ואז יחד עם שאלה לגיטימית המודל מקבל תופסת לא קשורה. בעיה ידועה, אה?

אז המאמר שבנידון חקר את האקטיבציות של שכבות המודל (טרנספורמר כמובן) ומצאו הבדלים משמעותיים בין האקטיבציות הנוצרות על ידי שאלה לגיטימית לבין אלו שנוצרו עם שאלה "מקושקשת". ואז הם בנו דאטהסט של שאלות טובות ושאלות מורעלות ואימנו מודל (קטן) שיודע להבדיל בין האקטיבציות של שאלות הטובות והלא טובות. המחברים לוקחים אקטיבציות של הטוקן האחרון של הפרומפט (השאלה) המלא

הם ניסו שתי שיטות: אחת היא אימון של שכבה לינארית המפרידה בין ייצוגים טובים ומורעלים. השיטה השניה שהם מנסים נקראת metric learning שבמילים פשוטות מנסה ללמוד ייצוג (המופק על ידי המודל "המבדיל") המקרב ייצוגים של העוגן (התחלת השאלה) עם השאלה הטובה ומרחיק אותו מהייצוג של השאלה המורעלת (התוספת המורעלת). אם מצליחים ב-metric learning אז בקלות אפשר לתפור שכבה לינארית המבדילה בין הטובים ללא טובים.

https://arxiv.org/pdf/2406.00799

⚡🚀המאמר היומי של מייק 25.06.24:🚀⚡
Improving Reinforcement Learning from Human Feedback with Efficient Reward Model Ensemble

הסקירה הזו ממשיכה את קו הסקירות על המאמרים שמנסים לשפר שיטות RLHF לטיוב (instruction tuning או פשוט fine-tuning) של מודלי שפה. בחלק של שיטת RLHF (למשל PPO) אנו מאמנים מודל reward מבוסס על סט של שאלות ותשובות מדורגות על ידי המתייגים האנושיים. מטרה של מודל זה לספק ציון לזוג (שאלה,

תשובה) כאשר ציון גבוה מצביע על תשובה טובה ורצויה. לאחר כן אנו מאמנים (מטייבים) מודל שפה כאשר המטרה היא מקסום של פונקציה reward תוך שמירת של משקלי המודל למשקלים שהתחלנו מהם (נמדד על ידי KL divergence בין התפלגויות הטוקנים של שני המודלים). כל זה מתבצע on-the-fly כאשר הדוגמאות נוצרות עלי ידי הגרסה העדכנית של המודל במהלך האימון.

הבעיה עם הגישה היא reward hacking כאשר למרות איבר הרגולריזציה (KL) המודל מתכנס למשקלים שמגיעים לערכים גבוהים של פונקציית reward כאשר המודל עצמו "לא מספק את הסחורה". המאמר מציע להשתמש בכמה מודלי reward כי ensemble זה תמיד טוב. הבעיה שלהחזיק יותר ממודל אחד בזמן האימון זה יקר מבחינת המשאבים. המאמר מציע שתי גישות להתגבר על זה:

- מתחילים מאותו המודל (שפה)
- לאמן מודלי reward זהים עם ראשים לינאריים (מאומנים) שונים. כך צריך לשמור רק מודל אחד והמשקלים עבור השכבה הלינארית עבור כל מודל.
- לאמן כמה מודלי reward בשיטה של LoRa - כך נצטרך לשמור רק את תוספת המשקלים לכל שכבה שזה יכול להיות די זול מבחינת המשאבים

ואז אפשר לקחת ממוצע של ה-rewards של כל המודלים או את המינומום ביניהם- יש לא מעט אופציות...

⚡🚀המאמר היומי של מייק 27.06.24:🚀⚡

Probing the Decision Boundaries of In-context Learning in Large Language Models

המאמר הזה מדגים בפעם מי יודע מה שיש משימות שמודלי שפה מתקשים בהם מאוד אבל אם נסביר לו את המשימה ב"שפתו" הוא די מצליח להסתדר איתה. הפעם המשימה היא סיווג מולטיקלאס - כלומר אנו מספקים למודל כמה זוגות של וקטורי x והלייבל שלו y. הווקטורים ניתנים להפרדה בצורה לינארית על ידי ישר מסוים כלומר נמצאים בשני צידיו (של הישר). זה ה-context שלנו. לאחר מכן המודל מקבל נקודה x ומתבקש לחזות את הלייבל שלו y.

המחברים ניסו להבין עד כמה המודל מצליח לזהות את הלייבלים של הנקודות הנמצאות בין המינימום למקסימום של נקודות x שנתנו לא ב-context. בשביל כך הם חקרו את את התפלגות הטוקן הבא אחרי השאילתה עבור כל הקטגוריות עבור ערכי x שונים. באופן לא מפתיע המודל לא למד להפריד את הנקודות על ידי הישר אלא התכנס לקו הפרדה די פרוע ורחוק מהישר המפריד. הגדלת מספר הנקודות ב-context לא עזר ובנוסף קו ההפרדה היה רגיש לסדר הנקודות וגם לסימון של הלייבלים השונים.

אז המחברים גילו שפיינטיון של המודל על משימות דומות די עוזר למודל להתכנס לפתרון הנכון (גם לא מפתיע). ויש עוד כמה דרכים לגרום למודל לפתור את המשימה הפשוטה הזו.

אם אתם שואלים אותי למשימות כאלו יש לכם רגרסיה לוגיסטית....

⚡🚀המאמר היומי של מייק 28.06.24:🚀⚡

ON-POLICY DISTILLATION OF LANGUAGE MODELS: LEARNING FROM SELF-GENERATED MISTAKES

מזמן לא סקרתי מאמר על שיטות זיקוק של ידע(knowledge distillation) - לא נתקלתי במאמרים מגניבים בנושא המעניין הזה. מה זה זיקוק ידע ממודל גדול למודל קטן יותר? למעשה זה ניסיון להעתיק למודל הקטן את הידע שיש למודל הגדול כלומר לגרום לו להפגין ביצועים הדומים למודל הגדול.

יש כמה שיטות לעשות זאת - הפשוטה ביותר זה לאמן אותו על הדאטה שהמודל הגדול אומן עליה. יש שיטות המאמנות את המודל הקטן על הדאטה המיוצר על ידי המודל הגדול. אם יש לנו גישה להתפלגויות (של הטוקנים) אז מאמנים את המודל הקטן לחקות את התפלגות הטוקנים שהמודל הגדול מוציא. אם יש לנו אקטיבציות של השכבות של המודל הגדול ניתן לנסות לחקות גם אותם (אם המודל הקטן הוא בעל אותה ארכיטקטורה אבל עם פחות שכבות).

בכל גישות האלו אנו מאמנים (או פיינטיון) את המודל הקטן בצורה supervised רגילה. כלומר יש לנו סט של דוגמאות (ground-truth או שנוצרו על ידי המודל הגדול) אנו מאמנים את המודל הקטן עליהם. המאמר שנסקור היום מציעה להשתמש בגישה מעולמות למידה באמצעות חיזוקים (reinforcement learning) ממשפחת on-policy. זה אומר שהאימון מתבצע על הדוגמאות שהרשת המאומנת עצמה יוצרת במהלך האימון (והיא משתנה כמובן).

המאמר הלך צעד אחד קדימה והחליט לשלב את שיטת אימון on-policy יחד עם האימון הסטנדרטי של זיקוק ידע. כלומר בהסתברות alpha השיטה בוחרת דוגמא מדאטהסט האימון ובשאר המקרים היא מגרילה דאטה מהמודל הקטן. כל פעם המודל מנסה למזער את המרחק בין התפלגות הטוקנים של הדוגמא (מהדאטהסט או מהמודל הקטן).

בד״כ כלל המרחק בין התפלגויות של הטוקנים בשיטות זיקוק ידע נמדד על KL divergence סטנדרטי (כלומר forward). המאמר מציע לשכלל את הגישה הזו עקב חולשה שיש ל- forward KL. החולשה הזו קשורה לעובדה ש-forward KL מנסה לקרב את התפלגות המודל המאומן לאזור המוד(mode) של התפלגות היעד (התפלגות המודל הגדול במקרה שלנו. הכוונה כאן שהתפלגות המודל המאומן עלולה ״להתרכז באזור בעל מסה הסתברותיות גבוהה״, נגיד ליד איזה מוד של ההתפלגות ומתעלמת מאיזורים אחרים שיש בהם מסה הסתברותית ליד מודים חלשים יותר של ההתפלגות.

למזלנו יש לנו reverse KL שהופך את המונה ואת המכנה בלוג של forward KL. ניתן להראות כי forward KL מנסה ״לכסות״ את כל האזור בה התפלגות היעד גדולה מאפס ובכך משלימה את forward KL. ניתן לשלב אותם לינארית (באופן קמור עם מקדם beta ו- beta-1) ואז מקבל Jensen Shannon Convergence או JSD שנותן מענה לבעיה האינהרנטית של forward KL. ובה המאמר משתמש במקום forward KL הרגיל.

ניתן לשלב את פונקציית הלוס של המאמר עם עוד איבר האחראי על מקסום פונקציית reward כלשהי עבור המודל הקטן (כמו ב-RLHF).

ושכחתי להגיד(לא קשור למאמר) ש- forward KL זה בדיוק מה יש לנו בכל פונקציית לוס המבוססת על cross entropy (נגיד במשימות סיווג).

⚡🚀המאמר היומי של מייק 29.06.24:⚡🚀

What Are the Odds? Language Models Are Capable of Probabilistic Reasoning

הסקירה הזו הולכת להיות ממש קצרה. לפני ימיים (27.06) סקרתי מאמר שבדק האם מודלי שפה ענקיים מסוגלים לבצע רגרסיה לוגיסטית והגיע למסקנה שבלי עזרה ורמזים מאוד משמעתיים הם לא מצליחים לפתור אותה.

הפעם המחברים בדקו האם מודלי שפה מסוגלים ״לנתח התפלגויות הסתברותיות״. למשל אומרים למודל שפה שאיזשהו ערך מפולג גאוסית עם תוחלת 3 ושונות 2 ושואלים אותו מה האחוזון ה-80 של ההתפלגות או לאיזה

אחוזון שייכת דגימה בעלת ערך 4. באופן די מפתיע המודל מצליח לא רע בשאלות האלו למרות שקיבל הוראה לא להריץ קוד (זה יכול לעזור כמו שאתם מבינים).

אז מה לדעתכם קורה כאן? איך המודל מצליח לפתור את השאלות האלו?
https://arxiv.org/abs/2406.12830

⚡🚀המאמר היומי של מייק 01.07.24:🚀⚡
Grokfast: Accelerated Grokking by Amplifying Slow Gradients

המאמר הזה משך את עיניי משתי סיבות. הסיבה הראשונה היא הופעת מילי Grokking בכותרת. מה זה בעצם Grokking בהקשר של אימון רשתות. אתם בטח יודעים אם אנו מאמנים את הרשת שלנו ליותר מדי זמן (כלומר אפוקים) אז באיזושהי נקודה היא מגיעה למצב של אוורפיט. כלומר הלוס על טריין סט ממשיך לרדת בזמן שהלוס על סט ולידציה מתחיל לעלות כלומר יכולת הכללה של המודל נפגעת.

אבל אם אנו נמשיך לאמן את הרשת שלנו עוד עוד באיזשהו שלב הלוס על סט ולידציה מתחיל לרדת לאט לאט כלומר יכולת הכללה של המודל משתפרת. כלומר אנו יוצאים מ"משטר האוורפיט" אחרי שלב מסוים של אימון וזה נקרא grokking. התופעה הזו נחקרת רבות על ידי המדענים בתחום למידה עמוקה אבל אין הבנה מלאה למה זה קורה. השורשים של grokking הזו נמצאים כנראה בתופעה שנקראת double descent.

הסיבה השנייה שבחרתי לסקור את המאמר כי נוכחתה של התמרת פורייה שם אלא אחרי התעמקות קלה התברר שניתן היה להסתדר גם בלעדיו ולהסביר את המאמר בצורה פשוטה יותר בהרבה (מה שאני עושה בסקירה הזו).

גרוקינג זו תופעה מאוד נחמדה וכל אדם המאמן את המודלים שלו חפץ להגיע אליך אך הבעיה שצריך לאמן את הרשת למשך מאות אלפי ולפעמים יותר איפוקים וזה מאוד יקר. השאלה האם ניתן לזרז את התהליך הזה ולהגיע לגרוקינג מהר יותר.

וזה בדיוק מה זה המאמר רוצה לעשות. המאמר טוען שאם נחליק טיפה את עדכון המשקלים של הרשת (כלומר את הגרדיאנטים) אז ניתן להגיע לגרוקינג מהר יותר. נשמע לא מופרך בגדול (למשל PPO בלמידה עם חיזוקים גם מרככת את עדכון הגרדיאנט וגם שיטות אימון כמו ADAM ומומנטום של נסטרוב) - אבל כמובן ההוכחה לא נמצאת במאמר. וכאן המחברים דוחפים התמרת פורייה מהסיבה הפשוטה שהחלקה זו היא למעשה העברת גרדיאנים דרך מסנן low-pass אבל כאמור אפשר היה להסתדר בקלות בלעדיהם.

בסופה של דבר המאמר מציע למצע כמה גרדיאנטים, להחליק(להוסיף) באמצעות הממוצע הזה את הגרדיאנט הנוכחה ואז לעדכן את משקלי הרשת (עם adam למשל). כמובן שזה דורש לשמור כמה גרדיאנטים וזה מצריך הרבה זכרון והמחברים הציע החלקה מעריכית (exponential smoothing) במקום זה בלי כמעט לפגוע בתוצאות (התוצאה היא כמובן זירוז של הגעה לגרוקינג).

מאמר חמוד אבל ציפיתי ממנו קצת יותר..
https://arxiv.org/abs/2405.20233

⚡🚀המאמר היומי של מייק 02.07.24:🚀⚡
From Artificial Needles to Real Haystacks: Improving Retrieval Capabilities in LLMs by Finetuning on Synthetic Data

היום סוקרים מאמר קליל שלא דורש כל התעמקות מתמטית אבל עדיין יש בו רעיון נחמד. המאמר מציע גישה מאוד פשוטה לשיפור יכולת של מודל שפה להפיק מידע מטקסט בצורה מדויקת. למשל בהינתן טקסט ארוך המוזן למודל, המודל נדרש לענות נכון על שאלות עליו (הטקסט) בלי קשר לאיפה נמצא פיסת הטקסט הרלוונטית לשאלה. מודלי שפה בד"כ מתקשים במשימה זה בהעדר אימון ייעודי.

שיטת פיינטיון מקובלת לתת למודל טקסטים ארוכים ולאמן אותו לענות על מגוון שאלות בטקסט הזה (למשל לוקחים פסקה לא קשורה, משתילים אותה לטקסט ושואלים אתה המודל לגביה. גישה זו מביאה לשיפור בביצועי המודל במשימה אבל כמה מחקרים הצביעו על כך שבמהלכה המודל למד "מידע ועובדות מיותרים" שהרע את יכולת ה-reasoning שלו.

המחברים הציעו שיטה כדי להקל הבעיה זו. הם בנו דאטהסט שהוא הרבה מאוד מילונים שהמפתחות והערכים שבהם הם מספרים. המודל מאומן להפיק נכון ערך של מפתח נתון. משימה יותר קשה להפיק ערך של מפתח מסיום המורכב מכמה מספרים כאשר אני מעבירים את המספרים מהפתח למודל בסדר שונה מאשר הם מופיעים באחד המילונים. היופי כאן שהדאטהסט הזה לא מכיל מידע עובדתי בכלל והמודל לא יכול ללמוד אותו (המידע). ככה מונעים את "הרעלת המודל" במידע זר...

https://arxiv.org/abs/2406.19292

⚡🚀המאמר היומי של מייק 03.07.24:⚡🚀

The Remarkable Robustness of LLMs: Stages of Inference?

מאמר מעניין החוקר איזה שכבות ניתן לזרוק ממודל השפה ועדיין לשמור על ביצועים נאותים. אתם אולי מכירים lottery ticket hypothesis הטוען כי ברשתות עתירות פרמטרים (overparameterized) בד:כ ניתן למצוא קטנה הרבה יותר עם ביצועים מאוד קרובים אך הבעיה שאנו לא יודעים לאתר אותה.

המאמר כאמור בחן איזה שכבות הן סוג של מיותרות במודלי שפה והגיע לתופעות מעניינות לגבי תהליך האינפרנס שלהם. הם זיהו 4 שלבים עיקריים

1. דה-טוקניזציה או רכישה התחלתית של קשרים קונטקסטואליים: טרנספורמציה ראשונית של ייצוג ה-raw (מהמילון) של הטוקנים לייצוג תלוי הקשר (חישובי attention כבדים לכל אורך הקונטקסט).
2. הנדסת פיצ'רים התחלתיים מהייצוגים תלוי הקשר מהשלב הקודם ו"הכנת קרקע" לחיזוי של הטוקנים הבאים. עדיין לא ניתן לחזות את הטוקנים האלו מהפיצ'רים בשלב הזה אבל המודל מתחיל "להבין הקשרים מרחבים ועתיים בטקסט (היה מחקר מעניין הזה)
3. בניית קבוצות נוירונים (אנסמבל) לחיזוי הטוקן הבא. בשלב הזה הרשת מתחילה להתכנס ולבנות קבוצות "prediction neurons" שישולבו יחד למטרת חיזוי הטוקן הבא.
4. חידוד של prediction neurons: הרשת "בוחרת" את הנוירונים החשובים ביותר לחיזוי הטוקן הבא על ידי הדעכה של חלק מה-prediction neurons מהשלב הקודם.

והכי חשוב שהשכבות מעורבות בשלב 1 ובשלב 4 הם הכי חשובות לביצוע המודל כאשר חלק מהשכבות של שלב 2 ו-3 ניתן להסיר ללא פגיעה משמעותית בביצועים.

הרבה טענות מעניינות במאמר הזה (חלקם הגדול זה סיכום של העבודות הקודמות בנושא הזה).
https://arxiv.org/abs/2406.19384

⚡🚀המאמר היומי של מייק 04.07.24:⚡🚀

How Do Large Language Models Acquire Factual Knowledge During Pretraining?

המאמר חוקר נושא מתי מודלי שפה אשכרה רוכשים ידע עובדתי (למשל שעיר בירה של צרפת היא פריס) במהלך אימון מקדים. בנוסף המאמר גם בודק כמה זמן לוקח לשכוח ידע עובדתי. אוקיי, אתם בטח זוכרים שאנו מאמנים מודלי שפה שלנו עם אחת הצורות של משפחת מורד הגרדיאנט (gradient descent או GD). בד״כ דוגמים כמה דוגמאות הסט האימון שלנו (מיני-באץ') ומזיזים לינארית את משקלי המודל לכיוון הנגדי של הגרדיאנט הממוצע של מיני-באץ'.

המאמר בונה דוגמא של טקסט המכיל ידע עובדתי ומכניס אותו למיני-באץ' כל כמה איטרציות של GD. המחברים מצאו כמה דברים מעניינים. למשל כמות דאטה שהמודל אומן עליו לפני התחלת הזרקת ידע עובדתי לא משפיע על מספר האיטרציות הנדרש ללמידה של ידע עובדתי. כלומר יותר ״ידע״ הנמצא כבר במודל לתורם למהירות הלמידה.

שנית, המאמר מראה שמהירות הלמידה של ידע עובדתי לא מושפעת ממתי מתחילים להזריק למודל את הידע. כלומר מודל מאומן לאו דווקא תלמיד יותר טוב. ויש עוד כמה תגליות מעניינות במאמר.

איך בודקים האם המודל אכן למד את הידע העובדתי שהזרקנו - המחברים לא מרחיבים על כך אבל כנראה זה מחושב דרך likelihood של התשובה הנכונה על השאלה לגבי פיסת ידע עובדתי זה, למשל ״מה עיר הבירה של צרפת״.

https://arxiv.org/abs/2406.11813

⚡🚀המאמר היומי של מייק 05.07.24:🚀⚡
A Survey of Large Language Models for Graphs

גרפים מודלי שפה גדולים: האם זה שידוך מהחלומות? גרפים נמצאים בכל מקום, מרשתות חברתיות ועד למבנים מולקולריים ורשתות נוירונים על גרפים (GNNs) הם הפתרון הנפוץ למשימות כמו ניבוי קישורים וסיווג קודקודים. אבל ל-GNNs יש מגבלות: הם מתקשים עם נתונים דלילים ולעיתים קרובות אינם מצליחים להכליל היטב לגרפים בעל מבנה שלא נראו קודם.

מאידך גיסא LLMs מספקים פתרון משלים: הם מצטיינים בהבנה וסיכום טקסטים (שזה דאטה דליל שהוא בעצם גרף - המתאר קשרים בין מילים או קבוצות של מילים) יותר מאשר גרפים. אז, מה אם נשלב את החוזקות של GNNs ו-LLMs? מאמר סקר חדש חוקר לעומק את החיבור המבטיח הזה.

המחברים מציעים טקסונומיה של ארבעה שילובים אפשריים בין LLM ל-GNNs: שימוש ב-GNNs בתור שלב מקדים ל-LLMs, שימוש ב-LLMs לפני GNNs, שילוב של LLMs וגרפים, ושימוש ב-LLMs בלבד למשימות גרפיות. לכל גישה יש יתרונות וחסרונות, אבל הפוטנציאל ברור. על ידי ניצול הכוח של LLMs, נוכל להתגבר על חלק מהמגבלות של טכניקות למידה מסורתיות על גרפים.

https://arxiv.org/pdf/2405.08011

⚡🚀המאמר היומי של מייק 07.07.24:🚀⚡
The Road Less Scheduled

היום סוקרים מאמר שלא נראה כמו מאמר למידה עמוקה רגיל. בהתחלה זה אולי יכול להיראות שהמאמר מציע עוד שכלול מי יודע מה ל-ADAM או שיטה אופטימיזציה של לוס אחרת. אבל זה לא בדיוק. המאמר כן מציע שיטת אופטימיזציה (מציאת מינימום) לפונקציות קמורות אבל זה בא ממטרה לשפר את Adam או משהו כזה אלא מציע שיטה לשיפור קצב ההתכנסות של אלגוריתם מורד הגרדיאנט (GD) הידוע.

המאמר מתחיל מכך שמבחינה תיאורטית האלגורית של Polyak-Ruppert (PR) הוא זה שאמור להביא התכנסות אופטימלי אבל בפרקטיקה זה פחות קורה (לא ברור לאיזה פרקטיקה הם מתכוונים כי התוצאות שהם נתנו מתייחסות לרשות עמוקות הלא קמורות). PR בעצם עושה אותו GD אבל העדכון האמיתי המוחלק מעריכית עם העדכון האחרון. כלומר באיטרציה t העדכון של GD נכנס עם המקדם t/1 (אפשר לשחק עם זה לפי המאמר אבל קשה להגיע לקצב החלקה אופטימלי).

המאמר מציע שיטה חדשה (3 שלבים במקום 2 ב-PR) שמשפרת ההתכנסות של PR ללא צורך בבחירה של פרמטר ההחלקה.

⚡🚀המאמר היומי של מייק 08.07.24: ⚡🚀
Mixture of A Million Experts

המאמר של היום מציע לקחת את שיטת Mixture of Experts(MoE) לבניית ארכיטקטורות של מודלים עמוקים פופולרית במיוחד במודלי שפה. מאוד בגדול ב- MoE הרשת מורכבת מתת-רשתות (בד״כ מחלקים את שכבת ה-FFN של הטרנספורמר לכמה חלקים זרים). MoE מאומן להשתמש כל בפעם בחלק מתת-רשתות אלו (הנקראות מומחים) כאשר רשת gating רדודה יחסית באיזה מומחים צריך להשתמש כל פעם. כלומר יש לנו כן סוג של מימוש הגישה שנקראת ״lottery ticket hypothesis" דינמי כאשר כל פעם בוחרים להריץ רק חלק מהרשת.

כנראה שככל יש ברשת יותר מומחים בעלי אותה הארכיטקטורה וכל פעם בוחרים אותו מספר של המומחים הביצועים אמורים להשתפר אולם המחיר הוא המודל גדול יותר.המאמר מנסה לבדוק האם שווה להשתמש בהרבה מאוד במומחים רזים מאוד. המחרים מציעים לעבוד עם מיליון של מומחים של כל אחד מהם היא דל במיוחד. כמובן שכל פעם צריך לבחון את המומחים כל פעם ומכיוון שיש מיליון מומחים אז נדרש מאמץ חישובי לא קטן. המאמר מציע להשתמש בטכניקה הנקראת product key retrieval כדי להקטין את הסיבוכיות (בגדול זה חלוקה של וקטור המפתחות (keys) לשני חלקים, ביצוע חישוב לכל אחד בנפרד ושילובם).

וגיליתי משהו מעניין במאמר הזה - יש scaling law גם ל-MoEs. אולי אסקור אותו בקוב...

⚡🚀המאמר היומי של מייק 09.07.24: ⚡🚀
Learning to (Learn at Test Time): RNNs with Expressive Hidden States

המאמר הזה המצהיר שהוא לומד ב״זמן טסט״ משך את עיניי היום. המאמר מציע ארכיטקטורה חדשה ומעניינת לעיבוד דאטה סדרתי. בעיקרון הרשת די דומה ל-RNN מבחינת המהות אבל יש כמה הבדלים מהותיים.

אז מה יש לנו בארכיטקטורה הזו? בדומה ל-RNN אנו מחשבים את הייצוג עבור יחידת דאטה בזמן t (נגיד טוקן t) אבל כאן עושים זאת בשיטה שונה. לפי המאמר במקום לחשב את הייצוג עצמו אנו מחשבים את וקטור המשקלים שיאפשר לנו לחשב את ייצוגו של יחידת דאטה t. כלומר אנו מעדכנים את משקלות מודל בתנועה בהתאם לדאטה כלומר הרשת מתאפטמת ומתאימה את עצמה לדאטה שעליה היא מופעלת. זה נעשה באמצעות הזזה של המקשלים בכיוון הנגדי של הגרדיאנט של פונקציית לוס l.

מה זה בעצם פונקציית l ואיך מאמנים אותה? נניח שהייצוג של איבר דאטה t מחושב על ידי פונקציית f. במקרה הזה פונקציית l יכולה להיות (למשל) נורמה של הפרש ריבוע של ייצוג דאטה z (המחושב עם f) מהדאטה עצמו. כלומר אנו מאמנים את וקטור הייצוג להיות מסוגל לשחזר (כלומר לזכור) את הדאטה עצמו x_t. כמובן שאין בזה

הרבה משמעות אבל אם נאמן רשת עם קלט מורעש ונשווה את ייצוג עם הדאטה האמיתי נקבל סוג של רשת denoising שהרשת לומדת להפיק ייצוג המאפשר לזכור את הפיצ'רים המהותיים של דאטה הנחוצים לשחזור.

דרך אחרת המוצעת במאמר לאמן את רשת לשחזר הטלה למימד נמוך של דאטה להטלה אחרת כאשר שתי ההטלות נלמדות גם כן. הייצוג של דאטה במקרה הזה מחושב עם הטלה נלמדת שלישית (עם פונקציית f). כלומר המטרה כאן ללמוד את ייצוג של דאטה כאשר המשקלים מחושבים עם GD מהמשקלים הקודמים.

הארכיטקטורה קיבלה שם ttt וניתן לשלב אותם על שכבות אחרות (כמו טרנספורמרים או SSM). רעיון מגניב שבינתיים לא הפנמתי אותו עד הסוף...

https://arxiv.org/pdf/2407.04620

⚡🚀המאמר היומי של מייק 11.07.24: ⚡🚀

DOLA: DECODING BY CONTRASTING LAYERS IMPROVES FACTUALITY IN LARGE LANGUAGE MODELS

המאמר שנסקור היום הולך להיות די קליל. הוא מתמקד בהקטנת הזיות (hallucinations) של מודלי שפה. מה זה הזיה של מודל שפה? זו שאלה לא טריוויאלית בכלל (יש כמה תרחישים). נתמקד בהזיה המתבטאת בכך שהמודל נותן תשובה לא נכונה עובדתית. נגיד, כלומר על השאלה מה עיר בירה של לטביה הוא עונה שזה ריגה בזמן שהתשובה הנכונה היא טאלין.

המחברים מציע שיטה ה"מכיילת" את התפלגות הטוקנים בשכבת החיזוי (האחרונה) של מודל שפה. המאמר טוען כי בהרבה מקרים שבהם הטוקנים הנכונים בתשובה מפגינים עליה משמעותית בהסתברות מהשכבות הראשונות ועד האחרונות. זה בולט במיוחד בטוקנים הלא טריוויאלים (לא מילות חיבור וכאלו) הדורשים ממודל שפה לגייס את הידע העובדתי שלו. בהתאם לאובזקבציה זו המאמר מציע שיטה המורכבת משני שלבים. בשלב הראשון מזהים את השכבה הרחוקה ביותר מבחינת התפלגות הטוקנים (השכבה הזו נקראת השכבה הכי פחות בשלה) מהשכבה האחרונה. מרחק כאן מוגדר על ידי Jensen-Shannon divergence או JSD בין התפלגויות הטוקן.

בשלב השני מחסירים (ב-log scale) את ההסתברויות של השכבה הכי פחות בשלה מההסתברויות של השכבה האחרונה. בנוסף מאפסים את כל לוגיטים של הטוקנים בעלי הסתברות הקטנות ביותר (שממילא לא אומורים להיבחר). לאחר מכן עושים סופטמקס ומשתמשים בשיטת decoding האהובה עליהם כדי לחזות את הטוקן הבא.
https://arxiv.org/abs/2309.03883

⚡🚀המאמר היומי של מייק 12.07.24: ⚡🚀

To Believe or Not to Believe Your LLM

מאמר מאוד מעניין מבית גוגל. המאמר מנסה להבין איך ניתן לזהות עד כמה המודל בטוח בתשובתו לשאלה. כלומר המאמר עוסק בכימות של אי ודאות של תשובות המודל. המאמר מנסה בין שני סוגים של אי-וודאות הידועים בתורת השערוך: אלטורי (aleatoric) אפיסטמי (epistemic). אי-הוודאות האפיסטמית מתרחשת כאשר המודל לא יודע מה התשובה לשאלה ומתחיל לאלתר (כלומר להוציא הזיות או hallucinations). לעומת זאת אי הוודאות אלאטורית מתרחשת כאשר יש כמה תשובות לשאלה נתונה והמודל בוחר אחת התשובות הנכונות.

המאמר מציע שיטת פרומפטינג המאפשרת להבדיל בין שני סוגי אי-וודאות. מאוד בגדול לשאלה נתונה מזינים למודל תשובות אחרות (לאו דווקא) נכונות לשאלה (other response is...). לאחר מכן בודקים האם ההסתברות

של התשובה הנכונה מושפעת מכמות התשובות האחרות המוזנות למודל. אם הסתברות זו מתחילה לרדת זה הסימן שמודל שפה לא כזה "יודע מה התשובה" ואי הוודאות האפיסטמית הינה גבוהה.

המאמר גם מציע פריימוורק מתמטי המבוסס על כלים מתורת המידע לאנליזה של אי-הוודאות האלו. נשמע מאמר שווה להתעמק בו.

https://arxiv.org/pdf/2406.02543

⚡🚀המאמר היומי של מייק 13.07.24: 🚀⚡

SaySelf: Teaching LLMs to Express Confidence with Self-Reflective Rationales

בהמשך לסקירה של אתמול, מאמר קליל יותר שמציע שיטה ללמד מודלי שפה לשערך אי וודאות בתשובתם. המחברים מציעים שיטה מאוד אינטואיטיבית המורכבת משני שלבים עיקריים: יצירת דאטהסט למשימה זו (כימות אי וודאות) וטיוב (fine-tuning) של המודל על הדאטהסט הזה. בשלב השני ממשיכים לאמן את המודל עם שיטת PPO מעולם למידה באמצעות חיזוקים כדי לשיפור נוסף של ביצועיו.

בשלב הראשון לוקחים דאטהסט של שאלות ותשובות הנקרא HotpotQA ומזינים את השאלות ממנו למודל שפה ומבקשים ממנו לתת תשובה מלווה ב-reasoning. לאחר מכן מקלסטרים את תשובות המודל (יחד עם ה-reasoning) לקלסטרים לפי האמבדינג שלהם ומחשבים את יחס של גודל הקלסטר המכיל את התשובה הנכונה (מהדאטהסט) יחסית לכל התשובות. זה מדד אי הוודאות שלנו שעליו נאמן את המודל בהמשך.

לאחר מכן מפלטרים את השאלות ובסוף מבקשים מ-gpt4 לתת הסברים למה המודל היה עשוי לתת תשובות לא נכונות לשאלה (כלומר "הסיבה" לאי וודאות). בשלב האחרון מטייבים (מאמנים מודל שפה נתון) קודים כל לתת תשובה נכון, לדייק בממד של אי הוודאות ובנוסף לתת reasoning נכון לנוכחות של אי הוודאות. כל אלה נמצאים באופו מפורש בפונקציית הלוס.
בשלב השני ממשיכים לאמן את המודל בשיטה PPO כדי למזער (או למקסם אותה עם מינוס) את ההפרש בין נכונות התשובה (0 או 1) ורמת ה-confidence של המודל לגביה. כמו בכל שיטת PPO הדוגמאות נוצרות "on the fly" אחרי כל עדכון של משקלי המודל.

https://arxiv.org/abs/2405.20974

⚡🚀המאמר היומי של מייק 15.07.24: 🚀⚡

Lookback Lens: Detecting and Mitigating Contextual Hallucinations in Large Language Models Using Only Attention Maps

אהבתי את המאמר הזה כי הרעיון מאחריו הוא מאוד אינטואיטיבי ופשוט. המאמר מציע גישה להתמודדות עם הזיות(hallucinations) של מודלי שפה. מאוד בגדול הזיות של מודל שפה קורות כאשר מודל שפה עונה לא נכון לשאלת המשתמש. יש לכך כמה סיבות: למשל המודל לא מסוגל לענות על התשובה כי היא פשוט לא נמצאת ב"זכרון שלו" (למשל השאלה על אירוע עדכני שהמודל לא אומן על הדאטה לגביו). הסיבה השניה היא העדר יכולות להבין את השאלה.

המחברים מנסים להתמודד עם הזיות של מודל שפה על ידי ניתוח של משקלי ה-attention של הפרומפט (השאלה) ושל תשובתו של המודל. נניח שהפרומפט מכיל N טוקנים וכרגע אנו חוזים טוקן מספר t של תשובתו של מודל שפה. קודם כל מחשבים את סכום מקדמי ה-attention עבור N טוקנים של הפרומפט P וסכום מקדמי ה-attention עבור כל t הטוקנים של התשובה R. מחשבים את היחס בין P ל- P + R ועבור כל שכבה של הטרנספורמר ועבור כל ראש (head) של בלוק הטרנספורמרים.

לאחר מכן בונים וקטור מהיחסים האלו ומאמנים מודל המכיל שכבה אחת שמטרתו היא לחזות האם המודל הוזה או לא. כיוון האורך תשובתו של המודל יכול להיות כלשהו המחברים מאמנים מודל עבור מספר קבוע של טוקני התשובה T. אם התשובה מכילה יותר מ- T וטוקנים מפעילים את המודל עבור כמה פעמים בשביל לזהות הזיות בחלקים השונים של התשובה.

איך בונים דאטהסט לאימון של המסווג הזה? בגדול נותנים למודל שפה לענות על שאלה ומפעילים מודל שפה חזק כדי לזהות תשובות נכונות ולא נכונות (הזיות).

⚡🚀המאמר היומי של מייק 16.07.24: ⚡🚀
How Does Quantization Affect Multilingual LLMs?

היום נסקור קצרות מאמר שחוקר נושא חשוב לכל מי שעוסק במודלי שפה. הנושא הזה הוא קוונטיזציה או קוואינטוט של מודלי שפה שמאפשר לנו גם להקטין את כמות הזכרון הנדרש לאחסון של המודל וגם מזרז את האינפרנס של המודל. אבל כמובן שזה לא בא בלי המחיר והמחיר הוא ביצועיי המודל. המאמר חוקר עד כמה חמורה פגיעה בביצועי המודלי לכמה רמות ושיטות קוואינטוט(ניתן לקוונטט שכבות שונות ברמות שונות וגם ניתן לקוונטט משקלי המודל והאקטיבציות ברמות שונות של קוואינטוט).

המאמר נכתב על ידי מדעני חברת cohere ובאופן טבעי מתמקד במודלי שלהם. המחברים לקחו מודלים בגדלים שונים ובדקו אותם במספר בנצ'מארקים שונים וגם ביצעו אבלואציה של ביצועי המודלים על ידי בודקים אנושיים. המחברים הגיעו למספר מסקנות מעניינות:

1. הפגיעה מהקוואינטוט הנמדדת על הבנצ'מארקים משמעותית קטנה יותר מזו הנעשית על ידי בודקים אנושיים.
2. הפגיעה לרוב מחמירה ככל שקוואינטוט נהיה יותר קשוח כלומר לפחות ביטים
3. מודלים גדולים בד״כ עמידים יותר לקוואינטוט מאשר מודלים קטנים יותר
4. מודלים מולטי-שפתיים (multilingual) סובלים יותר מקוואינטוט מאשר מודלים חד שפתיים והביצועים על השפות הפחות נפוצות נפגעות יותר מאשר על שפות נפוצות יותר
5. היכולת של המודלי ל-reasoning (למשל יכולת לפתור שאלות מתמטיות) נפגעת מאוד מהקוויטוט.

יש עוד כמה מציאות מעניינות...

https://arxiv.org/pdf/2407.03211

⚡🚀המאמר היומי של מייק 17.07.24: ⚡🚀
Learning Rate Curriculum

רוב המאמרים שסקרתי לאחרונה היו בנושא מודלי שפה והחלטתי לגוון טיפה ולסקור מאמרים בנושאים אחרים. מאמר שנסקור היום מדבר על שיטת אימון הנקראת למידת curriculum שבא אנו מאמנים את המודל כמו שאנו מלמדים חומר לתלמידים - מהקל לקשה. יש כמה וריאציות של למידת curriculum: באחת מהם אנו מתחילים לאמן מודל עם דוגמאות קלות ובהדרגה מעלים את קושי הדוגמאות. הוריאציה השניה אנו מתחילים ממשימה קלה יותר ומעלים את מורכבותה בהדרגה. בשלישית מאמנים מודל יחסית פשוט ומעלים את ״מורכבות״ של המודל.

המאמר מציע גישת curriculum אבל לקצב למידה. המחברים מציינים שלמשל ברשתות קונבולוציה עדיף בהתחלה להתמקד יותר בלמידה של השכבות הראשונות כי למעשה אם אלו לא נלמדו טוב ועדיין קרובים למצב האיתחול שלהם אז הם יוצרים דאטה ״רועש, מדי שזורם גם לשכבות הבאות שמתקשות להתמודד איתו (המאמר

מציין כמה עבודות שחקרו את הנושא והגיעו למסקנות האלו). תופעה דומה מתרחשת גם כאשר אנו עושים פיין טיון למודל למשימה מסוימת כאשר המודל לפני זה אומן למשימה אחרת.

כדי להתמודד עם סוגיה זו המחברים מציעים להתחיל מקצב למידה גבוה עבור השכבות הראשונות (שיורד ככל שמתקדמים לשכבות עמוקות יותר). במהלך האיטרציות לעלות את קצב למידה בכל השכבות כך (קצב עלייה לא שווה בין השכבות) כך שעם הזמן (=איטרציות) קצבי הלמידה של כל השכבות משתוות. נציין שהמחברים מציעים שמספר האיטרציות הנדרש להשוואת קצב הלמידה עבור כל השכבות צריך להיות משמעותית קטן יותר מכמות האיטרציות הכולל הנדרש לאימון המודל. כלומר כל השיטה הזו מופעלת בשלב ה״חימום״ של הרשת.
https://arxiv.org/abs/2205.09180

⚡🚀המאמר היומי של מייק 18.07.24: 🚀⚡
Trainable Highly-expressive Activation Functions

ממשיכים את קו הגיוון וסוקרים מאמר לא קשור ישירות למודלי שפה. היום נסקור מאמר של כמה חוקרים ישראלים המציע דרך חדשה לבנות פונקציות אקטיבציה ברשת נוירונים. היום פונקציות אקטיבציה הן לא נלמדות לרוב (ReLU, GeLU, tanh וכדומה). לפעמים פונקציות אקטיבציה מכילות hyperparameter שלא נלמד במהלך האימון אלא נקבע מראש (Leaky ReLU, Swish וכדומה).

המאמר מציע פונקציות אקטיבציה שהן(הפרמטרים שלהן) אשכרה נלמדות במהלך האימון. ד״א לאחרונה ראינו דוגמא נוספת לפונקצית אקטיבציה נלמדת ראינו לא מזמן במאמר המפורסם Kolmogorov-Arnold network או KAN - שם אלו היו ספליינים נלמדים. במאמר המסוקר אימצו שיטה אחרת לבנייה של פונקציות אקטיבציה נלמדות. הבנייה נעשה דרך שדות וקטורים שמגדירות את המסלול של נקודה במרחב.

במקרה הזה אנו מתחילים מנקודה x ובעזרת נגזרת של כיוון תנועת הנקודה(=שדה וקטורי) ב״זמן״ (שמתחיל ב t=0 ומסתיים ב- t=1) נבנה המסלול של נקודה x. המסלול מסתיים ב t=1 לכל x שלמעשה מגדיר לנו פונקציית אקטיבציה a(x). ניתן לתאר את התקדמות נקודה באמצעות משוואה אינטגרלית (כמו שיטת אוילר לפתרון משוואות דיפרנציאליות).

המאמר מתבונן במקרה של שדה וקטורי נתון על ידי פונקציה רציפה המורכבת מפונקציות אפיניות (לינארית מוזזת) באינטרוול נתון. פונקציית זו מכיל פרמטרים נלמדים המגדירים את הפונקציות האפיניות. ניתן להראות כי פונקציות אקטיבציה היוצאות מהתהליך הזה הם diffeomorphism, כלומר פונקציה גזירה בעלת פונקציה הופכית גזירה גם כן. פונקציות כאלו נקראות CPAB. דרך אגב פונקציות אלו שימשו בעבר לטרנספורמציות ״לוקאליות״ של דאטה בסדרות זמן או של תמונות (למשל ל-time warping דינמי של סדרות זמן).

המאמר מציע לשכלל את פונקציית אקטיבציה שתיארנו קודם ומגדירים אותה לכל x ולא באינטרוול נתון. הם מגדירים באינטרוול ״הרגיל״ פונקציית אקטיבציה שהרחבנו עליהם לפני תוכפל ב-GeLU (שזה התפלגות קומולטיבית של גאוסיאן המוכפל ב- x) ובשאר האינטרוול תהיה שווה ל-x. יש גם עוד גרסה שבה במקום x פונקציית אקטיבציה שווה ל LReLU מעבר לאינטרוול שלה.

בנוסף יש איבר רגולריזציה על הפרמטרים של CPAB של פונקציית האקטיבציה המוצעת. כדי לזרז את החישובים (הרי כל פעם צריך לפתור משוואה אינטגרלית לכל אקטיבציה) המחברים מציעים לבצע קווינטוט ולחשב את ערך הפונקציה רק ב-n נקודות באינטרוול ה-CPAB שלה.

מאמר כיפי וכתוב היטב - נהניתי לקרוא.

⚡🚀המאמר היומי של מייק 19.07.24: ⚡🚀
DataDream: Few-shot Guided Dataset Generation

מזמן לא סקרתי מאמר בנושא של מודלי דיפוזיה גנרטיביים - הנושא האהוב עליי לפני שנה - שנתיים. המאמרים בנושא הזה השתנו מאז ובד״כ לוקח לי קצת זמן לצלול לעומק. המאמר הזה היווה יוצא מן הכלל והיה די קל עקב האינטואיטיביות שלו ובנוסף שימוש בטכניקות דומות בתחום מודלי שפה.

המאמר מציע שיטה מעניינת לבניית מסווג לבעיות למידת few-shot דרך יצירה של דאטה סינטטי (מכאן בא הרעיון העיקרי של המאמר). כלומר יש לנו מודל דיפוזיה מאומן, כמה תמונות בודדת מכמה קטגוריות והמטרה שלנו לבנות מסווג לתמונות מקטגוריות אלו. כאשר יש לנו מעט תמונות פר קטגוריה וגם הקטגוריות עצמם הן לא טריויאלית ושכיחות אז המשימה הזו עלולה להיות לא פשוטה בכלל.

כאמור המאמר מציע לגנרט דאטה סינטטי ולאמן עליו את המסווג. הרעיון הוא ליצור דאטה סינטטי באמצעות מודל דיפוזיה מאומן שעובר פיין טיון על התמונות המעטות מהקטגוריות שיש לנו ביד. ואז אנו מאמנים את המסווג על התמונות האלו. הבעיה עם הגישה הזו היא שהתפלגות התמונות המגונרטות לא תמיד קרובה להתפלגות האמיתית של הקטגוריות עצמן ואז המסווג המאומן עליהן לא מפגין ביצועים גבוהים.

המאמר מציע גישה נחמדה להתגבר (או לפחות להקל) על הסוגיה הזו. המאמר מציע לבצע שני סוגים של פיין טיון של מודל דיפוזיה מאומן (שיודע ליצור תמונה מטקסט) על התמונות שיש לנו ביד. הפיין טיון הראשון הוא פר קטגוריה כלומר המודל לומד ליצור תמונה פר קטגוריה (שיוצר N מודלים כאשר N זה מספר הקטגוריות) והשני D_all לומד ליצור תמונה מהדאטהסט (לא מקטגוריה ספציפית).

הפיינטיונים מתבצעים בצורה של LoRA כלומר לומדים מטריצת תוספות בעלות רנק נמוך למטריצות key, value, query ומטריצות W_o (המשלבת את הפלט של כל ראשי הטרנספורמרים שיש לנו במודל דיפוזיה גנרטיבי). לאחר מכן יוצרים דאטהסט סינטטי גדול באמצעות N+1 המודלים שאומנו (המאמר לא מפרט איך מסווגים קטגוריות של התמונות המיוצרות על ידי D_all האומן על כל הקטגוריות).

בשלב האחרון לוקחים את מודל CLIP (מודל פופולרי של openai לפני chatgpt) ועושים פיין טיון באמצעות אותה LoRA לאנקודר של תמונות ולאנקודר של טקסט שלו על הדאטהסט המכיל את התמונות האמיתיות והתמונות המגונרטות. המטרה היא לקרב את הייצוגים של התמונות ושל הקטגוריות שלהן בהתאם לדאטה המתיוג.

מאמר נחמד וקל לקריאה.

https://arxiv.org/pdf/2407.10910

⚡🚀המאמר היומי של מייק 20.07.24: ⚡🚀
Consistency Models

המאמר הזה חיכה את תורו די הרבה זמן, קצת פחות משנה וחצי המאמר הזה נכתב על ידי Yang Song האגדי (זה שכתב מאמרים חזקים מאוד בתחום הדיפוזיה) ואחד המחברים הוא איליה סלוצקבר שאני מניח שאתם מכירים היטב. המאמר נכתב עוד בתקופה ששני המדענים הדגולים אלו עבדו ב-openai. דרך אגב שני המחברים אחרים גם תרמו לא מעט לתחום המודלים הגנרטיבים ושניהם עבדו ב-openai לפחות נכון למרץ 2023.

המאמר מציג גישה חדשה לאימון מודלי דיפוזיה גנרטיביים. מודל דיפוזיה גנרטיבי סטנדרטי מורכב מתהליך קדמי ומתהליך האחורי (forward & backward). בתהליך הקדמי אנו מוסיפים רעש (בבד"כ גאוסי) לדאטה באופן הדרגתי עד שפיסת דאטה הופכת להיות רעש. בתהליך האחורי אנו מאמנים את המודל להסיר רעש בצורה הדרגתית גם כן. כלומר המודל לומד מה הרעש צריך להחסיר מהדאטה המורעש באיטרציה t כדי לקבל את הדאטה באיטרציה t-1. אחרי שהמודל מאומן לעשות זאת אנו יכולים להשתמש בו ולבנות פיסת דאטה מרעש טהור על ידי הסרה של רעש בצורה הדרגתית.

מה הבעיה בתהליך הזה? הוא עלול להיות די ארוך (צריך להריץ מודל כמספר האיטרציות) ויצאו לא מעט מחקרים שניסו להקטין את מספר האיטרציות בלי לפגוע באיכות הדאטה המגונרט. מודלים קונסיסטנטיים(consistency models) זה עוד ניסיון לתקוף את הבעיה המעניינת הזו. בגדול הרעיון כאן הוא שעבור פיסת דאטה נתונה x_0 שלא משנה מאיזו איטרציה t (=דאטה מורעש x_t) נתחיל את הסרת הרעש בסופו של דבר אנו חייבים לחזור לדאטה הנקי x_0.

המאמר מציע שתי שיטות לאמן מודל דיפוזיה קונסיסטנטי. השיטה הראשונה מניחה שיש לנו ביד מודל דיפוזיה מאומן (consistency distillation) והשני מאמן את המודל מאפס(consistency training). כדי להסביר את השיטה הראשונה צריך טיפה לצלול למתמטיקה אבל נעשה את זה לאט ובזהירות.

נתחיל מזה התהליך הקדמי המודל דיפוזיה מתואר על ידי משוואה דיפרנציאלית סטוכסטית המתארת את היצירה הדרגתית של הדאטה המורעש. ניתן להראות שמשוואה דיפרנציאלית רגילה (ODE) ל x_t. מעניין ש-ODE הזה מכיל לוגריתם של פונקציית ההסתברות של ההדאטה המורעש x_t (נקרא score function או SF). המשוואה מתארת את בתהליך האחורי (הסרה הדרגתית של רעש). אז אם יש בידינו שערוך של SF אנו נוכל לשחזר (הדרגתית) את הדאטה שלנו על ידי הפתרון הנומרי (באיטרציות) של ה-ODE הזה (נגיד Euler-Maruyama).

הדבר הכי מגניב שאם יש לנו מודל דיפוזיה מאומן (שערוך של הריש באיטרציה t) אז ניתן בקלות לקבל שערוך של SF (בתנאי של רעש גאוסי).

אבל איך כל זה קשור למודלים קונסיסטנטיים שצריכים להיכנס לאותה הנקודה לא משנה מאיזו איטרציה של הרעשה מתחילים. אנו מאמנים את המודלי באופן הבא: לוקחים נקודה מורעשת t, עושים איטרציה אחת של הפתרון הנומרי של ODE (עם SF) כדי לקבל את הדאטה באיטרציה t-1. תזכרו שהמטרה שלנו היא לאמן את המודל לשחזר את הדאטה הנקי מכל איטרציה של הרעשה. אז מאמנים מודל למזער את ההפרש בין הדאטה המשוחזר מאיטרציה t לזה של האיטרציה t-1. בגדול יש כאן שני מודלים (בדומה לשיטה של למידת הייצוג הנקראת BUOL). המודל הראשון המודל המוחלק שהפרמטרים שלו הם ממוצע מעריכי של המשקלים של המודלים מהאיטרציות אימון הקודמות(לא מאומן - יש stop gradient) והוא נקרא target והמודל השני שהוא למעשה מאומן עם מורד הגרדיאנט.

ניתן גם לאמן מודל ללא מודל דיפוזיה מאומן ובמקרה הזה יוצרים את x_t-1 מ- x_t על ידי הורדת הרעש. ברגע שאימנו מודלי קונסיסטנטי ניתן ליצור דאטה נקי מרעש טהור באיטרציה אחת אך זה לא תמיד אופטימלי. ניתן לבצע מה שנקרא במאמר Multistep Consistency Sampling. להתחיל מרעש טהור, ליצור דאטה נקי, להוסיף רעש, שוב ליצור דאטה נקי ולחזור עד שאיכות הדאטה הוא לשביעת רצוננו. המאמר טוען שנדרש משמעותית פחות איטרציות בתהליך זה מאשר במודלי דיפוזיה סטנדרטיים.

סיימנו, מקווה שלא איבדתי אותכם כאן...

⚡🚀המאמר היומי של מייק 22.07.24: ⚡🚀
TRAINING DIFFUSION MODELS WITH REINFORCEMENT LEARNING

אוקיי, בסקירה הקודמת סקרתי מאמר בנושא מודלי דיפוזיה גנרטיביים וקיבלתי תיאבון בלסקור עוד כמה כאלו. אז בחרתי במאמר המגניב הזה שאחד ממחבריו הוא סרגיי לווין האגדי (בנוסף למאמרים הרבים יש לו קורס די מטורף מבחינת העומק בנושא deep reinforcement learning). באופן לא מפתיע המאמר שנסקור קשור ללמידה עם חיזוקים (או RL בקצרה) אבל יחד עם זאת מופיע בשמו גם מודלי דיפוזיה.

לדעתי בעבר כבר סקרתי אחד המאמרים שלו המשלב גישות מעולם ה-RL לאימון מודלי דיפוזיה. מתברר שניתן לאפיין אימון במודל דיפוזיה עם כלים מעולם ה-RL כלומר ניתן לבנות תהליך החלטה מרקובי (MDP) מאד אינטואיטיבי עבור מודל דיפוזיה.

כמו שאתם זוכרים אימון של מודל דיפוזיה מסתכם בניית מודל שמשערך את הרעש שהתווסף לדאטה באיטרציה t של התהליך הקדמי (של ההרעשה ההדרגתית של דאטה). אם יש לנו את האומדן של הרעש שהתווסף לפיסת דאטה באיטרציה t אנו יכולים לאמוד את הדאטה המורעש באיטרציה הקודמת t-1. כלומר אנו מאמנים מודל denoising לבנייה של דאטה מרעש טהור.

המאמר למעשה מצא פריימוורק מעולם RL (כלומר MDP) למידול של אימון מודל דיפוזיה גנרטיבי. בשביל כך נגדיר את כל הפרמטרים של ה- MDP באיטרציה t באופן פורמלי:

- המצב (state): השלישיה {פרומפר, מספר איטרציה t, הדאטה המורעש x_t}
- הפעולה (action) היא x_t-1
- הפוליסי היא הסתברות לקבל x_t-1 מ- x_t ומהפרומפט c
- המצב ההתחלתי מוגדר על ידי השלישיה: {רעש גאוסי סטנדרטי (ממנו מתחילים denoising), הסתברות על מרחב הפרומפטים, האיטרציה האחרונה T}
- פונקציית תגמול (reward) שהמאמר מגדיר בכמה צורות. היא מחושבת באיטרציה האחרונה (על התמונה המשוחזרת).

עכשיו אחרי שיש לנו הגדרת RL של אימון מודלי דיפוזיה אנו יכולים להשתמש בשיטות RL קלאסית כמו REINFORCE או PPO למקסום של פונקציית התגמול.

לגבי פונקציית התגמול המאמר מציע כמה אופציות. האופציה הראשונה היא לחשב את מה שנקרא BERT Score שבודק כמה התמונה מתאימה לפרומפט שלה (שווים את האמבדינגס שלהם). האופציה השניה היא להשתמש במה שנקרא LAION aesthetics predictor שאומן לשערך עד כמה התמונה היא אסתטית (זו למעשה שכבה לינארית על האמבדינג של CLIP המאומן עד דאטהסט של תמונות המתויגות על ידי בני אדם.

מאמר מעניין ויחסית לא קשה לקריאה.

https://arxiv.org/pdf/2305.13301

⚡🚀המאמר היומי של מייק 23.07.24: ⚡🚀
Feedback Efficient Online Fine-Tuning of Diffusion Models

ממשיכים את הקו של אתמול וסוקרים עוד מאמר המשלב מודלי דיפוזיה עם טכניקות מעולם של למידה עם חיזוקים (RL). הפעם המאמר משלב את שני התחומים המרתקים האלו כדי לבצע פיין טיון של מודל דיפוזיה. המאמר מתמקד במקרה שאין בידינו דאטהסט (לפיין טיון) אלא יש לנו דרך לשערך (סוג של reward) את איכות

של פיסת דאטה מג'ונרט, כלומר סוג של משוב על איכות הדאטה. למשל אם מטרתנו היא לאמן מודל לגנרט מולקולות המשוב יכול להיות "מידת פעילות ביולוגית" (bioactivity) של המולקולה הנוצרת.

בגדול מאוד המאמר מציע לאמן מודל דיפוזיה מאומן (pretrained) למקסום של פונקציית התגמול (=המשוב) תוך כדי שמירת של התפלגות הדאטה המגונרט על ידי המודל קרוב יחסית לזו של המודל ההתחלתי. מזכיר לכם PPO ו-TRPO מעולם ה-RL - אז זה בערך אותו הרעיון עם קצת סיבוכים. התהליך הוא איטרטיבי וכל איטרציה אנו מעדכנים את פרמטרי המודל (כאן זה רק המשקלים - יוסבר בהמשך) ויוצרים דאטה חדש עם המודל המעודכן.

למעשה התהליך מורכב מ 3 שלבים עיקריים.

בשלב הראשון בונים דאטהסט חדש עם מודל דיפוזיה מהאיטרציה הקודמת (בהתחלה מתחילים ממודל מאומן (pretrained). כמו שכתבתי ניתן לתאר מודל דיפוזיה מאומן על ידי משוואה דיפרנציאלית סטוכסטית עם אופיינים נלמדים (פונקציה נלמדת למעשה עם שיטות כמו score matching או flow matching). למעשה ה-SDE הזה מתאר את תהליך יצירת דאטה מרעש טהור. אז בשלב הראשון מתחילים מרעש מחדש ופותרים את ה-SDE (עם פונקציה נלמדת התלויה בדאטה מורעש באיטרציה t וב- t עצמו). משתמשים בשיטות סטנדרטיות כמו אוילר או אוילר מוריאמה.

בשלב השני בהתבסס על הדאטה שיצרנו בשלב הקודם מאמנים מודל תגמול reward עם רגולריזציה (נגיד L1 או כל פונקציה התלויה במאפייני ה-reward ובמשימה עצמה). בנוסף מאמנים מודל המשערך אי וודאות של פונקציית תגמול. בגדול במקרה הזה המטרה של הפונקציה היא שערוך של סוג של רווח סמך של הפרש של פונקציית התגמול אופטימלית עם רגולריזציה ופונקצית תגמול עצמה על הדאטהסט מהאיטרציה הקודמת(הפרטים קצת מורכבים והעדפתי לא לצלול בהם בסקירה).

בשלב השלישי של כל איטרציה מאמנים פונקציה חדשה f עבור ה-SDE שלנו וגם ההתפלגות ההתחלתית v של שממנה אנו מייצרים את הדאטה באמצעות ה-SDE. יש שם נוסחאות די מורכבות אך אנסה להסביר את ההיגיון מאחוריהם בכל זאת. פונקציית המטרה כאן מורכבת מ 3 איברים (ממקסמים אותה על הדאטהסט משלב 1). המקסום מתבצע ביחס לפונקציית f וגם על ההתפלגות ההתחלתית ממנה יוצרים את הדאטה באמצעות SDE:

1. התגמול האופטימיסטי (סכום של פונקציית התגמול ומודל אי הוודאות משלב 2).
2. איבר רגולריזציה השומר את פונקציית f הנלמדת (מה-SDE) באיטרציה הנוכחית (של האלגוריתם ולא של מודל דיפוזיה) קרובה מבחינת מרחק KL לפונקציית f מה-SDE של המודל התחלתי. בנוסף רוצים לשמור את התפלגות הדאטה באיטרציה ההתחלתית הנלמדת v קרובה להתפלגות הדאטה ההתחלתית של המודל שהתחלנו ממנו מבחינת KL. שני הקירובים הלא צריכים להתקיים מעל כל האיטרציות של מודל דיפוזיה (פתרון של ה-SDE).
3. אותם איברי הרגולריזציה עבור f ועבור v שלא "מאפשר" להם לסטות יותר מדי מה- f ומה-v מהאיטרציה הקודמת של האלגוריתם עבור כל האיטרציות של מודל דיפוזיה.

מאמר קצת מורכב מתמטית - מקווה שעזרתי לכם קצת להבין אותו.

⚡🚀המאמר היומי של מייק 24.07.24: 🚀⚡

The Empirical Impact of Neural Parameter Symmetries, or Lack Thereof

הסקירה היום תהיה קצרה וקלילה לעומת הסקירות האחרונות על מודלי דיפוזיה למיניהם. המאמר של היום חוקר סימטריות ברשתות נוירונים עמוקות. ניתן לראות די בקלות כי קיימות לא מעט פרמוטציות של המטרצות המשקלים בשכבות השונות של רשת שלמעשה לא משנות את המודל. כלומר אם תפעילו את המודל אחרי פרמוטציה על כל קלט תקבלו את אותה התוצאה כמו עם המודל המקורי.

האם הסימטריות האלו מביאות לנו משהו טוב? בכלל לא בטול - לי זה נראה (למרות שאני לא מומחה גדול בתחום) כמו סוג של יתירות של יש במודלים שבלעדיה אולי ניתן היה להגיע למודלים קטנים יותר למשל. המאמר בוחן מה קורה במודל עם אנו מפרים את הסימטריה שיש במודל. אחת הדרכים להרוס את הסימטריה היא לקבע משקלות (לערכים אקראיים אך קבועים) במקומות שנבחרו באקראי במטריצות משקלים של הרשת. הדרך השניה היא להפעיל פונקציה אקטיבציה רק על המשקלים מסוימים.

המאמר חוקר איזה אפקטים מתרחשים אחרי שהורסים את הסימטריה במודל ומגלה כמה דברים די מעניינים....
https://arxiv.org/pdf/2405.20231

⚡🚀המאמר היומי של מייק 25.07.24: ⚡🚀
AI models collapse when trained on recursively generated data

מאמר די חמוד שחוקר מה קורה שמאמנים מודלי AI על הדאטה הנוצר על ידי מודלי AI. בשתי מילים - לא הכל ורוד שם ויש כמה סיבות למה הדברים עלולים להשתבש:

1. דאטה דריפט (איך זה בעברית?) קיצוני: אימון מודלים על דאטה שנוצרה על ידי מודלים אחרים גורם להתרחקות של התפלגות הדאטה הנוצר על ידי המודל החדש מהדאטה האמיתי (כלומר אגרגציה של מרחק בין ההתפלגויות שלהן)..
2. הבעיות מחמירות בזנבות התפלגות הדאטה (תחומים או שפות עם מעט דאטה למשל): ההידרדרות משפיעה בעיקר על זנבות התפלגות הדאטה, שם דאטה נדיר הופך להיות עוד פחות מיוצג
3. עוד יותר שגיאות: שגיאות בדאטה שנוצרו על ידי מודלים מצטברות לאורך דורות, מה שמוביל לירידה משמעותית בביצועים.
4. קריסת השונות: דאטה שנוצר על ידי מודלים חסרים את המגוון והעושר של הדאטה מהעולם האמיתי, מה שמוביל ליותר הומוגניזציית יתר (פחות גיוון).

https://www.nature.com/articles/s41586-024-07566-y

⚡🚀המאמר היומי של מייק 26.07.24: ⚡🚀
Questionable practices in machine learning

הסקירה היום תהיה ממש קצרה. המאמר המסוקר דן בפרקטיקות פסולות שעלולות להכשיל אתכם במהלך פיתוח של המודלים שלכם. רוב הפרקטיקות הרעות שנזכרו במאמר נראות לחוקרי ML מנוסים די טריוויאליות ודי ברור למה לא כדאי להשתמש בהן. בין אלו ניתן למנות אימון על טסט סט, בחירה של בייסליין חלש להשוואה, הסקת מסקנות על אימון אחד בלבד של המודל, אימון על דאטה דומה מאוד לבנצ'מארק וכדומה. אבל ניתן למצוא גם דברים פחות טריוויאליים שחלקם לא ידעתי.

https://www.arxiv.org/abs/2407.12220

⚡🚀המאמר היומי של מייק 27.07.24: ⚡🚀

Data Mixture Inference: What do BPE Tokenizers Reveal about their Training Data?

אחרי שבוע שלא סקרתי עבודות על LLMs חוזר לנושא הזה היום עם סקירה של המאמר המציע התקפה מציע תקיפה על מודלי שפה מבוססת טוקנייזרים. ההתקפה מיועדת לגלות מה המשקל היחסי של דאטה מסוג מסוים (שפה, שפת תכנות וכדומה) בדאטהסט שעליו אומן מודל שפה. לא יודע עד כמה ההתקפה הזו חמורה אבל עושים זאת על סמך הטוקנייזר.

אם אתם זוכרים הטוקנייזרים נבנים על שילוב אותיות (מספרים, סימני פסוק הכדומה) הכי נפוצים בדאטהסט האימון. אם הדאטהסט מורכב מכמה שפות אז הטוקנים שייבחרו יכילו גם אותיות (ולפעמים מילים שלמות) מכמה שפות המופיעות בדאטהסט. בשיטת טוקניזציה מפורסמת הנקראת Byte Pair Encoding או BPE קודם כל מפצלים את הטקסט לבטים (bytes), מחפשים זוגות בתים הכי נפוץ בדאטהסט, מאחדים אותם לטוקן חדש וממשיכים את התהליך עד שמגיעים לגודל של מילון הטוקן (50k-100k היום במודלי שפה מודרניים).

אז המאמר מנצל את מבנה של אחגוריתם טוקניזציה כדי להציע אלגוריתם המבוסס על התכנות הלינארי למציאת אומדן למשקל יחסי של הדאטהסטים השונים בסט האימון של המודל.

⚡🚀המאמר היומי של מייק 29.07.24: ⚡🚀
Large Scale Dataset Distillation with Domain Shift

המאמר מציע שיטה מעניינת ודי מקורית לגנרוט דאטה מהתפלגות הנתונה על ידי דאטהסט מתויג. למשל בהינתן דאטהסט של תמונות D_s המטרה היא ליצור דאטהסט (מתויג) גדול בעל התפלגות ה"מושרה" על ידי D_s. המחברים טוענים כי השיטות הקיימות מתקשות לבנות(distill) דאטהסט גדול המשקף בצורה נאמנה את המאפיינים המהותיים של D_s.
המחברים מציעים לגשת לבעיה זו עם גישה מעולם של domain adaption או DA בקצרה. בגדול מאוד DA היא תהליך של "התאמת מודל" במקרים בהם התפלגות הדאטה בזמן האינפרנס שונה מזו של הדאטה שעליה אומן המודל. התחום הזה עשיר בשיטות שחלקן די מורכבות מתמטיות ומערבות לרוב מינימיזציה של מרחק בין התפלגויות הדאטה (KL וכאלה).

למעשה המאמר המסוקר מתרגם את בעיית יצירת הדאטה לבעיית DA. התפלגות הדאטהסט שאנו מגנרטים "ממנו" D_s משחק תפקיד של התפלגות המקור במקרה של DA (שעליו מאומן המודל ב-DA) ואילו התפלגות הדאטה המגונרט משחקת תפקיד של התפלגות היעד D_t (כלומר זו של הדאטה שעליו מפעילים את המודל ב-DA). המטרה כאן לאמן מודל המקרב את ההתפלגויות האל.

אבל איך נחשב את ההתפלגויות האלו? המאמר מייצג את ההתפלגויות האלו על ידי התפלגות של האקטיבציות של השכבות השונות של הרשת. בפשטות עבור הדאטסט D_s אנו מייצגים את התפלגות הדאטה על ידי וקטור הממוצעים ומטריצת קוואריאנס של כל השכבות של המודל M_s(מניחים שהם גאוסיים). בדיוק באותו האופן אנו מייצגים את ההתפלגות של הדאטה המגונרט.

אבל מה כאן M_s ומה עושים כדי לקרב את התפלגות של הדאטה המגונרט להתפלגות הדאטה האמיתי? המודל M_s אומן לשערך את ההתפלגות של הדאטהסט המתויג D_s (המאמר לא מפרט איך M_s מאומן בדיוק). למעשה האופטימיזציה מתבצעת על **הדאטה המגונרט כאשר המודל M_s נותר ללא שינוי**. כלומר מתחילים מתמונות הנדגמות באקראי עם הלייבלים והמטרה היא לבצע מורד הגרדיאנט(gradient descent) על התמונות האלו במטרה לקרב אותם להתפלגות של D_s.

עכשיו נשאלת השאלה מפונקציית הלוס כאן. כאמור בשלב הראשון אנו מאפטמים את התמונות המגונרטות במטרה למזער מרחק KL בין התפלגויות המשקלי המודל M_s(נותר ללא שינוי) של D_s (נותר קבוע לכל אורך הדרך) ולבין התפלגות של משקלי המודל M_s עבור D_t. המחברים מניחים ששתי התפלגויות אלו הם גאוסיים שעבורם מרחק KL ניתן לחישוב באופן מדויק בהינתם וקטורי תוחלות ומטריצות קוריאנס של D_s ו- D_t עם M_s. איבר נוסף בלוס מנסה למקסם (=למזער עם סימן מינוס) הוא ההתפלגות המותנית של לייבל y בהינתן פיסת דאטה מג'ונרט (הרי אנו מגנרטים דאטה מתיוג). התיוג של כל פיסת דאטה מגונרטת נקבע מראש ולא משתנה במהלך האימון.

השלב השני הוא מזעור של מרחק KL בין ההתפלגות המותנית של הלייבלים של הדאטה המגונרט לבין זה של הדאטה מ-D_s. בשביל כך מנצלים את הדאטה המגונרט מהשלב הראשון. מחשבים את התפלגות הלייבלים עבור הדאטה המגונרט הזה עם מודל M_s ומאפטמים את הדאטה המגונרט במטרה לקרב את שתי ההתפלגויות האלו של הלייבלים (של הדאטה המגונרט ושל הדאטה מ-D_s).

יש עוד לא מעט פרטים מעניינים על איך בדיוק מתבצע האימון (משתמשים בלא מעט מודלים לחישוב סטטיסטיקות המשקלים, עושים מיצוע מעריכים לסטטיסטיקות של הבאצ'ים וכדומה). המאמר לא כתוב מאוד ברור אבל הרעיון יפה.

⚡🚀המאמר היומי של מייק 30.07.24: ⚡🚀
Denoising Vision Transformers

מזמן לא סקרנו מאמר בראייה הממוחשבת והיום נתרענן עם סקירה של מאמר די מעניין מהדומיין הזה. המאמר מציע שכלול ל-Vision Transformer או ViT בקצרה. משפחת ViT כוללת מודלים מבוססי טרנספורמרים המיועדים לעיבוד דאטה ויזואלי ולהפקת ייצוג חזק של תמונה. מה אני מתכוון כאשר אני אומר ייצוג חזק של תמונה? למעשה זה ייצוג (לטנטי) של תמונה, בעל מימד משמעותית נמוך יותר מהתמונה עצמה בד״כ, שניתן לנצלו לאימון מודלים למגוון משימות downstream (כגון סגמנטציה, זיהוי אובייקטים, סיווג וכדומה).

המאמר טוען שניתן לשפר אתת את הייצוגים המופקים על ידי ViT באמצעות ניקוי רעשים הנוצרים בגלל השימוש ב-positional encoding או קידוד תלוי מיקום. מטרתו של קידוד תלוי מיקום היא להעביר למודל מידע על מיקום של הפאצ'ים של התמונה. אזכיר כדי להזין תמונה ל-ViT אנו מפרקים אותה לפאצ'ים, משטחים אותם ומזינים אותם למודל. לוקטור המייצג כל פאץ' אנו מוסיפים (אשכרה מחברים) וקטור המקודד את מיקומו היחסי בתמונה של הפאץ'.

המאמר טוען שהוקטורים המקודדים מיקום מרעישים את ייצוגי הפאצ'ים ומקשים על שימושם למשימות downstream. לטענת המחברים רעש המתווסף לייצוגי הפאצ'ים מכיל מידע על המיקום של הפאצ' בלבד ולא מכיל שום מידע על התוכן של הפאץ'. לעומתו שני החלקים האחרים בייצוג הפאץ' מכילים מידע על התוכן הסמנטי של הפאץ' והשני מכיל מידע המערבב את ייצוג התוכן וייצוג המיקום. המחברים טוענים שניקוי הייצוג מהרעש המידע על המיקום בלבד תורם לעוצמתו של הייצוג.

כדי לאתר את הארטיפקט המיקומי הזה בייצוג הפאץ' המאמר מציע לאמן מודל המזהה את שלושת החלקים של הייצוג שהזכרנו בפסקה הקודמת. זה נעשה עלי די אוגמנטציה של תמונה (הזזה, קרופ וכדומה) דרך ניצול התכונות האינהרנטיות של הרעש המיקומי ושל הייצוג התוכן. כלומר המידע המיקומי בייצוג ״זזה יחד עם הפאץ״ כאשר המידע המייצג את התוכן לא משתנה אם מזיזים את הפאץ' בתמונה. החלק שמערבב את המידע על המיקום והתוכן היא פשוט הפרש בין ייצוג של ViT לבין סכום של שני החלקים האחרים.

בשלב השני מאמנים מודל המזהה את הרעש המיקומי בייצוג הפאץ'. לאחר מכן באינפרנס מחסירים את הרעש הזה מהייצוג של הפאץ' וכדי לקבל ייצוג יותר נקי ועוצמתי.

⚡🚀המאמר היומי של מייק 31.07.24: ⚡🚀
DENOISING DIFFUSION IMPLICIT MODELS

זה מאמר לא חדש (אוקטובר 2022) אך חשוב מאוד בתחום של מודלי דיפוזיה. מאמר עם רעיון מאוד אלגנטי המלווה במתמטיקה די רצינית. אנסה לסקור אותו קצרות כי כאמור יש בו עומק מתמטי לא קטן אך עדיין ניתן להעביר את הרעיון העיקרי בלי לצלול יותר מדי לעומק.

כמו שאתם זוכרים במודלי דיפוזיה גנרטיביים יש לנו שני תהליכים: הקדמי והאחורי. תהליך הקדמי הוא הרעשה הדרגתית של דאטה והתהליך האחורי הוא הורדה הדרגתית של הרעש מהדאטה באמצעות מודל שאומן לצורך זה על דאטהסט מסוים. למעשה מודל כזה מאפשר ליצור דאטה מרעש טהור בצורה הדרגתית. הבעיה בתהליך הזה כמובן זה הזמן שזה לוקח כי צריך די הרבה איטרציות של denoising כדי להגיע מרעש לדאטה איכותי.

המאמר מציע דרך להקטין את מספר האיטרציות בדרך די מקורית. כמו שאתם זוכרים תהליך ההרעשה (הקדמי) במודלי דיפוזיה רגילים הוא מרקובי, כלומר הדאטה באיטרציה t מוגדר (מבחינת התפלגות) על ידי הדאטה המורעש מאיטרציה t-1 בלבד כל. המאמר הורס את ההנחה הזו ומגדיר תהליך קידמי לא מרקוב כאשר הדאטה באיטרציה t מוגדר לא רק על ידי הדאטה באיטרציה t-1 אלא גם על ידי הדאטה הנקי (x_0).

הנחה זה מאפשרת לנו להגדיר תהליך דטרמיניסטי של x_t-1 מ x_t באמצעות מודל שמאומן לשערך x_0 (הדאטה התחלתי מ-x_t). כלומר בכל איטרציה אנו קודם כל משערכים את x_0 באמצעות המודל ולאחר מכן בונים בצורה דטרמיניסטי אנו מחשבים x_t-1 מ-x_0 המשוערך.

אבל איך זה בעצם כאשר לזירוז של תהליך יצירת הדאטה? מתברר ששערוך של x_t דרך שערוך x_0 מאפשר להקטין משמעותית את מספר האיטרציות וככה הדאטה נוצר מהם יותר.

מאמר מאוד מעניין - הסברתי אותו ממש בגדול, חובת קריאה לכל מי שאוהב מודלי דיפוזיה גנרטיביים.
https://arxiv.org/pdf/2010.02502

⚡🚀המאמר היומי של מייק 01.08.24: ⚡🚀
IMPROVED TECHNIQUES FOR TRAINING CONSISTENCY MODELS

היום סוקרים קצרות עוד מאמר בנושא קרוב לליבי - המשך של המאמר שסקרנו לפני בערך שבוע הנקרא "consistency models". אם אתם זוכרים מודל קונסיסטנטי הוא שייך למשפחת מודלי דיפוזיה (כלומר הוא מתואר על ידי משוואת הדיפוזיה). אחת הבעיות של מודלי דיפוזיה קלאסיים (כמו DDPM) היא איטיות של גנרוט דאטה. הדאטה נוצר באמצעות תהליך denoising הדרגתי - מתחילים עם רעש גאוסי ומסירים אותו לאט לאט.

כדי להתמודד עם הבעיה הזו הוצעו כמה שיטות ואחת מהן DDIM סקרנו אתמול. השנייה היא מודלים קונסיסטנטיים(CM) שניתן להגדיר אותם כי מודל שונה (אך דומה) ממודל דיפוזיה קלאסי. בעיקרון ב-CM אנו מאמנים מודל להסיר רעש מכל פיסת דאטה מורעש באיטרציה t כך שהתוצאה תמיד תהיה פיסת הדאטה מקורית (ללא רעש). מכאן בא שם של המודל: קונסיסטנטי.

איך זה למעשה נעשה? יש שתי דרכים עיקריות לאמן CM. דרך אחת מסתמכת על מודל המשערך את מה שנקרא score function שהיא לוגריתם של פונקציית ההסתברות של הדאטה המורעש באיטרציה t. ידוע כי תהליך גנרוט של דאטה במודלי דיפוזיה (כלומר denoising) מתואר על ידי משוואת זרימה (דיפרנציאלית) שמתאר את המסלול של דאטה מהרעש עד הדאטה הנקי. ו- score function מופיע במשוואת זרימה זו. אז השיטה הראשונה ממזערת את המרחק בין שערוך x_0 (הדאטה הנקי) מ x_t+1 לבין שערוך של x_0 מ- x_t כאשר x_t מחושב ממשוואת הזרימה (איטרציה אחת של אוילר של משוואת הזרימה שכבר הזכרנו). ו

דרך אגב שערוך של score function די שקול לשערוך של הרעש הנוסף (לדאטה) במודלי הדיפוזיה הסטנדרטיים. הדרך השנייה "ליצור" את x_t+1 היא לשערך את x_0 מ-x_t+1 ולהוסיף רעש (כמו באיטרציה t).

המאמר המקורי על CM השתמש במרחק הנקרא LPIPS המודל דמיון סמנטי בין התמונות (דרך השוואה של אקטיבציות של מודלים מאומנים על דאטהסטים ענקיים של תמונות). המאמר המקורי גם התשמש ב-EMA (החלקה מעריכית) של משקלי המודל בתור המודל עבור x_t. יש כמובן חשיבות לבחירת השונות של האיטרציות.

אז המאמר שסוקרים היום משפר את תהליך האימון. השינוי הראשון הוא משקול של המרחקים כפונקציה של איטרציה t; ככל שמתקרבים ל 0 המשקול עולה. דבר שני זה שינוי של פונקציית מרחק מ-LPIPS לפונקציית הובר (Huber) עם טוויסט קטן. הדבר האחרון והמעניין הוא ביטול של EMA ל-x_t כלומר ההשוואה מתבצעת בין שני מודלים "טהורים" ל- x_t ו- x_t+1. גם הייפר פרמטרים אחרים עבור שינוי למשל השוניות של הרעש באיטרציות.

בקיצור יש לנו כאן שכלול מעניין של CM - בקרוב אסקור עוד מאמרים על זה...

⚡🚀המאמר היומי של מייק 02.08.24: ⚡🚀

NEFTUNE: NOISY EMBEDDINGS IMPROVE INSTRUCTION FINETUNING

הסקירה הזו הולכת להיות קצרה במיוחד. זוכרים שאחרי אימון מודל שפה אנו עושים לו מה שנקרא instruction fine-tuning. כלומר אנו מאמנים מודל שפה לעקוב אחרי הוראות המשתמש. בשביל זה בונים דאטהסט של שאלות ותשובות רצויות ולאחר מכן מטייבים (שם נוסף לפיין טיון) את המודל על הדאטהסט הזה עלי חיזוי של טוקן הבא של התשובה. המאמר מציע להוסיף רעש לייצוגי הטוקנים המופקים עלי ידי המודל באימון. כלומר אחרי כל מיניבאץ מעבירים את הטוקנים של השאלה והתשובה (אחד אחרי השני), מוסיפים רעש יוניפורמי בין -1 ל-1 לאמבדינגס וממשיכים לאמן. לא ברור אחרי איזה שכבה מוסיפים את הרעש (לדעתי יש משהו ב-ablation).

יש לי תחושה שהרעיון הזה לא חדש אך לפני המאמר הזה לא השתמשו בו ל- instruction fine-tuning.
https://arxiv.org/abs/2310.05914

⚡🚀המאמר היומי של מייק 03.08.24: ⚡🚀

Consistency Models Made Easy

כבר דיברנו רבות על מודלים קונסיסטנטיים (Consistency Models) או CM שהם בעצם שיפור של מודלי דיפוזיה גנרטיביים. בגדול יעד האימון של CM הוא למזער הפרשים בין חיזוי של פיסת דאטה נקייה מפיסות דאטה מורעשות איטרציות עוקבות. כלומר לוקחים פיסת דאטה מורעשת מאיטריה i ומאיטרציה i+1, חוזים את x_0 משניהם ומאמנים את המודל להגיע לאותה התוצאה. מכאן בא השם - Consistency Models.

המאמר מציע להכליל את השיטה הזו לא רק לאיטרציות עוקבות i ו- i+1 אלא לחיזויים מפיסות דאטה משתי איטרציות כלשהן t ו- s. ד״א המאמר מציג את בצורה קצת מורכבת - מסמן חיזוי מאיטרציה t בתור y_t ואז הנגזרת של y_t לפי t צריכה להיות 0 ומאמנים את המודל על דיסקרטיזציה של המשוואה הזו ברמות שונות.

אבל כאמור הכל מסתכם למזעור של ההפרשים בין החיזויים עבור איטרציות t ו- s שונות במהלך האימון עבור t ו-s נבחרו באקראי. כל הפרש כזה ממושקל ביחס הפוך לריבוע של t-s (זה הגיוני כי רמות רעש קרובות צריכות להסתכם בחיזויים קרובים ממש). עוד פרט חשוב: מתחילים את האימון ממודל דיפוזיה מאומן (למשל מ- DDIM).

https://arxiv.org/pdf/2406.14548

⚡🚀המאמר היומי של מייק 05.08.24: ⚡🚀

Improving Text Embeddings for Smaller Language Models Using Contrastive Fine-tuning

חוזרים לסקור מאמרים קלילים על מודלי שפה והיום בפוקוס מודלי שפה קטנים. המאמר שנסקור קצרות היום מציע שיטה לשיפור ייצוג של טקסט המופק על ידי מודל שפה קטן. ידוע שמודל שפה קטן (במאמר שיפרו את הייצוגים של הדקודרים) לא תמיד מצטיין ביצירה של ייצוג (אמבדינג) עוצמתי של טקסט - פשוט בגלל הגודל ו-expressiveness נמוכה יחסית.

אז המאמר מציע להשתמש בשיטת למידה ניגודית (contrastive learning) כדי לשפר את הביצועים. בגדול למידה ניגודית מאמנת מודל (לייצוג דאטה) במטרה לקרב פיסות דאטה (למשל תמונות או טקסט) שהן קרובות (סמנטית או בעלות אותה משמעות) ובאותו הזמן להרחיק את הייצוגים של פיסות דאטה לא דומות. השיטה הוצגה ב- 2018 על ידי Oord האגדי.

המאמר מציע להשתמש בלמידה ניגודית כדי לעשות פיין טיון לייצוגי הדאטה המופקים על ידי מודל שפה בפרט הפלט של השכבה האחרונה עבור טוקן EoS המסמן את סוף המשפט. עדכון משקלי המודל נעשה כמובן עם LoRA על דאטהסט המכיל משפטים בעלי משמעות קרובה וגם זוגות משפטים רחוקים סמנטית. המחברים טוענים שזה משפר את איכות הייצוג המופק על ידי המודל למספר משימות downstream (בפרט סיווג).

מאמר קלילי ונעים לקריאה....

https://arxiv.org/abs/2408.00690

⚡🚀המאמר היומי של מייק 06.08.24: ⚡🚀

TurboEdit: Text-Based Image Editing Using Few-Step Diffusion Models

חוזרים לסקור מאמרים על מודלי דיפוזיה עם מאמר כחול לבן של קבוצת חוקרים מאוניברסיטת תל אביב. הם מציעים שיטה מעניינת לעריכה מהירה של תמונה. כלומר בהינתן תמונה עם פרומפט נתון c אנו רוצים ליצור תמונה עם פרומפט אחר c1.

כמו שאתם זוכרים מודלי דיפוזיה מגנרטים תמונה על ידי הסרה רעש הדרגתית (denoising). בכל שלב המודל חוזה כמה רעש צריך להסיר מהתמונה והרעש המשוערך הזה מחוסר מהתמונה המורעשת באיטרציה הקודמת. השיטה הפשוטה לעשות עריכה של תמונה היא:

- להחסיר מהתמונה(המקורית) באיטרציה t את הרעש הזה המשוערך עם פרומפט c (כמו שעושים כאשר אין עריכה)

- להוסיף אל התוצאה את התוחלת המשוערכת של התמונה המורעשת(הערוכה) עם הפרומפט c1 החדש (עם התמונה המורעשת הערוכה.

כלומר בכל איטרציה מתקנים את הסרת הרעש בכיוון הפרומפט החדש.

דרך אגב ניתן שערוך הרעש הנוסף באיטרציה t ושערוך תוחלת התמונה אחרי הסרת הרעש אלו שתי בעיות שקולות, כלומר אחת מהן היא פשוט רפרמטריזציה של השנייה מבחינת השערוך.

הבעיה בשיטה הפשוטה לעריכת תמונות שהיא לא עובדת טוב ויוצרת ארטיפקטים בתמונה הערוכה. המחברים מנצלים מחקר קודם שמצא שהסקייל של הרעש (כלומר ההפרש בין התמונה המורעשת לתוחלתה) לא מתנהג לפי הסקייל של התהליך הקדמי של הדיפוזיה של התמונה המקורית (שבו מוסיפים רעש עם שונות עולה לתמונה עד שזו הופכת לרעש טהור). הרעש שנוצר במהלך עריכה כזו הוא בעל שונות משמעות גדולה יותר מאשר זה של התמונה המקורית.

אז המחברים מציעים להחסיר מהתמונה המורעשת המקורית באיטרציה t את שערוך התוחלת של התמונה המורעשת עבור האיטרציה t+d עבור d חיובי שהם מצאו. כלומר לוקחים תמונה x_t ומזינים אותה למודל שערוך התוחלת עם מספר איטרציה t+d. בסוף מכוונים את התמונה עם שערוך תוחלת המשוערכת של התמונה הערוכה עם איטרציה t+d.

בנוסף המאמר מציע דרך מעניינת לווסת את "עוצמת העריכה" בצורה דומה ל classifier guidance כדי לכוון את התוצאה של מודל דיפוזיה גנרטיבי ללא פרומפט עבור פרומפט נתון. הפעם על ידי ניתוח של נוסחת העריכה המחברים משקול של מרחק cross-prompt (הפרש שערוך התוחלת עבור התמונה הערוכה המורעשת עבור פרומפטים c ו- c1) לבין מרחק cross-trajectory שמודד הפרש בין חיזוי התוחלת בין התמונה הרגילה לתמונה המשוערכת). משקול כזה מאפשר לבצע את העריכה בפחות איטרציות denoising.

מאמר כתוב יפה ובהחלט מומלץ

https://arxiv.org/abs/2408.00735

⚡🚀המאמר היומי של מייק 07.08.24: 🚀⚡
Language Model Can Listen While Speaking

המאמר שמשך את תשומת ליבי בגלל שמו הקליט. המאמר מציע ארכיטקטורה של מודל Speech Language Model או SLM שיודע להקשיב תוך כדי שהוא מדבר, כלומר מודל full duplex (מושג מתחום התקשורת). בדרך כלל ל- SLM יש שני משטר עבודה: הקשבה או דיבור, כלומר המודל או מדבר או מקשיב. המאמר מעשיר את מרחב היכולות של SLM ומצייד אותו ביכולת להקשיב תוך כדי שהוא מדבר. מעניין שהמודל גם יכול לעצור אם הוא מזהה שיש דיבור (לא רעש) ומגיב עליו (בדיבור) לאחר מכן.

הארכיטקטורה של המודל המוצע LSLM מורכב מרכיבים סטנדרטיים. יש מודל שקולט אות דיבור, מחלק אותו לטוקנים (האות במקטעי זמן שונים) מקודד אותו לוקטור אמבדינג ומאזין אותו לדקודר. תפקיד הדקודר הוא לקחת בחשבון את ייצוג של טוקני הדיבור שנקלטו קודם וגם ייצוג טוקני הדיבור שנוצרו על ידי המודל כדי ליצור את הפלט הבא (אות הדיבור) של המודל. כאמור לפעמים הדקודר מחליט שהוא צריך לעבור למצב האזנה ולפעמים הוא צריך לעבור למצב הדיבור.

כלומר הדקודר במקרה הזה הוא vocoder המקבל כקלט את אות הדיבור הנקלט בנוסף לאות הדיבור המגונרט על ה-vocoder עצמו לפני.

https://arxiv.org/pdf/2408.02622

⚡🚀המאמר היומי של מייק 08.08.24: ⚡🚀
Masked Attention is All You Need for Graphs

היום סוקרים מאמר בנושא של גרפים, ומכיוון שאני סוקר מאמרים על למידה עמוקה המאמר הזה יהיה על רשתות עמוקות על גרפים או GNN. המאמר מציג גישה אלגנטית להפקת ייצוג (כלומר אמבדינג) של גרף וגם להפקת ייצוגם של צמתי הגרף או קשתותיו.

הגישה שהמאמר מציע הינה די פשוטה והייתי קצת מופתע שאף אחד לא עלה על זה קודם. למעשה המאמר מציע למסך (כלומר להעלים מהגרף) חלק מהמאפניים שלו. דרך אחת למסך (ברמה של צמתים) היא לאפס איברים מסוימים במטריצת שכניות (adjacency matrix) של הגרף (המתארת קשרים בין צמתים) או איברים ממטריצה שכניות של הקשת (node adjacency matrix) המתארת קשתות שיש להם צומת משותפת.

בשני המקרים המטרה היא לחזות את האיברים הממוסכים. המאמר משתמש בארכיטקטורה של set transformer (הרי בגרף אין חשיבות לסדר הצמתים והקשתות). הם לקחו ארכיטקטורת טרנספורמר מרובה ראשים די סטנדרטית למשימה הזו. הארכיטקטורה מורכבת מהאנקודר ומהדקודר (encoder-decoder transformer) כאשר לייצוג הגרף אנו משתמשים באנקודר ועבור ייצוג הקשות והצמתים משתמשים באדקודר.
https://arxiv.org/abs/2402.10793

⚡🚀המאמר היומי של מייק 09.08.24: ⚡🚀
Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters

בטח שמעתם על חוקי הסקיילינג של מודלי שפה. חוקים אלו מיועדים למציאת "קונפיגורציה" אופטימלית לאימון מודלי שפה. חוקי סקליינג מקשרים ערך של פונקציית לוס (ניתן להגדיר אותו בכמה אופנים) שניתן להשיגו עבור גודל מודל, גודל סט האימון וכמות משאבי החישוב (FLOps) המוקצית לאימון.

המאמר שואל האם ניתן לנסח חוקי סקיילנג דומים עבור האינפרנס, כלומר מה הביצועים המקסימליים שניתן להפיק בהינתן כמות משאבי חישוב נתונה. הרי יש כמה שיטות לבצע אינפרנס של מודל השפה ויש כמה פרמטרים חשובים של האינפרנס המשפיעים בצורה משמעותית על הביצועים. למשל יש שיטה הנקראת beam search שיוצרת בכל חיזוי של טוקן M סדרות טוקנים בעלי נראות (likelihood) הגבוהה ביותר. קיימות שיטות beam search עם מספר הסדרות השמורות לא קבוע ותלוי במספר הטוקן המגונרט.

יש שיטות איטרטיביות אחרות כמו במאמר "Consistency LLMs" שסקרתי לפני כמה שבועות. הוצעו גם שיטות שמשערכות את "איכות" התשובה המגונרטת (עם מודל מאומן נוסף) שמאפשר לבחור את התשובה הכי טובה מכמה תשובות מגונרטות (או להפסיק את יצירת התשובה אם רואים שהיא לא "בכיוון). כל שיטה כזו דורשת משאבי חישוב שונים שתלויים גם בהייפרפרמטרים של השיטה.

מה השיטה העדיפה לרמת ביצועים אופטימלית בהינתן תקציב חישוב נתון (FLOps) - זו השאלה שהמאמר מנסה לענות עליה ויש תוצאות מעניינות (לדעתי)

https://arxiv.org/abs/2408.03314

⚡🚀המאמר היומי של מייק 10.08.24: ⚡🚀
Synthesizing Text-to-SQL Data fromWeak and Strong LLMs

הסקירה של היום הולכת להיות די קצרה וקלילה. המאמר מציג שיטה די אינטואיטיבית לאמן מודל שפה קטן לבצע משימה מסוימת. במקרה שלנו המשימה היא גנרוט של שאילתת SQL לפי תיאורה הטקסטואלי ומבנה (schema) של הטבלה. מודלי שפה קטנים עלולים להסתבך עם המשימה הזו בטח במקרים שהשאילתה הנדרשת אינה טריוויאלית.

המאמר מציע תהליך דו שלבי של אימון מודל קטן למשימה זו. בשלב הראשון יוצרים דאטהסט עבור המשימה הזו באמצעות מודלי שפה גדולים וחזקים וכמה דאטהסטים רלוונטיים. עושים דברים רגילים, הנדסת פרומפטים קלה וכאלו. לאחר מכן עושים למודל הקטן פיין טיון על הדאטהסט הזה.

בשלב השני עושים למודל השפה הקטן Direct Policy Optimization או DPO שראינו אותו כשלב אימון מודלי יסוד (foundational). היתרון של שיטה זו היא בכך שהיא לא דורשת אימון של מודל reward. בשביל אימון מודל כזה אנו צריכים דוגמאות טובות ודוגמאות לא טובות. דוגמאות טובות יש לנו מהשלב הראשון.

בשביל לבנות את הדוגמאות הרעות לוקחים את המודל הקטן המתקבל על השלב הראשון כדי לגנרט שאילתת SQL לתיאור טקסטואלי נתון. לאחר מכן מריצים את השאילתה כדי לוודא האם התוצאה המתקבלת נכונה. אם היא לא נכונה קיבלנו דוגמא שלילית. ככה בונים דאטהסט של דוגמאות חיוביות ושליליות ומה ש נותר לעשות הוא PPO.

https://arxiv.org/abs/2408.03256

⚡🚀המאמר היומי של מייק 12.08.24: ⚡🚀
Img-Diff: Contrastive Data Synthesis for Multimodal Large Language Models

מודלי דיפוזיה גנרטיביים הגיעו לתוצאות מרשימות לאחרונה והפגינו יכולת לגנרט תמונות באיכות מרהיבה. למרות זאת מודלים אלו מתקשים לפעמים במשימות של עריכת תמונות ולא מצליחים להחליף אובייקטים לא גדולים בתמונה תוך שמירה של כל המאפיינים האחרים של התמונה.

המאמר המסוקר מציע שיטה ליצירת דאטהסט של זוגות תמונות שכל זוג מכיל תמונות זהות פרט לאובייקט אחד בתמונה. כל זוג תמונות מלווה בתיאור של האובייקטים שהוחלפו בשתי התמונות וגם במיקומם בתמונות. בין השאר דאטהסט זה יכול לשמש חוקרים ומהנדסים לאימון מודלים לעריכת תמונות.

איך הם עשו זאת? האמת הפייפליין שלהם די מורכב מכיל הפעלה לא מעט מודלים מולטימודליים, ומודלים לזיהוי ותיאור אובייקטים בתמונה כמו LLAVA, FastSAM, BLIP, CLIP וכדומה. נתאר רק את ה 3 השלבים של התהליך.

בשלב הראשון לקחו כמה עשרות אלפי תמונות מהדאטהסט הידוע MS COCO ויצרו זוגות של תמונות דומות על ידי החלפה של אובייקטים מסוימים באובייקטים אחרים בתמונה עם המודל שנקרא ViCUNA (ההחלפה עצמה בוצעה עם המודל הנקרא InstructPix2Pix).

בשלב השני אנו מפעילים כמה מודלים מולטימודליים כדי לזהות את האיזורים בתמונות שעברו שינוי (בזוגות מהשלב הראשון). קודם כל המחברים את התמונות הלא דומות עם CLIP (כלומר בהתבסס על דמיון של ייצוגי

התמונות). לאחר מכן שוב מפלטרים את הדאטהסט על ידי התאמה של תיאורם של האובייקטים והימצאותם בשתי התמונות עם BLIP. בסוף מזהים את מיקום האיזורים בתמונה שבהם הוחלפו האובייקטים (כלומר bounding boxes שלהם).

בשלב האחרון מפיקים תיאור טקסטואלי של כל החלפות של בוצעו בתמונה הראשונה בזוג שהפך אותה לתמונה השנייה בזוג. עושים זאת עם שילוב של LLAVA ו- CLIP.

וככה מקבלים דאטהסט איכותי של זוגות תמונות דומות שמה שהשינוי ביניהם מתואר על ידי התוצאה של השלב האחרונה (כולל מיקום השינוי).

https://arxiv.org/abs/2408.04594

⚡🚀המאמר היומי של מייק 13.08.24: ⚡🚀

Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2

בזמן האחרון התחלתי להתעניין בשיטות interpretability של מודלי שפה גדולים בעקבות כמה בלוגים מאוד מעניינים של אנטרופיק, OpenAI ולאחר מכן גוגל בנושא הזה. המטרה כאן היא לשפוך קצת אור על הקופסא השחורה שנקראת LLM - הרי אנחנו לא באמת מבינים איך הם עובדים ומה גורם להם לפלוט תשובה כזו אור אחרת לפרומפט שלנו.

אז המאמר הזה חוקר אחת השיטות המנסות להבין איך מודל שפה מייצג קונספטים סמנטיים שונים. המאמר עושה זאת דרך חקר של אקטיבציות הנוירונים בשכבותיהם השונות של מודלי שפה. עקב כך שיטה זו משויכת למשפחת שיטות המכונות mechanistic interpretability. הרעיון שהמאמר דן בו נקרא SAE או Sparse AutoEncoders.

אז מה הרעיון העיקרי ב- SAE? אנו מנסים להציג אקטיבציות של שכבה מסוימת של LLM על יד וקטור ארוך הרבה יותר מווקטור האקטיבציות אך מאוד דליל. כלומר וקטור n-ממדי של האקטיבציות אנו מייצגים (עם SAE) עם וקטור באורך n << M אך בווקטור האורך הזה יש פחות מ- n איברים לא שווים לאפס (דלילות). SAE במקרה הזה פשוט מאוד: שכבה אחת לינארית עם אקטיבציה לא לינארית באנקודר (של SAE) ושכבה אחת של דקודר. המטרה כמובן לאמן את SAE כך שיהיה ניתן לשחזר את האקטיבציות המקוריות מייצוגם (אחרי האנקודר).

אבל למה זה בכלל חשוב ואיך זה קשור ל-interpretability של LLMs. הנחת מוצא של גישה זו (הבלוג של אנטרופיק מדבר על זה בהרחבה) שכל נוירון (או קבוצת נוירונים) בשכבה (מסוימת) הוא "נדלק" (מקבל ערכים) על כמה קונספטים לא קשורים (נגיד כלב, מכונה וערפל). כלומר הוא סוג של תערובת עבור כמה קונספטים. אז הייצוג המופק על ידי SAE הוא למעשה מהווה ייצוג של כל קונספט (disentangled). כלומר עבור כל קונספט המקודד קבוצות נוירונים שונות בוקטור הדליל הזה.
אז מה המאמר הזה עושה? הוא מנסה לאתר שכבות שבהם SAE מאומן עם שגיאת שחזור מינימלית (עם רגולריזציה מתאימה) כלומר הוא מנסה להבין איזו שכבה ב-LLM (וגם בשכבות הפנימיות של בלוקי הטרנספורמר) מקודדת הכי טוב את הקונספטים הסמנטיים.

בימים הקרובים עוד כמה סקירות בנושא המרתק הזה.

https://arxiv.org/abs/2408.05147

⚡🚀המאמר היומי של מייק 14.08.24: ⚡🚀

Jumping Ahead: Improving Reconstruction Fidelity with JumpReLU Sparse Autoencoders

אתמול סקרנו מאמר שהשתמש בגישת SAE או Sparse AutoEncoders כדי לחדור ל"מחשובותיו" של מודל שפה גדול דרך האקטיבציות של הנוירונים שלהם. הנחת היסוד במאמר היתה כי נוירונים "מגיבים" לכמה קונספטים שונים וניתן לאמן SAE רדוד מאוד (שכבה אחת בדקודר ושכבה אחת באנדוקר) כדי להגיע לוקטור דליל המקודד (נדלק) קונספט אחד בלבד כלומר disentanglement של הפיצ'רים לנוירונים ייעודיים.

כמאמר יש באנקודר של SAE שכבה לינארית אחת עם פונקציית אקטיבציה הנקראת JumpReLU שראיתי אותה בפעם הראשונה במאמר הזה. פונקציה הזו היא בעצם הזזה של ReLU בציר X ובציר y בפרמטר t נלמד (במאמר זה נקרא טטה). הטענה במאמר שזה מאפשר ללמוד את הייצוג הדליל של דאטה על ידי האנקודר יותר טוב של פונקציית ReLU בגלל שהוא מאפשר לאפס את הקטיבציות בצורה "נלמדת יותר מ-ReLU".

עכשיו נשאלת השאלה איך אנחנו אוכפים דלילות על ייצוג הדאטה (אחרי האנקודר). בעבודות קודמות השתמשו ב-L1 בשביל כך אך כאן המחברים משתמשים באותה JumpReLU כדי להפוך את איפוס האיברים בייצוג יותר נלמד. ושימו לב ש- JumpReLU בא עם פרמטר נלמד הזה לזה של האנקודר עצמו שזה עוזר לאכוף דלילות על הייצוג.

יש עוד טריק אחד קטן ולא מאוד מהותי במאמר הנקרא Kernel density estimation או KDE. אם אתם זוכרים KDE עוזר לנו לשערך(כלומר לקרב) פונקצית צפיפות בהינתם דאטהסט של נקודות באמצעות פונקציית קרנל. פונקציית קרנל יכולה להיות גאוסית למשל ומטרתה לשערך את פונקציית הצפיפות לנקודות לא ידועות על ידי קירובה בין הנקודות בדאטהסט (בדומה לספליין). אז המחברים משתמשים בטריק הזה כדי לשערך את JumpReLU בנקודה t שבה היא לא גזירה.

מאמר נחמד בנושא די חשוב שאמשיך לסקור כנראה גם בעתיד...

https://arxiv.org/pdf/2407.14435

⚡🚀המאמר היומי של מייק 15.08.24: 🚀⚡

Your Classifier Can Be Secretly a Likelihood-Based OOD Detect

משנים טיפה את הכיוון היום וסוקרים מאמר לא על LLM. המאמר דן בזיהוי של דאטה שלא מתפלג לפי התפלגות הדאטה במהלך אימון המודל. למשל אימנתם מודל לזהות חתולים, כלבים וסוסים ופתאום מפעילים את המודל שלכם על תמונה של טנק. אם לא נקטתם אמצעים נגד זיהוי דאטה מחוץ להתפלגות האימון (או OOD) אתם עלולים לזהות את הטנק הזה בתור אחת הקטגוריות שאימנתם את המודל עליהם כלומר בתור כלב, חתול או סוס.

כמובן שהמצב הזה מאוד בעייתי ועקב כך הוא נחקר רבות במהלך השנים האחרונות. המאמר שנסקור קצרות היום מציע שיטה מאוד אלגנטית וטבעית להתמודד עם הסוגיה הזו. המאמר מציע לאמן מודל לזהות קטגוריות היעד (שמופיעות בסט האימון) אלא גם לכפות התפלגות מסוימת על הייצוג שלהם המופק על ידי המודל (כלומר של הפלט של השכבה האחרונה של הרשת).

הפרמטרים של ההתפלגות הזו נקבעים מראש (הממוצע ופרמטר ששולט בכמה ההתפלגות מרוכזת סביב הממוצע - סוג של מטריצת קוואריאנס). ואם עבור דוגמא נתונה וקטור הייצוג יוצא רחוק מספיק מכל וקטורי הממוצע של כל הקטגוריות (כאשר מקדם הפיזור נלקח בחשבון) אז הדוגמא הזו מזוהה בתור OOD.

בתור התפלגות היעד המחברים לקחו התפלגות von Mises-Fisher על ספרה במימד של וקטור הייצוג p (כלומר הספרה היא במימד p-1). המחברים טוענים שזה עובד טוב יותר מאשר התפלגות גאוסית.

https://arxiv.org/abs/2408.04851

⚡🚀המאמר היומי של מייק 16.08.24: ⚡🚀
On the Geometry of Deep Learning

אני ממש אוהב מאמרים שחוקרים מה שקורה בתוך המודלים העמוקים שלנו - הרי לדעתי זה התנאי הכרחי לכך שנוכל להתחיל באמת לסמוך על- AI (לפחות חלקית). ואכן הכותבים מדגישים כי למידה עמוקה, על אף הישגיה המרשימים במגוון תחומים, נשארת עדיין בגדר "קופסה שחורה" עם הבנה חלקית בלבד של אופן פעולתה.

המחברים מנסים להסביר מודלים עמוקים באמצעות ספליינים אפיניים (Affine Splines) שהן למעשה פונקציות רציפות ולינאריות למקוטעין במרחב רב מימד. המחקר מתבונן ברשתות נוירונים מזווית גיאומטרית באמצעות ניתוח של חלוקות הנוצרות על ידי ספליינים אפיניים, המקרבות אותן (הרשתות).

בפרט המחברים דנים בחלוקות של מרחב הקלט לפי הקטגוריות שלו הנוצרות על ידי ייצוג לטנטי (השכבה האחרונה לפני שכבת הסיווג) של הרשת. הבנת החלוקה הזו מסייעת להסביר כיצד רשתות עמוקות לומדות ומייצרות חיזוים עבור קלטים שונים.

המחברים גם דנים במבנים גיאומטריים הנוצרים על ידי משקלי המודל במרחב הלוס (כלומר מנתחים את פונקציית הלוס למשקלי הרשת השונים). בנוסף המאמר גם מדבר על החלוקות הנוצרות במרחב משקולות המודל בשכבות שונות לאתחולי רשת שונים וגם לאימון עם ובלי BatchNorm. כמובן שזה נעשה על דוגמאות מלאכותיות(toy examples) בעלי מימד נמוך. ויש עוד מספר ניתוחים גיאומטרים די מעניינים במאמר.

מעניין כי המחברים כותבים כי אחת המטרות המרכזיות של המחקר היא לדרבן מתמטיקאים לעסוק בניתוח גיאומטרי של רשתות עמוקות.

⚡🚀המאמר היומי של מייק 17.08.24: ⚡🚀
Faster Machine Unlearning via Natural Gradient Descent

היום סוקרים מאמר כחול לבן בנושא מעניין הנקרא unlearning. בדרך כלל אנו מעוניינים שהמודל שלנו ילמד מהדאטה אבל כאן אנו רוצים שהמודל ישכח דאטה מסוים. הנושא די חשוב לחברות שרוצה להיות compliant עם הדרישות של תקנים סטייל GDPR כאשר יוזר או קבוצת יוזרים מקבשים למחוק את הדאטה שלו באופן מוחלט. כמובן שבנוסף למחיקת הדאטה עצמו צריך "למחוק" אותו מה"מוח" כלומר המשקלים של המודלים שאומנו (בפרט) על הדאטה הזה.
אחת השיטות הנאיביות לעשות unlearning היא למחוק את הדאטה ולאמן מודל מחדש. אבל זה יכול להיות די יקר ולא יעיל במיוחד למודלים גדולים. האם קיימת שיטה אחרת לעשות את זה?

אכן יש לא מעט מחקר בנושא של unlearning ואחת הגישות הפופולריות היא לקחת מודל מאומן ולמזער את הפרש של הביצועים על הדאטה שנותר והדאטה שאמור להימחק. כלומר אנו רוצים למזער את הלוס על הדאטה הנותר ולמקסם אותו על הדאטה שנמחק. ככה "נמחק" מהמוח(אולי הזכרון) של המודל את הדאטה המיועד למחיקה.

כמובן ששיטה נוספת "למחוק" את הדאטה מהמודל היא פשוט למקסם את הלוס על הדאטה המיועד למחיקה.

כמובן שיש שיטות רבות לעשות את זה באמצעות וריאציות שונות של מורד הגרדיאנט (SGD - stochastic gradient descent). המאמר מציע לעשות את זה עם מה שנקרא natural gradient או NG. זה קונספט פחות ידוע ואני אסביר אותו בקצרה. אתם בטח זוכרים מה זה קצב למידה ב-SGD, נכון? זה פרמטר קריטי לתהליך הלמידה וקיימות לא מעט שכלולים של SGD כמו ADAM ו-RMSProp שבפועל (בצורה לא מפורשת) קובעים את קצב הלמידה האופטימלי כתלות בפונקציית לוס.

יש כמובן דרך נוספת לבחור את קצב הלמידה בצורה אופטימלית וזה מה שעושה שיטת ניוטון קלאסית (נראה לי שזה השם) לאופטימיזציה. במקום להשתמש בקצב למידה סקלרי משתמשים בהופכית של ההסיאן של פונקציית לוס (מטריצה של נגזרות שניות). זה אופטימלי מבחינת ההתכנסות (כי משתמשים בקירוב טיילור מסדר שני של פונקציית לוס). אבל כמובן לא ניתן לעשות זאת לרשתות (יש קירובים אמנם) כי קשה מאוד להפוך מטריצה בגדול מיליארד על מיליארד.

המאמר מציע להחליף את ההיסאן ב- FIM או Fisher Information Matrix. למעשה FIM היא תוחלת של המכפלה הוקטורית של ה**גרדיאנט הלוג של הנראות (likelihood) של הדאטה המקורב על ידי המודל עם עצמו**. למעשה FIM מודד עד כמה שינוי בפרמטרים של המודל משפיע על הנראות של הדאטה באמצעות המודל (עם המשקלים הנוכחיים). זה בעצם מצביע לנו עד כמה הנראות של הדאטה רגישה לשינוי בערכי המודל.

יש ל-NG הרבה יתרונות (למשל הוא חסין לרפרמטריזציה של המודל) אבל כמו ההסיאן עדיין מאוד קשה לחשב אותו עבור מודלים ענקיים. כמובן שקיימות שיטות המחשבות אותו באופן מקורב באמצעות שילוב עם פונקצית רגולריזציה "נוחה".

בנוסף לעדכון הרגיל של הגרדיאנט כמו ב-SGD עם FIM (כלומר בהופכית שלו) המאמר משתמש במה שנקרא proximal operator כדי לתקן את משקלי המודל אחרי שעודכנו עם SGD. למעשה PO מתקן את משקלי המודל אחרי העדכון ולא מאפשר להם להתרחק יותר ממשקלי המודל לפני עדכון כאשר ה"מרחק" כאן מנורמל עם ההופכית של FIM תוך כדי לקיחה בחשבון של פונקציית רגולריזציה (שלא תתפוצץ).

המאמר די קשוח מתמטית ומקווה שהצלחתי לשפוך קצת אור עליו...
https://arxiv.org/abs/2407.08169

⚡🚀המאמר היומי של מייק 19.08.24: 🚀⚡
DIGRESS: DISCRETE DENOISING DIFFUSION FOR GRAPH GENERATION

היום סוקרים קצרות מאמר לא רגיל על מודלי דיפוזיה. אתם בטח זוכרים (וסקרתי לא מעט לאחרונה) מודלי דיפוזיה עבור תמונות, וידאו, אודיו וכדומה. במאמר שנסקור אותו היום מודל דיפוזיה נבנה על גרף. אציין כי המאמר מלפני שנה וחצי ולמיטב ידיעתי יצאו כמה מאמרי המשך.

אז מה זה מודל דיפוזיה רגיל ואיך מאמנים אותו? מודל דיפוזיה גנרטיבי מאומן על ידי הוספה הדרגתית של רעש לדאטה כאשר המטרה היא לאמן מודל המסיר את הרעש הזה (כלומר משחזר את הדאטה מאיטרציה הקודמת). מודל כזה מאפשר לנו לגנרט דאטה מרעש טהור על ידי הסרתו הדרגתית.

אבל איך ניתן "להטיל" את הרעיון הזה על גרפים? נניח שיש לנו גרף בו כל הצומת וכל קשת שייכים לקטוריה מסיומת (קטגוריות שונות לקשתות ולצמתים). עכשיו בתהליך קדמי (הוספת רעש) אנו בעצם משנים באקראי את הלייבלים (קטגוריות) של הצמתים ושל הקשתות לקטגוריה אחרת. כלומר צומת נתונה יכולה להישאר בקטגוריה שלה בהסתברות 0.95 ובהסתברות 0.05 היא תקבל כל לייבל אחר בצורה אחיד. תהליך דומה נעשה על הקשתות. בסוף התהליך הגרף הופך להיות עם קשתות וצמתים בעלי קטגוריות רנדומליות לגמרי.

כמו במודלי דיפוזיה המטרה של המודל המאומן (על דאטהסט של גרפים מתויגים) היא לשחזר את הלייבלים מהאיטרציה הקודמת (של הצמתים ושל הקשתות). זה יאפשר שחזור גרף עם התלויות כמו בסט האימון.

כמובן שיש כאן הרבה משחק על איך מרעישים את הלייבלים בתהליך קדמי. האם יש תלות בתהליך ההרעשה בין צמתים וקשתות שונים, אולי בהתחלה משנים לייבלים רק לתת-גרפים מסוימים וכדומה.

בקיצור מאמר מאוד מעניין ואני מניח שאסקור בעתיד גם מאמרי ההמשך שלו.
https://arxiv.org/abs/2209.14734

⚡🚀המאמר היומי של מייק 20.08.24: 🚀⚡
JPEG-LM: LLMs as Image Generators with Canonical Codec Representations

המאמר הזה תפס את עיניי כי מילה "jpeg" הופיע בשמו. למרות שלא יצא לי לעבוד בתחום של דחיסת דאטה אני מאוד אוהב את הנושא המרתק הזה. בנוסף המאמר הזה מדבר על מודל VQ-VAE שהיה די פופולרי לפני שמודלי דיפוזיה השתלטו לנו לחלוטין על GenAI בראייה הממוחשבת.

אוקיי, אז כל זה קשור? קודם כל jpeg זו גישה ידועה לדחיסת תמונות. המאמר גם מדבר על AVC/H.264 שהיא גישה לדחיסת וידאו המתבססת על עקרונות דומים לאלו של jpeg. בגדול jpeg עובד בצורה הבאה:

- מחלקים תמונות לפאצ'ים באותו הגודל ועושים לכל אחד DCT - Discrete Cosine Transform (כמו התמרת פוריה ללא החלק המדומה).
- מבצעים קווינטוט של מקדמים DCT לכל פאץ' כאשר המקדמים לתדרים גבוהים "נחתכים" בצורה רצינית יותר
- משתמשים בקידוד run length וגם בקידוד האפמן כדי לדחוס את כל המקדמים המקונטטים של הפאצ'ים.

אוקיי, עכשיו נרענן לכם מזה VQ-VAE. קודם כל VAE זה מודל גנרטיבי שלומד לגנרט דאטה מהייצוג הלטנטי שלו (במימד נמוך). VAE מורכב מהאנקודר מהדקודר שהראשון בהם מאומן להפיק ייצוג של דאטה במימד נמוך והדקודר משחזר את הדאטה ממנו. VAE מאומן בצורה המשרה התפלגות נתונה (בד"כ גאוסית) על המרחב הלטנטי וזה מאפשר לגנרט דאטה חדש באמצעות הדקודר מווקטור הדגום מהתפלגות זו.

VQ-VAE היא שכלול של VAE כאשר הוא מאומן לגנרט תמונה בצורה סדרתית (מפאצ'ים/טוקנים ויזואליים) כאשר כל פאץ מיוצג על ידי וקטור (לטנטי) מהמילון שנלמד גם כן. כלומר התמונה נבנית פאץ'-פאץ' כאשר כל פאץ' (כלומר וקטור מהמיליון שמייצג אותו) נדגם בהינתן כל פאצ'ים שכבר גונרטו. זה בטח מזכיר לכם מודל שפה שמגנרט טוקנים בדיוק באותה צורה.

VQ-VAE מאומן בשני שלבים: בראשון מאמנים את האנקודר, המילון והדקורד (המשחזר פאצ'ים מהווקטורים במילון) ובשלב השני מאמנים מודל לחזות טוקן ויזואלי הבא בהינתן הטוקנים שכבר נוצרו.

המחברים שילבו את הרעיונות האלו (חלקית) ואימנו מודל שיודע לחזות ייצוג jpeg או avc בצורה סדרתית. אבל מה הטוקנים כאן? בדומה למודלי שפה המחברים השתמשו ב-BPE או byte-pair encoding (עם שפצורים קלים). מכאן המחברים בנו מודל היודע לרנרט ייצוג jpeg של התמונה שניתן להפוך אותו לתמונה די בקלות.

רעיון די חמוד אבל יש לי הרגשה שכבר ראיתי רעיונות דומים בעבר...
https://www.arxiv.org/abs/2408.08459

⚡🚀המאמר היומי של מייק 21.08.24: ⚡🚀
Tree Attention: Topology-Aware Decoding for Long-Context Attention on GPU Clusters

היום נסקור מאמר בנושאה שכבר סקרתי כמה מאמרים לפני כחודש. הנושא הזה נקרא אופטימיזציה והאצה decoding של מודלי שפה כלומר התהליך שגנרוט טוקן חדש בתלות בכל הטוקנים בתוך חלון ההקשר שכבר גונרטו. ואם חלון ההקשר הוא ארוך (מאות אלפי טוקנים) זה יכול לקחת די הרבה זמן בעיקר בגלל מנגנון ה-attention של הטרנספורמרים שמהווים backbone של כל מודלי השפה החזקים.

בשנים האחרונות הוצעו מספר רב של שיטות לייעול והאצה של חישוב ה-attention שהכי מפורסמים מהם הם Flash Attention ו-KV-Cache. שיטות אלו בדרך כלל מנצלות את העובדה שהיום אינפרנס של מודלי שפה מתבצע על GPU וניתן לייעל את החישוב על ידי שימוש ביכולת של GPUs לחשב דברים במקביל.

יתרה מזו מכיוון שמודלי שפה רצים היום על קלסטרים של GPUs יצאו מספר עבודות על איך ניתן לחשב את ה-attention על קלסטרים אלו. מכיוון שמנגנון ה-attention מכיל מכפלות פנימיות (סכומים רבים) אז ניתן לחשבו בצורה מבוזרת די ביעילות.

והמאמר הזה מציע מנגנון מעניין של חישוב ה-attention. הדבר המעניין בו שהמאמר הזה מייצג את חישוב ה-attention (עבור וקטור שאילתה נתון q) כנגזרת של הלוג של "פונקציה יוצרת" של ה-attention המחושבת בנקודת 0. פונקציה יוצרת זו נבנית על ידי מניפולציה פשוטה של נוסחת ה-attention וממש מזכירה פונקציה יוצרת של משתנה אקראי.

ניתן להכליל את החישוב הזה ל-attention עבור וקטורי שאילתה q מרובים כאשר במקום נגזרת רגילה יהיה לנו נגזרת לפי n משתנים (n הינו מספר וקטורי השאילתה).

למה זה טוב בכלל? מתברר שהחישוב של attention בצורה כזו מערב פעולות כמו logsumexp ו- max שניתן לבזר אותם בצורה יעילה בין ה-GPUs. החישוב נעשה בצורה של עץ, כלומר מחלקים את הסכומים לכמה חלקים, מחשבים כל חלק ואז מתחילים לסכם את התוצאות בצורה היררכית. זה כמו Map-Reduce רב שלבי.

https://arxiv.org/abs/2408.04093

⚡🚀המאמר היומי של מייק 22.08.24: ⚡🚀
Approaching Deep Learning through the Spectral Dynamics of Weights

היום נסקור מאמר החוקר מה הסיבות לתופעה של גרוקינג. למי שלא מכיר גרוקינג זו תופעה די מעניינת המתרחשת כאשר ממשיכים לאמן רשת נוירונים (למרות שזה קורה גם במודלים אחרים) גם אחרי לוס הווליידציה מתחיל לעלות (כלומר אנו נכנסים למשטר אוורפיט). מתברר אם לא עוצרים וממשיכים לאמן לוס הווליידציה מתחיל לרדת כלומר המודל נכנס למשטר ההכללה כלומר לומד את ה"חוקיות האמיתית" מאחורי הדאטה.

התופעה הזו היא מקרה פרטי של double descent (יש גם multiple descent) שמתרחש גם אם אנו מוסיפים פרמטרים למודל בצורה עקבית ומגיעים למצב שיש לנו over-parametrization. כלומר יש המודל שלנו לכאורה מתחיל "יותר מדי פרמטרים" כדי "להבין את הדאטה". וגם שם זה קורה בצורה בלתי רציפה כלומר יש אינטרוול של פרמטרים שביצועי המודל יורדים עבורם ורק אז מתחילים לרדת.

המאמר חוקר מה קורה עם משקלי המודל כאשר הוא נכנס למשטר הגרוקינג. מתברר שתופעה הגרוקינג קשורה לירידה בראנק של מטריצות המשקלים של המודל. בשבילי זה די אינטואיטיבי כי לדעתי במהלך גרוקינג המודל מצליח להתכנס ל"פתרון פשוט ביותר עבור הדאטהסט. פתרון פשוט הכוונה הוא מודל שאפקטיבית הוא קטן, כלומר רוב וקטורי המשקלים בו או אפס או תלוים לינארית זה בזה.

⚡🚀המאמר היומי של מייק 30.08.24: ⚡🚀

Platypus: A Generalized Specialist Model for Reading Text in Various Forms

חוזרים לסקירות אחרי שבוע של חופשה עם מאמר בנושא שלא סקרתי די הרבה זמן והוא Optical Character Recognition או OCR בקצרה. מטרת OCR היא לזהות טקסט בתמונה או במסמך כאשר הטקסט יכול להופיע בצורות ומגוונות. מודלי OCR הקודמים בדרך כלל התמקדו בזיהוי של סוג של טקסט (נגיד נוסחה, טקסט מודפס או כתב יד). המחברים מציעים גישה שמאחדת את מומחי ה-OCR ה"צרים" לזיהוי סוג ספציפי של טקסט - כלומר מסוגלת לזהות כל סוג של טקסט בתמונה כולל המקרים שיש כמה סוגים של טקסט בתמונה.

בנוסף ב-OCR יש 3 משטרי הפעלה. הראשון זה RAT או Recognize All Text שמטרתו לזהות את כל הטקסטים בתמונה. השני הוא PPR או Point Prompt Recognition שמטרתו לזהות את הטקסט סביב נקודה נתונה (סוג של עוגן) בתמונה. האחרון הוא Box Prompt Recognition או BPR שמיועד לזיהוי של טקסט בתוך מלבן נתון בתמונה (כמו שיש לנו Bounding Boxes בזיהוי אובייקטים בתמונה אבל בכיוון ההפוך).

אז המחברים מאמנים מודל המורכב מהאנקודר (שהופך תמונה לאמבדינג) הדקודר האוטורגרסיבי. הדקודר מקבל כקלט את סוג הטקסט בתמונה (מודפס או כתב יד). בנוסף הדקודר מקבל את סוג המשימה (RAT, PPR או BPR) עם כל הפרטים הנחוצים לביצוע משימה (כלומר קואורדינטות של הפאץ'). בנוסף המודל מקבל גרנולריות של זיהוי הטקסט (כלומר word-level או line-level שהראשון הוא זיהוי מילה בודדת והשני הוא זיהוי טקסט שלם). הפרטים האלו מוזנים כאמור לדקודר שמטרתו לגנרט את הטקסט המופיע בתמונה.

זה כל הפרטים המעניינים - מאמר די קליל....

⚡🚀המאמר היומי של מייק 31.08.24: ⚡🚀

Counterfactual Explanations and Algorithmic Recourses for Machine Learning: A Review

היום סוקרים מאמר לא רגיל - קודם כל זה מאמר סקירה בעצמו והוא לא מאוד טרי (מלפני כמעט שנתיים). המאמר בנושא של explainability של מודלי למידת מכונה. רוב מודלי ML היום הם רשתות נוירונים מאוד עמוקות ולרוב הם נשארים בתור קופסא שחורה עבורנו - מחקרי explainability מנסים לשפוך אור על "מה שקורה בתוך הקופסא השחורה הזו".

המאמר הזה נותן סקירה של אחד הפרדיגמות העיקריות המשמשות למחקר explainability של מודל ML - ניתוח counterfactual. כלומר חוקרים מה צריך לשנות בדגימה(איזה פיצ'רים) כדי שהיא תסווג לקטגוריה (קלאס) אחרת ועל ידי כך נבין יותר טוב למה המודל סיווג את הדוגמא המקורית לקטגוריה המקורית. דרך אגב יש שיטות explainability שחוקרות את המודל בצורה אחרת. למשל קיימות שיטות שמנסות לקרב את המודל

המורכב על ידי מודל פשוט יותר (עץ או רגרסיה לינארית) בטווח מסוים של דוגמאות. שיטות נוספת מנתחות המנסות להסביר את חיזויו של המודל לדוגמא ספציפית.

אז מה בעצם חשוב לנו מאוד בשיטות counterfactual? קודם כל חשוב לנו לשנות כמה שפחות פיצ'רים של הדוגמא הנחוצים ל"העברתה" לקטגוריה אחרת וגם השינוי בפיצ'רים אלו צריך להיות די קטן כדי להבין את "מבנה גבול" בין הקטגוריות השונות מבחינת המודל. השני ולא פחות חשוב השינוי הזה צריך להיות "חוקי" כלומר הדוגמא הנוצרת צריכה להיות הגיונית וולידית (כלומר שטח הבית לא יכול להיות שלילי). בנוסף השינוי בדוגמא צריך לעבור במסלול הגיוני כלומר בקרבה של הדוגמאות האחרות מהדאטהסט. וכמובן יש עוד דרישות לשינוי שאנו מחוללים לדוגמא כדי להפוכה ל-counterfactual.

וכל הפרטים המעניינים במאמר כמובן...

https://arxiv.org/abs/2010.10596

⚡🚀המאמר היומי של מייק 01.09.24: ⚡🚀

DIFFUSION MODELS ARE REAL-TIME GAME ENGINES

טוב, על המאמר הזה פשוט לא היה לדלג מכמה סיבות. הסיבה הראשונה שאני מספיק עתיק ועוד שיחקתי במשחק הנקרא דום (doom) במו ידיי כאשר הייתי נער. דבר שני לא כל יום מחליפים לך מנוע משחק במודל למידת מכונת או בשמו המוכר AI. כמובן שזה כיוון מחקר מאוד מעניין עם פוטנציאל להתפתח לכלים מבוססי AI לבניית משחקי מחשב חדשים.

הרעיון של המאמר הינו די אינטואיטיבי. בשלב הראשון הסוכן (agent) מאומן לשחק משחק דום בעצמו על דאטהסט של המשחקים ששוחקו על ידי בני אדם. כלומר בהינתן כמה ממצבי המשחק (פריימים) והפעולות האחרונות (ירי, תנועה, פגיעה וכדומה) מטרת הסוכן היא חיזוי הפעולתו הבאה. זה נעשה באמצעות טכניקות RL די סטנדרטיות כאשר פונקציית ה-reward נבחרה בצורה הגיונית בהתאם ללוגיקת המשחק (כלומר פגיעה או מוות של הסוכן מקבלות תגמול שלישי ואילו פגיעה באויב, איסוף נשק וכדומה מקבלים תגמול חיובי).

אחרי שהסוכן למד לשחק דום, מגנרטים כמות מאוד גדולה של משחקים דום עם הסוכן. כלומר הסוכן משחק במשחק אמיתי כמו אחד האדם. לאחר מכן מאמנים מודל דיפוזיה לחזות את הפריים הבא בהינתן הפריימים הפעולות הקודמות והנוכחית.

האימון מתבצע בצורה די סטנדרטית: מודל דיפוזיה מקבל כקלט את הפעולות הקודמות אחרי האנקדור (שמאומן גם כן) ובנוסף את הפריימים הקודמים מוזנים למודל דיפוזיה (בצורה מורעשת לשיפור יכולת הכללה של המודל). מודל דיפוזיה שהמחברים השתמשו בו הינו לטנטי (כלומר חיזוי הרעש מתבצע במרחב הלטנטי של הפריים הנחזה). נציין כי כאן להבדיל ממודלי דיפוזיה ישנים יותר מודל הדיפוזיה במאמר מאומן לחזות את מה שנקרא "מהירות" של הפריים המורעש שהיא פונקציה של הפריים הנקי והרעש המתווסף אליו באיטרציה. רפרמטריזציה זו משפרת את איכות המודל ומאיצה התכנסותה (מוכח אמפירית כרגיל)...

מאמר מאוד מגניב...

https://arxiv.org/pdf/2408.14837

⚡🚀המאמר היומי של מייק 02.09.24: ⚡🚀

Transfusion: Predict the Next Token and Diffuse Images with One Multi-Modal Mode

היום נסקור מאמר על מודל מולטימודלי בצורה די מעניינת. המודל שאימנו במאמר יודע לגנרט גם תמונות וגם דאטה טקסטואלי ומהווה שילוב של מודל דיפוזיה ומודל שפה.

הייחודיות של המודל הזה מתבטאת בכך שהיא מגנרטת גם את הדאטה הטקסטואלי וגם הדאטה הויזואלי בצורה שאנו מגנרטים טקסטים, כלומר טוקן אחרי טוקן (עבור תמונה זה למעשה טוקן ויזואלי או ייצוג של פאץ'). כלומר אם אנו צריכים לגנרט תמונה יחד עם תיאורה המלא המודל יגנרט את התיאור טוקן ואחרי טוקן (next token prediction או NTP) ואחרי שיסיים יגנרט את התמונה טוקן אחרי טוקן (בצורת NTP גם כן). זה די נחמד האמת.

המודל שהמאמר אימון מכיל 7 מיליארד פרמטרים שזה די צנוע למודלי שפה וגודל די סטנדרטי למודלי דיפוזיה גנרטיביים (המודל הגדול של stable diffusion מכיל בערך 8B פרמטרים). אבל כאן יש לנו מודל המשלב את שתי היכולות האלו (גנרוט תמונות וגנרוט טקסטים) באיכות די גבוהה.

אבל אין מאמנים את המודל הזה? בגדול בהינתן קלט שהוא ערבוב של תמונה וטקסט (למשל תמונה מעורבבת עם טקסט). עם הטקסט הכל פשוט, מזינים אותו טוקן אחרי טוקן. לפני כל תמונה מכניסים טוקן BOI המסמן את תחילת התמונה וכאשר כל הטוקנים הויזואליים של התמונה הוזנו מכניסים טוקן EOI לסימון סיום הזנת התמונה. כאמור טוקנים של תמונה זה טוקנים ויזואליים המהווים ייצוגים של פאצ'ים לאחר האנקודר (של VAE).

איך מאמנים את החיה הזו? לטקסט זה די ברור - מאמנים את המודל לחזות טוקן טוקן כמו ב-LLM עבור מילון טוקנים נתון. עבור התמונה מחלקים את התמונה לפאצים, מעבירים כל פאץ דרך האנקודר של VAE ומזינים את התוצאה כטוקן. הייצוגים של הטוקנים הויזואלים מועברים דרך שכבה לינארית או unet להורדת מימד. במהלך האימון לומדים להסיר רעש מהגרסאות המורעשות של ייצוגי הטוקנים הויזואליים.

בגנרוט המודל יוצר את התמונה פאץ' פאץ' מהרעש (אחרי הסרת הרעש וקטור הייצוג מוזן לדקודר של VAE כדי לחזות את הפאץ' עצמו). לאחרונה השיטה הזו ליצירת תמונה לא פופולרית במיוחד - רוב השיטות יוצרות את התמונה המלאה (מהייצוג הלטנטי שלה). וכמובן כל הטוקנים האלו מוזנים לטרנספורמר אחד גדול!

מאמר מעניין ומומלץ לקריאה!

https://arxiv.org/pdf/2408.11039

⚡🚀המאמר היומי של מייק 03.09.24: 🚀⚡

Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling

אחת הדרכים הדי מפתיעות לשיפור יכולות reasoning של מודלי שפה היא שיפור עצמי או self-improvement. בגדול עבור דאטהסט של שאלות ותשובות אנו מבקשים ממודל שפה לענות על התשובה ולספק הסבר. לאחר מכן מפלטרים את השרשראות reasoning שלא התכנסו לתשובה הרצויה. לאחר הפלטור מבצעים פיינטיון של המודל על הדאטהסט המפולטר. וכאמור באופן די מפתיע (לפחות אותי) הדבר אכן מוביל לשיפור יכולות reasoning של מודל שפה.

ואם יש בידינו מודל יותר חזק אז ניתן לבנות את הדאטהסט הזה באמצעותו ולעשות את הפיינטיון על הדאטה הנוצר באמצעותו בצורה דומה.

אולם המאמר שואל שאלה די מעניינת: מה עדיף (מבחינת הביצועים), ליצור יחסית מעט דאטה עם מודל גדול וחזק או ליצור יחסית הרבה דאטה עם מודל קטן וחלש יותר. הרי יצירת דאטה עם מודל חזק היא יקרה יותר (מבחינת כמות ה-FLOPS הכוללת הנדרשת לכך) אבל מצד שני הדאטה שהוא יוצר הוא יותר איכותי.
המחברים מציעים לבצע את ההשוואה של "תפוזים לתפוזים" - כלומר לקחת את הדאטה הנוצר עם מודל חזק ומודל חזק תחת אותו תקציב של FLOPS ולהשוות מה מהם מוביל לביצועים טובים יותר של המודל שעובר פיינטיון על הדאטה הזה.

ויש תוצאות די מעניינות במאמר..

.

https://arxiv.org/pdf/2408.16737

⚡🚀המאמר היומי של מייק 04.09.24: ⚡🚀
Flexora: Flexible Low Rank Adaptation for Large Language Models

המאמר הזה נסקר קודם כל בגלל שהוא למעשה מימוש של רעיון שחשבתי עליו והוא גם רשום לי בבקלוג (שהוא באורך די אינסופי). הרעיון הוא למעשה שיטה לבחירה (לפעמים קוראים לזה אופטימיזציה) של ההייפרפרמטרים של LoRA (סוג של).

כמו שאתם בטח זוכרים LoRA היא משפחה (די גדולה שממשיכה לגדול) של שיטות מהמשפחה (גדולה עוד יותר) של שיטות חסכוניות פיינטיון של מודלי שפה ענקיים (או PEFT - Parameter Efficient Fine-Tuning). ב-LoRA אנו מאמנים תוספת של משקלים לכל שכבה במקום לאמן את כל המשקלים במודל. כל תוספת כזו היא מטריצה בעלת רנק נמוך כלומר אפקטיבית מכילה מעט פרמטרים מאשר מטריצת המשקלים של השכבה.

פרקטית כל תוספת היא מכפלה של שתי מטריצות בעלות רנק נמוך (מלבניות) וככל הרנק נמוך יותר יש לנו פחות פרמטרים לאפטם במהלך פיינטיון. הבחירה של הרנק של מטריצות התוספות הנדרשת למקסום ביצועים איננה בעיה פשוטה ויש מספר מאמרים שדנים בנושא הזה (בד"כ עד רנק מסוים הביצועים משתפרים ומנקודה מסוימת מתחיל אוורפיט).

המאמר (וגם אני) חשבו על דרך אחרת של אופטימיזציה של LoRA. המחברים שואלים שאלה פשוטה - למה בנוסף לאימון של מטריצות התוספות לא נאמן את ה-importance שלה בכל שכבה. ה-importance במקרה הזה היא המקדם המכפיל את מטריצת התוספות לפני הוספתה מטריצת המשקלות המקורית במודל (שנותרת קבועה במהלך פיינטיון). האלגוריתם המוצע עושה כמה איטרציות של משקלי ה-importance לעדכון אחד של משקלות התוספות.

האמת שהרעיון שלי הכיל עוד שלב של pruning. כלומר אחרי מספר של איטרציות אימון מתחילים לאפס ומפסיקים לאמן מטריצות התופסות עם importances נמוכים מאיזה סף. כנראה שאצטרך לבדוק את זה לבדי :)

⚡🚀המאמר היומי של מייק 05-06.09.24: ⚡🚀
EAGLE: Speculative Sampling Requires Rethinking Feature Uncertainty
EAGLE-2: Faster Inference of Language Models with Dynamic Draft Trees

חדי זכרון מביניכם אולי שמו לב כי לא פרסמתי סקירה יומית אתמול (בד"כ אני לא מפרסם סקירה בימי ראשון שבהם אני מקליט פודקאסט). אתמול היה לא חמישי ולא פרסמתי סקירה כי הכנתי לכם סקירה כפולה להיום. היום נסקור שני מאמרים שהשני מהם הוא שכלול של הראשון.

שני המאמר הם בנושא של speculative decoding או SpDe (הקיצור הומצא על ידיי). SpDe זו דרך להאיץ דגימה (גנרוט טקסט) ממודלי שפה. כמו שאתם זוכרים הגנרוט ממודלי שפה מתבצע באופן אוטורגרסיבי כמו טוקן לאחר טוקן. כמובן שזה יכול להיות איטי בטח עבור מודלים עצומים בעלי מאות מיליארדי פרמטרים.

האם ניתן להאיץ את תהליך הגנרוט - התשובה היא כן ובשנתיים האחרונות נעשה מחקר מאוד רציני בנושא והוצעו מספר שיטות שבאמצעותם ניתן להגיע לקצב דגימה גבוה יותר. SpDe היא משפחת שיטות להאצת קצב גנרוט באמצעות שימוש במודל קל (וחלש יותר) בנוסף למודל היעד (שאותו אנחנו מעוניינים להאיץ כאמור).

שיטות SpDe מבוססות על אובזרבציה כי מהלך הגנרוט טוקן אחרי טוקן צוואר בקבוק הוא העברת הדאטה מהזיכרון SRAM המהיר (אך קטן) של יחידת החישוב של GPU לבין זיכרון DRAM הגדול אך איטי יותר. ולא החישוב עצמו. נובע מכך שניתן לנצל את צוואר הבקבוק הזה ולבצע יותר חישובים (עבור יותר טוקנים) בזמן שהדאטה מטייל בין SRAM ל-DRAM.

הבעיה לממש את הגישה הזו בצורה נאיבית נובעת מאופן אוטורגרסיבי של הגנרוט ממודל שפה שלא מאפשר לבצע את החישובים עבור חיזוי של יותר טוקנים באותו הזמן. שיטות SpDe עוקפות את המכשול הזה על ידי הוספת מודל קטן ומהיר יותר שיאפשר למודל הגדול לחזות כמה טוקנים באותו הזמן.

איך זה עובד? אנו חוזים כמה טוקנים עם המודל הקטן L_s ולאחר מכן "מתקנים" את החיזוי עבור הטוקנים שנדגמו עם המודל הגדול כאשר ה"תיקון" עבור כל הטוקנים שנחזו על יד L_s מתבצע באותו הזמן. כלומר L_s חוזה הסתברויות עבור k טוקנים רצופים (בהינתן הטקסט שכבר גונרט), הם מוזנים (יחד עם הקשר) למודל הגדול L_b והוא חוזה את ההסתברויות p_i עבור k טוקנים אלו באותו הזמן.

לאחר מהם מבצעים משהו דומה למה שעושים ב-rejection sampling ו"מתקנים" הסתבריות אלו כך שיתאימו להתפלגות של המודל הגדול L_l. לאחר מכן יש שלב של rejection שבו אנו מחליטים האם אנחנו מקבלים או לא מקבלים את הטוקנים שנדגמו על יד המודל הקטן L_s. זה מתבצע טוקן טוקן (חישוב מהיר מאוד) וכל הטוקנים שבאים לפני הטוקן הראשון t שקיבל reject מתקבלים (נכנסים לטקסט המגונרט) ואיטרציית דגימה חדשה מתחילה מ-t.

ניתן להוכיח שדגימה כזו היא בעלת אותה ההתפלגות של הטוקנים כמו מודל היעד L_l. יש כאן כמה שאלות חשובות על איך לבחור מודל קטן L_s, בין כמה טוקנים לדגום איתו כל פעם במטרה למקסם את קצב הדגימה. עכשיו השאלה האם ניתן לבחור מודל L_s כך שהוא גם יהיה מהיר מאוד וגם איכותי מספיק כך שמספר הטוקנים שנדגמו איתו יתקבלו לרוב על ידי L_l. מודל כזה עתיד להגביר את מהירות הדגימה האפקטיבית מ- L_l.

זה בדיוק מה שהמאמר Eagle מציע. הוא מציע לקחת מודל קטן ולאמן אותו להיות L_s, כלומר לתפור אותו למשימה שהוא מיועד. בשביל זה עבור מודל L_l נתון לוקחים מודל טרנספורמר רדוד ומאמנים אותו לחזות לא רק את הטוקן הבא (ההסתברות שלו) אלא גם הייצוג בשכבה אחרונה של L_l לפני שכבת החיזוי (כלומר יש כאן בעיית רגרסיה). החיזוי מתבצע בהינתן הטוקנים הקודמים (הייצוגים שלהם) וגם הייצוג מהשכבה האחרונה של הטוקנים הקודמים. מכיוון שהמודל L_s הוא קטן ומהיר אנו חוזים איתו כמה סדרות טוקנים (עבור הקשר נתון) על ידי בחירה של כמה טוקנים(אך מספר קבוע כל פעם, נגיד 3) בעלי הסתברות הגבוה ביותר כל פעם. כלומר להקשר נתון חוזים כמה המשכים עבורו - בונים סוג של עץ חיזוי עבור הטוקנים. ככה יותר טוקנים רצופים עשויים לא לקבל reject מ- L_l אחר כך.

לאחר שאימון L_s הסתיים לוקחים אותו ומבצעים אינפרנס בצורה די דומה ל-SpDe עם שכלול של מנגנון ה-reject.

מה בעצם Eagle2 משכלל את מנגנון בניית עץ החיזוים על ידי L_s על ידי בחירת סדרות בעלי הסתברות כוללת מקסימלית. סדרות השונות עם Eagle2 יכולות להיות בעלות אורך שונה כמובן (הכל מסתמך על ההסתברות הכוללת של הסדרה). ככה נוצרות סדרות בעלות פוטנציאל גבוה יותר להתקבל על ידי L_l.

היה ארוך - מקווה שלא איבדתי אותכם....

https://arxiv.org/pdf/2401.15077
https://arxiv.org/pdf/2406.16858

⚡🚀המאמר היומי של מייק 07.09.24: 🚀⚡
ReMamba: Equip Mamba with Effective Long-Sequence Modeling

סוקר את המאמר הזה משתי סיבות. קודם כל הוא קשור לממבה. הסיבה השני היא זה שהתבקשתי לסקור אותו. ואוקיי, המאמר לצערי לא חידש לי הרבה ולדעתי לא נזכור אותו בעוד כמה חודשים.

אתם זוכרים את State Space Models או SSM בהקשר של למודלים עמוקים? SSM הוא ארכיטקטורה יחסית חדשה עבור רשתות לעיבוד דאטה סדרתי (שפה טבעית וגם תמונות). השוס הגדול ב-SSM היא שהם מאוד מהירים גם באימון וגם באינפרנס עקב כך שניתן לייצג אותם בתור רשת קונבולוציה וגם במודל רשת recurrent. הגמישות הזו כמובן גובה מאתנו מחיר בדמות חוסר expressiveness (יכולת למדל חוקיות מורכבות) של ארכיטקטורה הזו עקב העובדה המעברים בין המצבים החבויים הם לינאריים וקבועים לכל איברי הסדרה (מכאן בא הדואליות בייצוג).

ארכיטקטורת ממבה מחזירה לנו קצת מה-expressiveness בכך שהופכת את המעברים בין המצבים החבויים לתלוי במצב החבוי אך משאיר אותם לינאריים. זה עוזר אבל עדיין ממבה מתקשה במשימות reasoning מורכבות עקב מחסור ב-expressiveness. ייתכן שאחת הסיבות לאי הצלחה זו היא חוסר יכולת של ארכיטקטורת ממבה לדחוס את המידע הרלוונטי למשימה (לגיטימי אבל כמובן יש עוד סיבות לכך).

המחברים מציעים לדחוס את ייצוגיהם של תת סדרות של טוקנים. נניח שיש לנו L טוקנים בהקשר ואנו רוצים "לדחוס" את טוקנים שייצוגיהם דומה לזה של הטוקן L. כלומר מחשבים את הדמיון בין תת-סדרה רציפה נתונה של טוקנים (הייפרפרמטר) ודוחסים את הייצוגים של הטוקנים בתת-סדרה זו לפחות טוקנים (הייפרפרמטר גם כן). כלומר במקום הייצוגים של M טוקנים בתת סדרה נקבל ייצוגים של K טוקנים אחרי הדחיסה.הדמיון מחושב דרך דמיון קוסיין (עם כל מיני שכבות לינאריות).

הם מראים שזה עובד - לי זה מריח קצת אוברפיט וגם קושי באופטימיזציה של ההייפרפרמטרים....

⚡🚀המאמר היומי של מייק 08.09.24: 🚀⚡
DO TRANSFORMER WORLD MODELS GIVE BETTER POLICY GRADIENTS?
לא הייתי אמור לכתוב סקירה היום אך הקלטת הפודקאסט שלנו התבטלה והתפנה לי קצת זמן אז אסקור מאמר שכבר נמצא כמה זמן אצלי במגירה. המאמר בנושא למידה עם חיזוקים (RL) וטרנספורמרים אז לכאורה זה נשמע מאמר די נחמד.

המאמר מדבר על שיטה לשיפור של למידה פוליסי בבעיות של RL בבעיות שיש לנו גישה ישירה לדינמיקה של הסביבה (כלומר אנו לא יכולים לאסוף עליה דאטה רלוונטי המאפיין את פיצ'רים המהותיים שלו קרי non-observable). בגדול המטרה שלנו בלמידת פוליסי היא לחזות את הפעולה (action) האופטימלי בהינתן המצב s של הסביבה והפעולה האחרונה s. אופטימלי כאן משמעותו מקסום של התגמול (reward) הכולל המתקבל במהלך אפיזודה. המודל שחוזה את הפעולה הזו הוא למעשה מממש את הפוליסי שלנו.

אבל מה לעשות אם אין לנו גישה ישירה לסביבה? במקרה הזה אנו יכולים לאמן מודל שהוא חוזה לנו את המצב הבא s בהינתן המצב הקודם(כלומר ייצוגו) והפעולה האחרונה(עם הנחת המרקוביות) או בהינתן N ייצוגים של המצבים האחרונים והפעולה האחרונה. זה למעשה נקרא world model (לדעתי יחד עם מודלים המשערכים את התגמול הצפוי למצב נתון - value function אבל זה פחות חשוב כרגע).

איך המודל הזה מאומן? מאינטראקציה עם הסביבה - הסוכן מבצע פעולות בסביבה ואנו מעדכנים את ה-world model שלנו בהתבסס על משוואות Bellman). שימו לב אם או ללא הנחת מקרוביות אנחנו משערכים את הייצוג של המצב ה"עולם" הבא בהינתן המצב(-ים) הקודמים. המאמר טוען שזה יוצר גרדיאנטים לא יציבים ושונות גבוהה עקב שימוש ישיר בשערוך של המצבים הקודמים לשערוך של המצב הבא.

הם מציע לשערך את המצב הבא מהפעולה ולא מייצוגי המצבים שטענתם "הופך את הגרדיאנטים במודל העולם לפחות מעגליים" וזה תורם ליציבות השערוך. יש גם קצת הוכחות במאמר (סוג של) של הטענה הזו. המאמר מראה אם יש לנו מקרוביות (התלות של המצב הבא היא רק במצב האחרון) השיטה המוצעת עובדת כמו RNN מבחינת הגרדיאנטים. בתחושה זה נשמע לי די טבעי (אשמח אם מישהו ירחיב על זה). במקרה שאין לנו מרקוביות הטענה לביצועים טובים יותר של השיטה המוצעת.

לא ראיתי אזכור משמעותי מדי של הטרנספורמרים במאמר (תקנו אותי אם אני טועה).

⚡🚀המאמר היומי של מייק 09.09.24: 🚀⚡

MemLong: Memory-Augmented Retrieval for Long Text Modeling

אחד המאמרים ראשוניים בנושא Retrieval Augmented Generation או RAG שאני סוקר. הנושא צובר תאוצה רצינית בזמן האחרון והגיע הזמן להשלים את הפערים (גם בידע וגם בסקירות).

RAG זה בעצם דרך להתגבר על כך שלמרות כל ההישגים בתחום אפילו מודלי שפה החדשים ביותר מתקשים לעבוד עם אורך הקשר מאוד ארוך. מה בעצם קורה כאן? נניח שיש לנו אוסף נתונים D ואנחנו רוצים שמודל השפה שלנו יענה על שאלות על D תוך כדי שילוב יכולות שהוא צבר במהלך האימון לפני זה.

אחת הדרכים היא לעשות למודל שפה פיינטיון על D אולם זה עלול להיות בעייתי כי המודל יכול לשכוח חלק מהדברים שידע קודם וגם יתקשה ללמוד את כל מה שיש ב-D בצורה יעילה (פתיר כמובן אבל קשה). הדרך השנייה כי להוסיף את D לכל שאלת המשתמש (כחלק מפרומפט) אבל זה גם בעייתי ל- D גדולים עקב אי יכולת של מודלי שפה להתמודד עם אורך הקשר גדול מאוד.

דרך נוספת היא לעשות RAG (אפשר לשלב אותו עם פיינטיון קליל - ראיתי מאמר שעושה את זה) כלומר לכל שאילתה של משתמש לבחור את המידע מ- D (כמה צ'אנקים) הכי רלוונטיים לשאלה והוסיף אותם לפרומפט. הבעיה בגישה הזו היא מטריקה לבחירת הצ'אנקים הרלוונטיים ביותר לשאלה. בד"כ זה נעשה על סמך המרחק קוסיין בין ייצוג השאלה לייצוגי הצ'אנקים (כלומר אמבדינגס). כלומר בוחרים כמה צ'אנקים הקרובים ביותר לשאלה מבחינת מרחק זה.

גישה זו עלולה להיות בעייתית גם כן כי לא תמיד מרחק קוסיין בין הייצוגים משקף את רלוונטיות של צ'אנק לשאלה. המאמר שנסקור היום מציע בנוסף לצ'אנקים לתת ל-RAG את הזכרון המאחסן את הייצוגים של השאלות האחרונות(או/ו השכיחות) ובנוסף לכל שאלה מחזיק סוג של KV-cache עבור השאלה הזו (מניחים שיש לנו דאטהסט המכיל שאולות ותשובות וגם אוסף נתונים D). אז KV-cache הזה הייצוג של וקטורי Key and Value עבור שכבה מסוימת (לקראת הסוף המודל וזה אחד הייפרפרמטרים של השיטה). KV-cache יעזור לנו לבנות תשובה בצורה טובה יותר.

אז איך כל העסק הזה עובד? במהלך האימון אנו לוקחים שאלה ותשובה מהדאטהסט של שאלות ותשובות ובאמצעותו בונים את ה-KV cache של המודל כי אנחנו יודעים מה הצ'אנקים הרלוונטיים ביותר לכל שאלה. הרי לכל צ'אנק אנו שומרים את ה-KV שלו (מחושב כאשר הצאנק מוזן למודל יחד עם השאלה).

עכשיו אנו רוצים לאמן את הרשת לנצל את ה-KV caches האלו בצורה יעילה. בשביל כך באימון לכל שאלה לוקחים את צ'אנקים הכי קרובים אליה (מבחינת האמבדינג), לוקחים את ה- KV cache עבורים ומאמנים את השכבות האחרונות של המודל להוציא את התשובה הנכונה. כלומר לומדים איך לשלב את התוצאה (attention maps) מהשכבות התחתונות יחד עם ה-KV cache שצברנו מהזכרון (יש עוד איזה שכבה לינארית מאומנת בנוסף). עדכון הזכרון מתבצע בצורה די סטנדרטית (LRU ובנוסף השכיחות נלקחת בחשבון).

האינפרנס עובד באותה הצורה פחות או יותר. בגדול המאמר מציע שיטה לשדרוג RAG באמצעות ניצול המצב של KV-cache במהלך האימון. די נחמד מודה...

⚡🚀המאמר היומי של מייק 10.09.24: 🚀⚡

Can LLMs Generate Novel Research Ideas? A Large-Scale Human Study with 100+ NLP Researchers

האם מודלי שפה יכולים לייצר רעיונות מחקר חדשניים? 🤔 מחקר חדש מעורר גלים. ראינו לאחרונה התלהבות רבה סביב השימוש ב-LLMs לגילויים מדעיים. אך האם הם באמת מסוגלים להגיע לרעיונות חדשניים ברמת ראויה לחוקר במוסד אקדמי או בתעשיה?

מחברי המאמר תכננו ניסוי כדי לבדוק את הסיפור הזה. הם שכרו מעל 100 מומחי עיבוד שפה טבעית לכתוב רעיונות מחקר ולבחון רעיונות שנוצרו על ידי בני אדם ו-LLMs (בעיוור כלומר הבודקים לא ידעו מה מקורה של הרעיון שהם בודקים).

מתברר כי הרעיונות של ה-LLM נשפטו (באופן לא מפתיע קלוד נבחר למשימה זו) כחדשניים יותר מרעיונות מומחים אנושיים (עם מובהקות סטטיסטית), אך דורגו נמוך יותר בהיתכנות.

המחברים מציינים את מהחוזקות הבאות של רעיונות ה-LLM:

- הצעת מכילה שילובים ייחודיים של טכניקות מדומיינים שונים
- חקירת תחומים שלא נחקרו מספיק
- יצירת ניסויי מחשבה יצירתיים ומקוריים

עם זאת, היו להם גם כמה נקודות בעיותיות:

- חוסר פירוט מספק בנוגע ליישום
- שימוש לא נכון במאגרי נתונים

- החמצת בייסליינים (לא מפתיע כלל)
- הנחות לא מציאותיות

לעומת זאת, רעיונות אנושיים נטו להיות מעוגנים יותר במחקר קיים ובשיקולים מעשיים, אך לעתים קרובות היו פחות חדשניים, ובנו באופן הדרגתי על אינטואיציות ותוצאות ידועות.

המחברים מציינים שהחוקרים מכירים בקושי לשפוט חדשנות, אפילו עבור מומחים. כצעד הבא, הם הציעו לתת לחוקרים לממש את הרעיונות הללו, כדי לראות אם דירוגי החדשנות וההיתכנות מתורגמים להבדלים משמעותיים במציאות.

⚡🚀המאמר היומי של מייק 12.09.24: 🚀⚡

https://openai.com/index/learning-to-reason-with-llms/

היום במקום הסקירה אשתף איתכם את מחשבותיי על המודל החדש של openai שקיבל שם o1. אני בדרך כלל נמנע מלהגיב ולכתוב פוסטים על כל מודל חדש שמנצח את כל ה-benchmarks בעולם אבל הפעם אחרוג ממנהגי. ולא מהסיבה שמהמודל הזה השאיר אבק לרוב ה-benchmarks אלא בגלל שאני זיהיתי כאן שינוי מסוים בפרדיגמה בעולם ה-llms.

השינוי בפרדיגמה בא בדמות של שינוי היחס בכמות הקומפיט המוקדש ללמידה ולהסקה (אינפרנס). אנחנו רגילים למודל שמצריכים כמות אדירה של קומפיוט במהלך הלמידה (אימון מקדים, SFT, יישור המודל וכדומה) כאשר האינפרנס הוא די זול (כמובן יחסית לאימון כי גם בהסקה יש עלויות די גבוהות בשל עצמם). O1 לעומות זאת מאתגר את ההנחה הזו ושואל את השאלה: האם זה אופטימלי? אולי אנו צריכים לאמן את המודל שלנו פחות ולהשקיע יותר קומפיט בהסקה.

לפני כמה זמן סקרתי מאמר שדי שינה (או לכל הפחות רענן) את תפיסתי בעניין זה (Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters). המאמר הזה היה של deepmind אולם הייתה לי תחושה שהם לא היחידים שהגיעו לתובנה הדי לא טריוויאלית הזה.

בעקרון הכל מסתכם לשתי הנקודות הבאות:

- אולי אתה לא צריך מודל שפה ענק להסקה. חלק ניכר מהפרמטרים כנראה ממשמשים לאחסון עובדות, כדי שהמודל לא ידבר שטויות לשאלות לידע כללי (כמו מתי נולד מוצרט). לדעתי ניתן להפריד בין הסקה לידע, כלומר אפשר להסתפק ב"ליבה להסקה" קטנה שיודעת איך להשתמש בכלים כמו וולפרם, בראוזר ובודק קוד כלומר המשימות הדורשות סוג של ידע עובדתי (ידע בשפת תכנות). ככה ניתן להפחית את כמות החישוב המוקדשת לאימון המוקדם.

- כמות משמעותית של קומפיט מועברת להסקה בזמן הרצת המודל ולא לאימון המודל. ניתן לחשוב על מודלי שפה בתור סימולטורים מבוססי טקסט. על ידי הרצת תרחישים ואסטרטגיות רבות (גנרוט טקסט), המודל יגיע בסופו של דבר לפתרונות reasoning טובים. התהליך בחירת הפתרון נראה די דומה לבעיות שנחקרו היטב כמו חיפוש העץ של מונטה קרלו (MCTS) ב-AlphaGo.

כמובן שאם יש שימוש בטכניקות כמו MCTS אנו צריכים את פונקציית ה-reward. בניית פונקצייה כזו היא לא טריוויאלית כאן כי אין לנו דרך טובה (אלא אם כן יש לנו דאטהסט reasoning מגוון ועצום שניתן לאמן עליו מודל כזה) לשערך את איכות ה-reasoning. כמובן שניתן לנצל מודלי שפה אחרים, בדיקות עצמיות על ידי מודלי שפה

וכדומה אבל עדיין לא ברור ב-100% איך לעשות את זה (ד״א אני בכלל לא בטוח שהם השתמשו ב-mcts). אולי הם פיתחו שיטה מגניבה לעקוף את ה-reward כמו שנעשה ב-dpo וב-orpo שעשו זאת עבור ppo -אין לדעת.

בקיצור מחכה לדוח הטכני שבתקווה ישפוך אור על הסיפור הזה (גם בזה אני לא בטוח בכלל)....

⚡🚀המאמר היומי של מייק 13.09.24: 🚀⚡
LLMs Will Always Hallucinate, We Need to Live With This

טוב, המאמר הזה הוא פשוט קליקבייט לדעתי ואז גיליתי שם משפט גדל אז בכלל. הוא מציג ניתוח מעמיק של הזיות (hallucinations) ב-LLMs וטוען כי הזיות אלו הן תכונה אינהרנטית בלתי נמנעת של המבנה המתמטי/ארכיטקטוני ואופן החישובי שלהם (אולי o1 החדש יאתגר את זה טיפה).

כמה נקודות עיקריות מהמאמר:

1. **הזיות כבלתי נמנעות**: הזיות אינן רק טעויות אלא תוצאה בלתי נמנעת של הארכיטקטורה וההיגיון השולטים במודלים גדולים לשפה. הן נוצרות כאשר המודלים מנסים להשלים פערים בידע או לייצר מידע סביר אך שגוי על סמך נתונים חסרים או מעורפלים.
2. **חוסר שלמות של נתוני האימון**: המאמר מדגיש כי אף מאגר נתונים אינו שלם ב-100%, ולכן LLMs תמיד יתקלו במצבים שבהם עליהם להסיק או להמציא מידע שלא קיים במאגר הנתונים (המוחבא במשקלים שלו או במערכת נתונים חיצונית).
3. **4 סוגים עיקריים של הזיות**:
   - **אי דיוק עובדתי**: המודל עלול לגנרט מידע עובדתי שגוי בשל ״אופן שליפה שגוי״ של מידע ממאגרי הידע שלו.
   - **אי הבנה**: המודל נכשל בהבנת קלט המשתמש, ונותן תשובות שגויות.
   - **מחט בערימת שחת (needle in a haystack)**: קושי בשליפת מידע ספציפי ממאגר נתונים (במשקלים שלו או במערכת נתונים חיצונית), מה שלעתים מוביל למידע מעורב או חלקי.
   - **המצאות**: LLMs לעיתים ממציאים מידע כאשר הקלט אינו מוכר להם מהטריין סט ולא תואם לשום עובדה ידועה במאגר הנתונים שלהם.

4. **״בלתי מוכרעות״**: המחברים משתמשים במשפטי אי שלמות של גדל ובתיאוריית חישוביות, ומדגימים שבעיות מסוימות, כגון שליפת מידע עובדתי מדויק (וסיווג כוונת המשתמש (intent classification), אינן ניתנות להכרעה. המשמעות היא שאין אלגוריתם שיכול למנוע לחלוטין הזיות.

5. **LLMs לא מסוגלים לנבא מתי הם ייעצרו**: המחברים טוענים כי LLMs לא מסוגלים לחזות מתי ייעצר הגנרוט (מזכיר הבעיה הידועה של עצירת מכונה בתיאוריה חישובית). הם טוענים שנובע מכך כי המודלים אלה אינם מסוגלים לשלוט או לצפות במדויק איזה תוכן הם ייצרו, מה שמעלה סיכוי להזיות.
6. **הוכחה שהזיות אינן ניתנות לביטול**: המאמר מראה (יש הוכחה) שגם כוונון מושלם או מנגנוני בדיקת עובדות לא יכולים לבטל לחלוטין הזיות. זאת משום שמאגר הנתונים תמיד יהיה חסר או בלתי מספיק, ומודלים גדולים לשפה חייבים לייצר פלט שאינו ניתן לאימות או סותר.

7. **השפעה של RAG**: למרות שטכניקות כמו הפקת מידע מוגברת נועדו לשפר את הדיוק העובדתי באמצעות שליפת מידע חיצוני, הן עדיין מסתמכות על פונקציות שליפה לא מושלמות, מה שמוביל לתוצאות חלקיות או מעורבות.

8. **תפקיד קידוד מיקומי (positional encoding או PE)**: המאמר נוגע בטכניקות PE מתקדמות כמו RoPE וכיצד הן משפרות את ביצועי המודלים באמצעות שילוב מיקומים מוחלטים ויחסיים. עם זאת, טכניקות אלו עדיין לא פותרות את בעיית ההזיות.

9. **הזיות מבניות**: המחברים מציגים את המושג "הזיות מבניות", ומדגישים שהן תוצאה בלתי נמנעת של הארכיטקטורה של LLMs ולכן אינן ניתנות למניעה, גם לא באמצעות שיפורים באימון או כוונון.

10. **השוואה למודלים אחרים**: המאמר משווה בין מודלים לשפה למודלים אחרים כמו ממבה, KANs אך מסיק שהמגבלות המובילות להזיות קיימות בכל הארכיטקטורות.

11. **מכונת טיורינג ו-LLMs**: מודלי שפה מוצגים כשווים למכונות טיורינג אוניברסליות, מה שאומר שהם יורשים את אותן מגבלות חישוביות, כולל בעיות בלתי-מוכרעות כמו בעצירה.

12. **השלכות לעיצוב עתידי של LLMs**: המאמר מציע שהפיתוחים העתידיים של LLMs צריכים להתמקד בניהול והפחתת הזיות במקום לנסות לבטל אותן, שכן הדבר בלתי אפשרי מתמטית וחישובית.

⚡🚀המאמר היומי של מייק 14.09.24: 🚀⚡
Beyond Neural Scaling Laws: Beating Power Law Scaling via Data Pruning

חוקי סקיילינג זה נושא מאוד מעניין אך לצערי אני מתקשה למצוא מאמרים באמת שווים סקירה (שמכילים מעבר לניסויים אינסופיים עם הייפרפרמטרים שונים). הפעם התמזל מזלי ונתקלתי במאמר הלא חדש הזה שהוא נראה די שווה.

המאמר מציע סוג חדש חוקי סקיילינג בנוגע ל- Data Pruning (צמצום דאטה או DP). המחברים מספקים ראיות תיאורטיות (זו הסיבה שאני סוקר אותו) ואמפיריות לכך שצמצום פיסות דאטה מיותרות או פחות אינפורמטיביות יכול לשבור את חוקי הסקליינג המסורתיים, ולהשיג הפחתה מהירה יותר בשגיאה תוך שימוש בפחות משאבים.

רקע: חוקי הסקיילינג של רשתות נוירונים מתארים כיצד השגיאה(טסט) יורדת עם הגדלת גודל המודל, כמות הדאטה או כמות הקומפיוט, בהתאם לחוק חזקה (Power Law). עם זאת, סקלינג זה אינו יעיל, שכן שיפור בביצועים דורש כמות דאטה/משאבים אקספוננציאלית. המחברים שואלים האם ניתן להשיג סקלינג טוב יותר מחוק חזקה על ידי בחירה מושכלת של דאטה.

התמצית: המחברים מפתחים מסגרת תיאורטית המבוססת הלקוחה ממכניקה סטטיסטית, תוך שימוש במודל בסגנון זיקוק מידע (מודלי סטודנט-מורה). מודל זה מתאים לבחינה תיאורטית של data pruning (זריקת נתונים) בגלל פשטותו המתמטית, תוך שמירה על תכונות הכללה (generalization) שנשמרות במודלים מורכבים יותר.

המסגרת המתמטית המוצעת מורכבת מ"מורה" שמייצר דאטה, ומודל "סטודנט" שמנסה ללמוד אותו. הרעיון המרכזי הוא "להעיף דוגמאות על על בסיס הקושי שלהן". קושי של דוגמא נמדד על פי המארג'ין(המרחק של דוגמא מגבול ההחלטה). בגדול הם הראו כי יש לשמור דוגמאות קלות (עם מארג'ינים גדולים) עבור דאטהסטים קטנים, בעוד שדוגמאות קשות יותר (עם מארג'ינים קטנים) הן אינפורמטיביות יותר דאטהסטים גדולים. המחברים מראים כי שגיאת ההכללה E_g, תלויה ביחס בין מספר דוגמאות כולל לפרמטרי המודל (alpha) ובחלק מהדאטה f שהוסר. המסקנה המרכזית היא שחיתוך אופטימלי שובר את חוק החזקה בסקיילינג, ומוביל לסקיילינג מעריכי של הפחתת השגיאת הכללה.

אז אלו דוגמאות להשאיר: כאמור עבור דאטהסטים קטנים, עדיף לשמור דוגמאות קלות כדי להימנע אוברפיט, בעוד שעבור דאטהסטים גדולים, משתלם להשאיר דוגמאות קשות כדי ללמוד גבולות החלטה עדינים יותר. יש טענה במאמר שברגע ששומרים את הדוגמאות הקשות ביותר, מתאפשר סקלינג מעריכי של הפחתת שגיאת ההכללה E_g, עבור דאטהסטים גדולים. המחברים מצאו כי הדעיכה המעריכית מחזיקה עד לנקודת שבירה קריטית, שבה הדוגמרו הנותרים כבר אינם מספקים מספיק מידע. מעבר לנקודה זו, דעיכת השגיאה מאטה ועוברת לחוק חזקה.

רווח מידע (Information gain או IG): המחברים טוענים כי בלמידה עם רשתות המידע השולי שמספקת כל דוגמא נוספת פוחת עם מספר הדוגמאות, מה שמוביל ליחס חוק חזקה בין גודל הדאטהסט להפחתת שגיאת הכללה. אולם, עם אסטרטגיית בחירה חכמה, המצב משתנה. חיתוך מסיר נתונים מיותרים או בלתי אינפורמטיביים, ומאפשר לכל דוגמה שנותרה לספק מידע ייחודי יותר על המשימה. מתמטית, תכולת המידע של דאטהסט (לסטודנט) פרופורציונלית למספר דוגמאות שנותרו, אך ניתן להאט את קצב הירידה עם בחירה מושכלת של הדוגמאות. כלומר רווח המידע לדוגמא נשאר משמעותי גם כשהדטאהסט נחתך, מה שמאפשר דעיכה מעריכית של השגיאה.

חוסר איזון בין קטגוריות: המאמר דן בכך שבחירת דוגמאות ללא התחשבות בהתפלגות קטגוריות עלול להוביל לחוסר איזון בינן יגרום לירידה בביצועי המודל. המחברים מציעים טכניקת איזון קטגוריות שמבטיחה שכל אלו יישארו מיוצגות היטב בדאטהסט החתוך.

https://arxiv.org/abs/2206.14486

⚡🚀המאמר היומי של מייק 15.09.24: 🚀⚡
Q*: Improving Multi-step Reasoning for LLMs with Deliberative Planning

אחרי סערה החשיבה בזמן האינפרנס במודל החדש של openai התחלתי לבנור בפוסטים בנושא הזה ונתקלתי במאמר הדי מפורסם הנקרא Q*. מתברר שהוא נמצא אי שם ברשימת המאמרים האינסופית שאני רוצה לסקור אך לא ב-20 הראשוניים אפילו. מכיוון שקיימות די הרבה סקירות של המאמר הזה ייתן סקירה יחסית קצרה בלי לרדת לפרטים יותר מדי.

המאמר מדבר על "תהליך החשיבה או תכנון" עבור מודלי שפה. למעשה זה סוג של CoT מנוהל על ידי פונקציית Q המשערך ערך של כל שלב במהלך "החשיבה" של המודל. כלומר עבור כל שלב ב-reasoning אנו רוצים להבין עד כמה מענה נתון של LLM יקרב אותנו לתשובה הסופית הנכונה. אתם מריחים כאן פונקציית Q ידוע מעולם למידה עם חיזוקים וזה הניחוש הנכון כאן.

כדי לפרמל את הבעיה במונח RL צריך להבין מה זה מצב (state) ופעולה (action). במקרה שלנו פעולה היא תשובה של LLM בשלב נתון של תהליך החשיבה שלו ומצב הוא סדרה של כל הפעולות עד השלב הזה כלומר כל התשובות (בסדר כרונולוגי) שהמודל נתן. והמטרה כאמור לבנות את פונקציית Q בהינתן מצב s_t ופעולה a_t נתונים בשלב t, כלומר לשערך את איכות תשובה a_t עבור התשובה הקודמות a_1, ….a_t-1. ברגע שיש בידנו את Q אנו יכולים לבנות את ההמשך האופטימלי של שרשרת החשיבה a_1, ….a_t-1. כמובן היינו רוצים פונקציית Q אופטימלית כלומר כזו שמקיימת משוואת בלמן ובעלת תכונות טובות.

אבל איך נוכל לשערך את הפונקציה הזו אם יש בידינו רק מודל עם פרמטרים נתונים שלא מותאם (ישירות) לכל הסיפור של בחירת שרשרת חשיבה אופטימלית. כלומר אין לנו פוליסי אופטימלי שאותה אנו יכולים למנף ליצירת Q אופטימלי. המאמר מזכיר 3 אפשרויות.

1) בהינתן דאטהסט נתון של שרשראות חשיבה וציונים ניתן לשערך Q אופטימלי יחד עם השערוך שלו עבור הפוליסי המוקפא שלנו (כלומר מודל שפה) בצורה alternating (שערוך של של כל אחד באמצעות השני כל פעם).
2) מריצים את הפוליסי הקיים וכל פעמים בוחרים את הפעולה (תשובה) בעלת ערך Q מקסימלי, ומשפרים את שערוכה באמצעות חישוב של התגמול הכולל (עבור כל השלבים). דרך אגב קביעת מה זה התגמול המידי במצב s_t לא נראה לי טריוויאלי
3) שימוש במודל שפה חזק אחר כדי "לחקות" את הפוליסי האופטימלי ובאמצעות הרצתו לשערך את Q האופטימלי.

כאמור ברגע שיש לנו שערוך טוב של Q האופטימלי אנו תמיד בוחרים את התשובה בעלת Q הגבוה ביותר מפול התשובות של LLM.

אז למה יש כן כוכבית בשם. האלגוריתם שהתקבל מאוד מזכיר את A* המפורסם אך זה כבר נושא לסקירה אחרת...

https://arxiv.org/pdf/2406.14283

⚡🚀המאמר היומי של מייק 16.09.24: 🚀⚡
Rethinking Benchmark and Contamination for Language Models with Rephrased Samples

חתיכת נושא זה. לאחרונה אני ניהלתי מספר שיחות עם אנשי NLP לא מעטים על הנושא הזה. מי שעוקב אחריי ברשתות החברתיות אולי שם לב כי אני בד"כ לא מתלהב ממודל שפה שניצח את כל המודלים הקיימים בכל הבנצ'מרקים. הסיבה לכך היא די טבעית ונובעת מכך שבלא מעט מקרים לא מפרסמים באופן גלוי את כל הדאטה שעליה המודל אומן.

כמובן שהחשד שלי הוא הדאטה(משימות) האימון יהיו דומות מדי לאלו שמופיעות בבנצ'מרקים האלו. כמובן אני לא בא להאשים אנשים על כך שהם מרמים בכוונה (למרות שבטח יש מקרים כאלו) אלא אני בא להגיד שזיהוי דוגמאות בדאטהסט הדומות מדי לבנצ'מרקים אינן מצליחות לפלטר את הדוגמאות האלו. והתוצאה היא מודל שהוא אוברפיט על בנצמרק כזה או אחר.

כאמור יש שיטות די בסיסיות הבודקות את הדמיון בין הדוגמאות בדאטהסט לדוגמאות בבנצ'מארק מבוססות על n-grams ועל דמיון סמנטי המחושב באמצעות מרחק בין הייצוגי של הדוגמאות בדאטהסט ובבנצ'מרק. המאמר המסוקר טוען שזה לא מספיק וצריך לעשות בדיקה נוספת לזיהוי של דוגמאות אלו. בגדול המאמר מציע בנוסף לבדיקה הסמנטית לרתום איזה LLM עוצמתי לבדיקה של דמיון דוגמאות.

בגדול מזהים K דוגמאות הכי דומות סמנטית לכל דוגמא בבנצ'מרק ואז מפעילים LLM חזק כמו GPT4 עם איזה פרומפט מתוחכם כדי לזהות את הדוגמאות הבאמת דומות. המאמר מראה כי בצורה כזו הצליחו לתפוס דוגמאות שלמרות שנראות שונה מהוות rephrasing של דוגמא מסוימת מהבנצ'מרק. ואז מעיפים את הדוגמה הזו מהדאטהסט.

המאמר טוען כי ללא שימוש בשיטה שלהם ניתן "לאמן" מודל 13B כדי ש"ינצח" את GPT4 על כל הבנצ'מרקים - נצחון לא אמיתי אמנם.

מאמר ללא יותר מדי חדשנות אך מעלה נושא מאד מעניין

https://arxiv.org/pdf/2311.04850

🚀המאמר היומי של מייק 17.09.24: ⚡🚀
STaR: Self-Taught Reasoner Bootstrapping Reasoning With Reasoning

אני ממשיך לחפור במאמרי שאולי עיצבו את הנתיב הובילו ל-o1 של openai. הפעם נברתי כה עמוק שהגעתי למאמר שיצא לפני שנתיים וחצי (בדיפ היום זה כמו 100 שנה במתמטיקה). שימו לב שהמאמר יצא עוד לפני chatgpt. המאמר הזה מציע שיטה לשיפור יכולת reasoning של מודל שפה כאשר בידנו יש דאטהסט גדול של שאלות ותשובות D ודאטהסט קטן D_R הרבה יותר (המאמר מדבר על 10 דוגמאות בלבד) המכיל בנוסף גם את שרשרת ה-reasoning.

כאשר אני מדבר על שיפור איכות ה-reasoning אני בעצם מתכוון לפיינטיון של המודל במטרה לקבל מודל חזק יותר ב-reasoning. המחברים מציעים אלגוריתם המורכב משני שלבים עיקריים. בשלב הראשון מזינים את הבאץ' של שאלות למודל שפה כאשר בנוסף לשאלות הפרומפט מכיל את דוגמאות לשרשראות ה-reasoning m מ-D_R. המודל מתבקש לבנות שרשרת reasoning לכל השאלות מבאץ' (לא מ-D_R) ולהגיע לתשובה הסופית.

את שרשראות ה-reasoning לשאלות שהצליחו להגיע לתשובה נכונה מוסיפים לסט שנקרא לו D_N. לשאלות שהמודל לא הצליח להגיע לתשובה סופית נכונה אנחנו מוסיפים רמז (במאמר זה נקרא rationalization) שעוזר למודל לבנות את שרשרת ה-reasoning. השאלות שהצליחו להגיע לתשובה הנכונה אחרי הרמז גם נוספים ל D_N. לאחר מכן מבצעים איטרציה אחת של שיטת מורד הגרדיאנט נבחרת על D_N ומעדכנים את משקלי המודל. חוזרים על השלבים האלו עד שהלוס מתייצב.

זהו זה, שיטה אינטואיטיבית ופשוטה שקיבלה כמה מאמרי השמך די כבדים שבתקווה אסקור אותם גם כן
https://arxiv.org/pdf/2203.14465

🚀המאמר היומי של מייק 19.09.24: ⚡🚀
Training Chain-of-Thought via Latent-Variable Inference

ממשיכים בקו הסקירות שהובילו (לפחות לעניות דעתי) למודל החדש (יחסית, יצא כבר לפני שבוע) של openai. במאמר הקודם שסקרתי STaR דיברנו על איך ניתן לשפר יכולת ריזונינג של מודל שפה כאשר יש בידינו דאטהסט גדול יחסית של שאלות ותשובות D ודאטהסט קטן של שאלות ותשובות עם הריזונינג. בגדול הרעיון שם היא לרתום מודל שפה לייצר ריזונינג לשאלות, להוסיף שאלות שהריזונינג שלהם הוביל לתשובה נכונה לדאטהסט הקטן ולהמשיך לאמן עד ההתכנסות.

המאמר הנוכחי שיצא בערך שנה וחצי אחריו משכלל את הגישה הזו ומציע שיטה ש"ממנפת" גם את השאלות שעבוד המודל יצר ריזונינג שלא הוביל לתשובה הנכונה. המאמר מכיל מתמטיקה די כבדה אז אנסה להעביר לכם את הרעיון הכללי יחסית בפשטות.

הרי המטרה שלנו היא לעשות פיינטיון למודל שפה כך שיכולת הריזונינג שלו תשתפר. מתמטית ניתן לתרגם את הבעיה לבעיה וריאציונית באופן הבא. אנו מעוניינים לאמן מודל שיוצר ריזונינג עבור שאלה x. מה שיש לנו זה דאטהסט של שאולות x ו-תשובות y. אז אנחנו רוצים לאמן את המודל להפיק ריזונינג z (ניתן להתייחס אלי כמו אל משתנה לטנטי) מהתפלגות בהינתן השאלה x מ-D תוך כדי ניצול של התשובה y. כלומר אנו רוצים למקסם את

הנראות (likelihood) של ההתפלגות המותנית של הריזונינג z בהינתן (עבור) שאלה x ותשובה y. במילים פשוטות אנו מאפטמים את פרמטרי המודל כך שהנראות הזו תהיה מקסימלית על D.

אולם אנו לא יכולים לעשות זאת בצורה ישירה כלומר לא ניתן לדגום את הריזונינג בהינתן שאלה x ותשובה y. הסיבה לכך היא שאנו לא רוצים לאמן מודל שמייצר ריזונינג לשאלה יחד עם התשובה (כי אנו רוצים מודל שיפתור לנו שאלות בלי לדעת את התשובה). אז המאמר הקודם בחר לנצל את תשובה y על ידי פלטור החוצה של z שהובילו לתשובות לא נכונות. לעומת זאת המאמר הזה מציע שיטה שבה אנו ממנפים גם את ה- z-ים הלא נכונים לשיפור המודל.

כאמור המאמר מנצל כמה שיטות מתמטיות די כבדות לכך ואחת מהם הוא שכלול של Markov Chain Monte Carlo כאשר ה-proposal distribution (שממנו דוגמים במטרה שזו תתכנס עם הזמן להתפלגות היעד כלומר זו של ריזונינג z בהינתן שאלה x ותשובה y) משתנה עם האיטרציה להאצת התכנסות (Markovian score climbing שהוא שכלול של Robbins-Monro לחישוב "גודל העדכון").

מה הקשר ל-MCMC אתם שואלים? אנו כל פעם דוגמים מהמודל עם המשקלים מהאיטרציה הקודמת (באץ') ומקווים שזה יתכנס להתפלגות הרצויה. המחברים מציעים לעדכן את משקלי המודל גם עבור התשובות הלא נכונות וגם הנכונות (בכיוונים שונים כמובן). ככל שהאיטרציות עוברות השיטה מעדכנת את המודל יותר עבור דוגמאות עם ריזונינג לא נכון (מוביל לתשובה לא נכונה) כי רוב השאלות כבר מקבלות ריזונינג נכון ו"פחות שווה" להתחשב הזה.

בנוסף המאמר משכלל את עדכון משקלי המודל המדובר על ידי כך שהוא שומר את הריזונינג האחרון z לכל דוגמא ומחשב את גודל (כמו קצב למידה) של עדכון משקלי המודל בהתאם. למשל העדכון עבור הריזונינג של דוגמא שהוביל לתשובה נכונה באיטרציה הנוכחית ולתשובה שגויה באיטרציה הקודמת גורמת לעדכון גדול יותר עבור המודל. הגישה מקורה במה שנקרא memoized wake-sleep שמציע שיטת אימון למודלים נוירו-סימבוליים גנרטיביים בכלל דרך ניצול הזכרון המצטבר של העדכונים.

וכל זה כדי לשפר את הריזונינג של המודל - מקווה שהצלחתם להבין את העיקר🙂
https://arxiv.org/pdf/2312.02179

🚀המאמר היומי של מייק 20.09.24: ⚡🚀
Training Large Language Models for Reasoning through Reverse Curriculum Reinforcement Learning

ממשיכים בסקירות מאמרים "החשודים" בסלילת נתיב למודל o1 (שרבים כבר התאכזבו ממנו אמנם אך אותי הוא מסקרן מבחינת חידוש הפרדיגמה). המאמר שנסקור היום פחות מתמטי מזה של אתמול (הכל פורסם בערוץ הטלגרם שלי) ובתקווה הסקירה תהיה יחסית קצרה וקולעת.
מזכיר שהמאמר מציע שיטה לשיפור הרוזונינג של מודלי שפה כאשר יש לנו דאטהסט D גדול יחסית של שאלות ותשובות ודאטהסט קטן בהרבה של שאלות ותשובות עם שרשרת ריזונינג. המאמר מציע שיטה בסגנון של למידת curriculum די נפוצה בלמידה עמוקה - כמה מודלי שפה הכי טריים אומנו עם השיטה הזו (בשילוב עם עוד שיטות כמובן). בלמידת curriculum מאמנים מודל החל מדוגמאות קלות ובמהלך הלמידה מעלים את קושי הדוגמאות.

אבל איך קשורה למידת curriculum לשיפור יכולת ריזונינג של מודל שפה. וזה בדיוק היופי של המאמר דרך אגב. המחברים שמו לב שאם נספק למודל את כל שרשרת הריזונינג מהתחלה ועד השלב די קרוב לתשובה הסופית אז יהיה לא יותר קל לשחזר את השלבים החסרים בשרשרת. וזה בדיוק מה שהמאמר עושה. כלומר המאמר מאמן

את מודל (בשיטת RL דומה ל-STaR שסקרתי ב 17.09, למידת פוליסי די סטנדרטית) אבל הפעם המודל לומד לשחזר את שלבי הריזונינג מנקודות שונות בשרשרת.

המאמר טוען ששיטת למידת curriculum הסטנדרטית פחות מתאימה למקרה הזה כי המודל שלמד להשלים שלבי ריזונינג אחרונים מתקשה ללמוד לעשות את מההתחלה ו"מאבד" את הידע שצבר. בעקבות כך המחברים מאמנים משימות ריזונינג ברמות קושי שונות (בהקשר המדובר) יחד עם איזושהי אסטרטגיה חכמה מעולם ה-multi-tasking.

שני דברים אחרונים לגבי המאמר הזה. קודם כל פונקצית תגמול (reward) הינה די סטנדרטית כאן עם חידוש קטן שעבור משימות עם תשובה מספרית המודל מקבל פרס קטן (ולא אפס) אם הוא נותן תשובה מספרים לא נכונה (ו-1 במקרה של תשובה נכונה). המאמר משתמש ב-PPO שהיא שיטה די סטנדרטית לפיינטיון של LLM אם אתם לא רוצים שהוא ישכח את כל מה שהוא למד לפני הפיינטיון.

https://arxiv.org/pdf/2402.05808

🚀המאמר היומי של מייק 21.09.24: ⚡🚀
REFT: Reasoning with REinforced Fine-Tuning

ממשיכים לסקור מאמרים שסללו לכאורה נתיב ל-o1. הפעם מאמר די בסיסי יחסית שהיה שווה לסקור אותה לפני יומיים אך התעצלתי לעבור על רשימת המאמרים שבניתי כדי להבין את זה. הרווח היחיד לאלו שעוקבים אחרי סקירותיי באופן יום יומי יתבטא בכך שיהיה לכם מאוד קל להבין את הסקירה הזו אם הצלחתם להבין (בערך) את 4 הקודמות.

המאמר מניח שיש בידינו דאטהסט של שאלות ושרשרת הריזונינג המובילה לתשובה (הנכונה). המאמר מציע לשפר את יכולת הריזונינג של מודל שפה בשני שלבים:

אימון רגיל (Self-Supervised Fine Tuning): על כל שרשראות הריזונינג מהדאטהסט. כלומר המודל לומד לשחזר את שרשרת הריזונינג של כל שאלה ברמת הטוקן כמו ש נעשה ב-SFT הסטנדרטי.

אימון של למידת פוליסי (שזה המודל עצמו) מעולם Reinforcement Learning: (מכאן נגזר שם המאמר) כאשר המודל מקבל פרס 1 אם המליח לגנרט שרשרת ריזונינג המובילה לתשובה הנכונה. תגמול צנוע הרבה יותר ניתן לתשובות מספריות לא נכונות עבור השאלות שהתשובות עליהן מספריות גם כן (כמו במאמר הקודם). תגמול 0 מתקבל בכל המקרים האחרים. אימון מתבצע עם PPO די סטנדרטי עם שערוך די סטנדרטי של פונקציית ערך V ופונקצית יתרון A (כמו במאמר המקורי של ג'ו שולמן מ-openai לשעבר)

https://arxiv.org/pdf/2401.08967

🚀המאמר היומי של מייק 22.09.24: ⚡🚀
Quiet-STaR: Language Models Can Teach Themselves to Think Before Speaking

סקירה זה ממשיכה את קו הסקירות "בדרך ל-o1" והפעם המאמר לפחות לפי השם התקרב די מהר למה קורה לכאורה ב-o1. כלומר "המודל חושב" לפני שהוא מחזיר את תשובתו למשתמש. כמובן שגם המטרה כאן גם שיפור ריזונינג של המודל.

המאמר משפר את Sta שסקרתי לפני כמה ימים ועושה את דרך בניית "שרשראות ריזונינג לוקליים" העוזרים

למודל לחזות בצורה יותר מדויקת. כל שרשרת ריזונינג כזו מורכבת ממה שנקרא במאמר "טוקני חשיבה" (thought tokens) שהמודל מייצר ותהליך זה ניתן לפרש בתור "חשיבה של המודל". מכיוון שחיזוי של רוב הטוקנים אינה משימה קשה במיוחד ולא נדרשים עבורה טוקני חשיבה המאמר מציע לשלב את הייצוג המגיע מטוקנים אלו עם ייצוג הקונטקסט המופק מהטקונים הקודמים.

לטענת המחברים (הדי הגיונית) טוקני חשיבה של טוקן נתון עוזרים לא רק לחיזוי של הטוקן הבא אלא גם לטוקנים שבאים אחריו. אז המודל מאומן למקסם את דיוק החיזוי כמה מהטוקנים הבאים. בנוסף המאמר מאמנים טוקנים מיוחדים המסמנים את ההתחלה ואת הסוף של שרשראות טוקני החשיבה: <|startofthought|> ו-<|endofthought|> שגם את הייצוגים שלהם נלמדים במהלך האימון.

האימון מתבצע בשיטת REINFORCE מאוד סטנדרטית הלקוחה מעולם למידה עם חיזוקים. בכל איטרציה עבור כל שרשרת של טוקני חשיבה אנו ממקסמים את ההפרש בין איכות החיזוי של כמה טוקנים הבאים (כלומר log-likelihood) לבין הממוצע של אותה איכות החיזוי עבור כמה שרשראות טוקני חשיבה (שנבנים כל פעם). המאמר טוען שזה מקטין את השונות שערוך ה-likelihood. ד"א מה שממקסמים זה סוג של פונקציית ה-advantage שדי נפוצה בעולם RL. כאמור השכבה (mixing head) המשלבת ייצוג טוקני חשיבה יחד עם ייצוג הקונטקסט הרגיל מאומנת גם כן.

יחד עם זה המודל עצמו (המשקלים) מאומן יחד עם טוקני חשיבה וכל השאר (ראו את האלגוריתם). המחברים שמו לב שלא צריך לבנות טוקני חשיבה לטוקנים הנחזים בקלות (אנטרופיה נמוך של וקטור ההסתברויות) ומאפשר לחסוך לא מעט כוח חישוב באינפרנס.

https://arxiv.org/pdf/2403.09629

🚀המאמר היומי של מייק 23.09.24: ⚡🚀
Training Language Models to Self-Correct via Reinforcement Learning

ממשיכים בקו הסקירות על שיפור יכולת הריזונינג של מודלי שפה (מסדרת "כל הדרך ל o1"). המאמר הזה של דיפמיינד שיצא לפני כמה ימים משך את עיניי מרגע ששמתי לב עליו (לראשונה ראיתי אותו בלינקדאין נראה לי). לקח לי לא מאוד זמן להבין את העיקר של המאמר הזה כי הוא מכיל הסברים מאוד מפורטים ומעמיקים והיתה לי הרגשה ש"מרוב עצים לא רואים את היער".

אוקיי כמו שאתם כבר מבינים מהשם המאמר מציע שיטה לשיפור של יכולות תיקון עצמית (self-correction) של מודלי שפה. הנושא נחקר רבות בשנתיים האחרונות (וגם לפני) והוצעו מספר שיטות לטיפול בבעיה. אלא, כמו שמחברי המאמר מציינים שיטות אלו אינן מובילות לשיפור ביצועים משמעתי עקב העובדה שהן מאומנים על התפלגות מוטעית של התפלגות התשובה הראשונה (שאותה מתקנים) של ה-LLM (זה מה שלקח לי לא מעט זמן לזקק מהמאמר).

המאמר מתבונן בשתי שיטות לתיקון עצמי (הם עשו SFT על הדאטהסטים המגונרטים על ידיהם): Star (שסקרתי לפני כמה ימים) ומהמאמר הזה (נקרא Pair-SFT במאמר). בגישה בנו דאטהסט על שלישיות המכילות שאלה, תשובה לא נכונה (כלומר שרשרת ריזונינג המוביל אליה) ותשובה נכונה (גם הריזונינג שהוביל אליה) כאשר ניתנה על ידי המודל אחרי התיקון העצמי (עם פרופמט מסוים). במקרה השני השלישיה הורכבה מהשאלה, תשובה לא נכונה ותשובה נכונה אקראית (לא אחרי התיקון עצמי) לשאלה הזו.

בשני המקרים המחברים ראו שאין שיפור משמעותי אחרי התיקון העצמי ואחרי אנליזה די רצינית הגיעו למסקנה כי זה נובע מאי "התאמה של התפלגות התשובה הראשונה" להתפלגות ההתחלתית של המודל. הרציאונל כאן

הוא שאנו מאמנים מודל לתקן לא בדיוק מה שהמודל יוצר אלא משהו קצת אחר.

המחברים מציעים שיטה דו שלבית לפתרון בעיה זו. בשלב הראשון אנו מנסים לגרום למקסם את תגמול(מניחים שיש פונקצית reward נתונה) עבור תשובה נכונה אחרי תיקון עצמי (כלומר שרשרת ריזונינג שמובילה לתשובה הזו תוך שמירה של התפלגות הפלט המאומנת (או פוליסי בשפת RL) של LLM קרובה להתפלגות התחלתית שלו. כלומר עושים סוג של PPO כאשר היעד הוא מקסום של ההפרש בין התגמול עבור התשובה הנכונה לבין מחקר KL בין ההתפלגות המאומנת (כלומר פוליסי) להתפלגות ההתחלתית.

בשלב השני ממקסמים את סכום התגמולים אחרי שתי התשובות (לפני ואחרי התיקון) תוך שמירה של הקירבה של ההתפלגויות שלהם להתפלגות ההתחלתית של LLM.

מקווה שהסברתי פחות או יותר מובן...

https://arxiv.org/pdf/2409.12917

🚀⚡🚀 :המאמר היומי של מייק 24.09.24🚀
LLMS STILL CAN'T PLAN; CAN LRMS? A PRELIMINARY EVALUATION OF OPENAI'S O1 ON PLANBENCH

סקירה של מאמר שלא מכיל מתמטיקה בצורה מפורשת...מאמר זה בוחן את יכולות התכנון של מודלי שפה גדולים (LLMs) ומודלי חשיבה גדולים (LRMs) כמו משפחת o1 באמצעות סדרת מבחנים הנקראת PlanBench.

PlanBench הוא מערך מבחנים מקיף שפותח ב-2022 להערכת יכולות התכנון של LLMs. מרכיביו העיקריים:

- מערכת סטטית של 600 בעיות Blocksworld הכוללות 3 עד 5 קוביות.
- גרסה מוסתרת (Mystery Blocksworld) של אותן בעיות, שבה המונחים והפעולות מוחלפים במילים אקראיות כדי לבחון הבנה מופשטת.
- בעיות Blocksworld מורכבות יותר עם 6 עד 20 קוביות, הדורשות תוכניות ארוכות יותר של 20 עד 40 צעדים.
- בעיות בלתי פתירות, שנוצרו על ידי הוספת "יעד" בלתי אפשרי לבעיות קיימות.

PlanBench נועד להיות כלי גמיש ומקיף להערכת יכולות תכנון של מודלי שפה תוך בחינת היבטים שונים של תכנון כמו הבנה מופשטת, התמודדות עם מורכבות, וזיהוי בעיות בלתי פתירות.

החוקרים מצאו כי LLMs השתפרו בביצועי תכנון בסיסיים, כאשר המודל הטוב ביותר, LLaMA 3.1 405B, השיג דיוק של 62.5% במשימות Blocksworld פשוטות. עם זאת, LLMs נכשלו במשימות בעלי פתרון סבוך יותר.

לעומת זאת, מודל ה-LRM החדש של OpenAI, o1, הציג שיפור משמעותי, עם דיוק של כמעט 98% במשימות Blocksworld פשוטות ו-52.8% במשימות עם פתרון סבוך. למרות זאת, הביצועים של o1 ירדו משמעותית במשימות מורכבות יותר ובבעיות בלתי פתירות.

עם זאת החוקרים מדגישים את החשיבות של הטרייד-אופים הכוללים יעילות, עלות וערבויות לנכונות הפתרון (ככה כתוב במאמר) בהערכת מודלים אלה. הם מציינים כי o1 יקר משמעותית להפעלה ואינו מספק ערבויות לנכונות, בניגוד למתכנני AI קלאסיים. המסקנה היא שבעוד LRMs כמו o1 מציגים התקדמות, הם עדיין רחוקים

מלהיות פתרון כללי ואמין לבעיות תכנון.

🚀המאמר היומי של מייק 26.09.24: ⚡🚀
RRM: ROBUST REWARD MODEL TRAINING MITIGATES REWARD HACKING

מאמר נחמד שמשך את עיניי עקב העובדה שהוא דן בנושא פונקציית תגמול (reward model או RM) של מודלי שפה. RM הנחוץ בתהליך היישור (alignment) של מודלי השפה המבוססים על RLHF שמטרתו מאוד בגדול לאמן מודל שפה להבחין בין תשובה טובה לתשובה רעה.

הנושא נחקר באינטנסיביות בשנים האחרונות והוצעו מספר שיטות לעשות רובן שכלולים שונים של (Proximal Policy Optimization (PPO כגון DPO, ORPO ועוד רבים שחלקם סקרתי. בדרך כללי לאימון RLHF נדרש דאטהסט המורכב משלישיות של שאלות ו-2 תשובות, אחת יותר מועדפת (המנצחת או w) והשנייה הפחות מועדפת (מפסידה או l). במהלך אימון RLHF המודל לומד להגדיל את הנראות של התשובה w להקטין את הנראות של תשובה l דרך מקסום של הפרש ה-reward שלהם (עם סיגמויד ולוג) תחת אילוצים כמו שמירה על הקרבה בין התפלגות הפלט של המודל המאומן למודל ההתחלתי.

המאמר מציע להתבונן באימון RLHF מזווית די מעניינת ושואל את השאלה הזה האם הצורה של תשובות משפיעות לנו בצורה לא מכוונות על תוצאת אימון בלי קשר לשאלה. כלומר המודל עושה "reward hacking" ומשתמש בתכונות של התשובות בלבד ללא קשר לשאלה כדי לאפטם את משקלי המודל. כלומר המודל יכול ללמוד לנצל דפוסים שונים כמו (sure, this is the response או n-grams מסוימים של התשובות) **בלבד.**

כדי להתגבר על הבעיה הזו המאמר מציע לערבב תשובות לשאלות שונות כלומר לעשות סוג של אוגמנטציה ולאמן את המודל כך שזה יקשה עליו לבצע reward hacking. למשל שתי תשובות לא רלוונטיות משאלות אחרות (w ו- l) לשאלה נתונה אמורות לקבל אותו התגמול ואילו תשובה w המתאימה לשאלה ותשובה l משאלה אחרי אמורה עדיין לתת reward גבוה ל-w ו-reward נמוך ל-l מהשאלה האחרת. יש כמובן צירופים נוספים שניתן להנדס ולאמן את המודל עליהם בצורת RLHF.

דרך אגב המאמר בונה פריימוורק סיבתי לבעיה הזו כולל DAG, סטים שהם d-separate וכדומה אבל אני לא בטוח שכל זה נחוץ להבנת המאמר . זה אמנם שגזל ממני זמן רענון המושגים האלו אבל כמה שיחות עם סונט עזרו לי מאוד.

https://arxiv.org/abs/2409.13156

🚀המאמר היומי של מייק 27.09.24: ⚡🚀
REWARD-ROBUST RLHF IN LLMS

הסקירה של היום הינה בנושא שהוא די דומה לסקירה של אתמול (26.09.24). נושא של הסקירה הוא שיפור של יישור (alignment) של מודלי שפה במהלך אימון RLHF. גם המאמר הזה מציע שיטה שבאה "לתקן" את פונקציית התגמול (reward) אבל מזווית טיפה שונה מאשר המאמר שסקרנו קודם.

המחברים מצביעים על כך ששימוש בפונקציית תגמול יחידה במהלך אימון RLHF אינו אופטימלי מכמה סיבות. הסיבה הראשונה היא חוסר עקביות בין המתייגים במהלך תיוג הדאטה המשמש לאימון RLHF (כלומר תשובות מועדפות ולא מועדפות לשאלות מהדאטהסט) שעלול לגרום לתשובות "מבולבלות" של המודל לאחר האימון. הבעיה השניה היא reward hacking של המודל המתבטא בכך שהמודל לומד להחזיר תשובות הממקסמות את פונקציית התגמול תוך מתן תשובות לא "מיושרות" עם העדפות המתייגים או לא הגיוניות.

המאמר ניגש לסוגיה זו מנקודת מבט בייסיאנית. אם נניח שקיימת פונקציית תגמול אידאלית שאין לנו גישה אליה אז ניתן להתבונן בכל פונקציית תגמול שנבנה איזה דגימה ממרחב "פונקציות תגמול רועשות". המחברים מציעים

לכמת את אי וודאות שיש לנו בפונקציית התגמול על ידי אימון של כמה פונקציות תגמול.

אז איך כל הסיפור הזה עובד? קודם כל מאמנים פונקציית תגמול רגילה דרך נוסחת Bradley-Terry הסטנדרטי.

לאחר מכן מאמנים כמה פונקציות תגמול שימדלו לנו את אי הוודאות. בשביל זה לוקחים backbone רגיל (מודל שפה) ומוסיפים אליו כמה ראשים (heads) שכל אחד הוא למעשה פוקנצית תגמול. כל ראש מאומן לפלוט את התוחלת ואת השונות של ערך התגמול והתגמול עצמו מוגרל מהתפלגות גאוסית המוגדרת על ידיהם.

פונקציית לוס שהם משתמשים לאימון הראשים היא די לא טריוויאלית אך בגדול ממזערת את השגיאה הריבועית של שערוך התגמול (וזה קצת מורכב ומסתמך על פונקציית תגמול סטנדרטית מהשלב הראשון בנוסף לגישת Bradley Terry). במהלך האימון כל דוגמא מוגרלת (מנווטת) לראש שלו וכך אנו מקבלים כמה פונקציות תגמול.

המחברים אומרים שהם "היו רוצים" (והם השתמשו בה על דוגמאות הצעצוע שלהם) לבנות את הלוס עבור אימון RLHF בתור צירוף לינארי של פונקצית התגמול הרגילה התגמול המינימלי בין כל פונקציות התגמול. כאן האיבר השני למעשה מהווה שערוך של אי הוודאות שדנו בה למעלה. באופן פרקטי במהלך אימון RLHF הם בוחרים ערך התגמול המתקבל בפונקציית התגמול בעלת שונות הנמוכה ביותר.

https://www.arxiv.org/abs/2409.15360

🚀⚡ המאמר היומי של מייק 28.09.24🚀:
Meta-Whisper: Speech-Based Meta-ICL for ASR on Low-Resource Languages

מזמן לא סקרתי מאמר על אודיו ומשלים את הפער היום עם סקירה קצרה וקלילה. בדיוק כמו במודלי שפה גם במודלי אודיו כמו whisper למשל יש יכולת למידה in-context או ICL בקצרה. ICL היא יכולת של מודל לבצע משימה שלא אומן עליה באופן מפורש אחרי ש"מראים לו" כמה דוגמאות המדגימות את המשימה (נגיד, כמה זוגות של שאלות ותשובות רצויות).

מתברר שמודלי אודיו גם ניחנים ביכולת כזה. כלומר בהינתן זוג של קטעי אודיו (שאלה ותשובה) ניתן לאמן את המודל לענות על שאלה אחרת, שמוגשת לא לאחר כן בצורה של טקסט. אבל איך ניתן לבחור את הדוגמא מהדאטהסט (אודיו) של שאלות ותשובות שתמקסם את ביצועי המודל לשאלה נתונה.

זה בדיוק מה שהמאמר המסוקר עושה. הוא מציע לבחור זוג אודיו (שאלה ותשובה) לשאלה טקסטואלית נתונה על סמך דמיון בין ייצוגה לבין הייצוג של הזוג. הייצוג כאן הוא הפלטים (hidden states) של השכבות השונות של המודל עבור האודיו והשאלה הטקסטואלית. והמטריקה KL divergence הדי סטנדרטי. לדאטהסט אודיו של שאלות ותשובות נתון אני שומרים את כל הפלטים של השכבות ולכל שאלת אודיו בוחרים את הזוג הדומה ביותר לפי מטריקה זו.

שכחתי לציין שהמודל עובר פיינטיון למשימת ICL בשיטת LoRA הידועה...

זהו זה - סקירה קלילה כמו שהבטחתי.

https://arxiv.org/abs/2409.10429

🚀המאמר היומי של מייק 29.09.24: ⚡🚀
ASR Error Correction using Large Language Models

ממשיך לסקור מאמרים בדומיין אודיו. הפעם נדבר על מאמר המציע שיטה לשיפור איכות של פענוח אות דיבור ניתן להשתמש בה במערכות ל-Automatic Speech Recognition או בקצרה ASR. המטרה בכל הסיפור הזה היא לתמלל אות קולי או במילים פשוטות להבין מה נאמר שם.

בד"כ הקלט ל- ASR הוא כמה פלטים של המודול שנקרא Error Correction או EC שמטרתו היא ליצור כמה וריאנטים של תמלול **Z** (בעלי "סבירות גבוהה ביותר") עבור אות דיבור נתון. למעשה מטרתו של ה- EC היא לבנות את התמלול הסופי בהינתן **Z.**

בעידננו של מודלי שפה עוצמתיים ניתן למנף את יכולתם למשימה הזו בצורה די ישרה. כלומר אנו מזינים ל-LLM את הוריאנטים השונים של התמלול ומבקשים מ-LLM לבחור את התמלול הגיוני ביותר מבחינה סמנטית (עם פרומפט מתאים). המאמר בחר LLM לא סטנדרטי המורכב מאנקודר ומדקודר (כמו במאמר המקורי של הטרנספורמרים) למשימה זו וזה עבד לא רע. אם יש לנו דאטהסט המכיל את התמלולים מה-ASR והתמלול הנכון, ניתן לבצע פיינטיון.

האם ניתן לעשות יותר טוב? מתברר שכן אם בנוסף לתמלולים אנו מזינים למודל שפה גם את תכונות אות הדיבור עצמו (למשל ייצוגו אחרי האנקודר או מטה-דאטה שלו) ניתן לשפר את הביצועים של ה-EC. המחברים מציעים לבנות את התוצאה באמצעות מקסום של סכום משוקלל של הנראויות (log-likelihoods) מהסעיף הקודם (בהינתן התמלולים מהסעיף הקודם) והנראות של התמלול בהינתן התכונת של סיגנל הדיבור עצמו. באופן לא מפתיע זה משפר את הביצועים כי המודל מקבל יותר מידע רלוונטי.

עוד שכלול אחד הוא תוספת ההתחשבות במרחק Levenstein מינימלי בין הפלט הסופי של EC לבין הפלטים של ASR (המוזנים ל-EC). מרחק לבינשטיין הוא מדד הבודק את מספר השינויים המינימלי הנדרש כדי להפוך מחרוזת אחת לאחרת. כלומר אנו בוחרים את התיקון הקרוב ביותר (מבחינת LD) לאחד הפלטים של ה-ASR.

מקווה שלא פספסתי שום דבר...

arxiv.org/pdf/2409.09554

🚀המאמר היומי של מייק 30.09.24: ⚡🚀
SCHRODINGER'S MEMORY: LARGE LANGUAGE MODELS

ביום הסוער הזה (למרות שהסקירה שייכת פורמלית לאתמול - אשלים את הפער בימים הקרובים) נסקור מאמר די קליל עם שם מאוד לא קליל. כי אין דבר קליל שכולל בתוכו את שמו של שרדינגר - ספק אם הצלחתי להבין בצורה טובה מספיק את המשוואה של שרדינגר עוד בקורס פיזיקה 3 באוניברסיטה במוסקבה לפני עשרות שנים. גם סיפורו של חתול שרדינגר לא התבהר עד עכשיו.

אוקיי, סיימנו עם הצחוקים. המאמר חוקר (אמפירית) נושא די רציני והוא הזכרון של מודלי שפה. כשאנחנו שואלים LLM מה עיר הבירה של שבדיה, איך הוא יודע שזה סטוקהולם. המאמר טוען כי זיכרון LLM פועל על ידי התאמה דינמית של פלטים לקלטים. כלומר המודל "בוחר" איך לשלוף את המידע מהזיכרון ובונה אותו על סמך הקלט.

המחברים מסבירים את איך פועל הזיכרון של מודלי שפה באמצעות ניתוח של ארכיטקטורת הטרנספורמרים. מנגנון ה-attention (כלומר מקדמי ה-attention שלו) למעשה מאפשרים למודל לבנות את הפלט כפונקציה דינמית של הקלט (כלומר לא קבועה כמו ב-MLP או ConvNets).

המחברים משתמשים ב- Universal Approximation Theorem או UAT כדי להסביר את היכולת של שליפת מידע שנלמד במהלך האימון על בסיס תוכן של הקלט. המחברים טוענים כי ניתן להבין מנגנון זה בתור "יכולת

קירוב דינמית בסגנון UAT" (המשפט המקורי מדבר על יכולת קירוב סטטית של מודלי ML) כאשר המודל מתאים תוצאה מתאימה על בסיס הקלט, והתופעה הנצפית ניתן להגדיר בתור זיכרון.

הם מכנים זאת "זיכרון שרדינגר" מכיוון שאנו יכולים לקבוע של-LLMs יש את הזיכרון הזה רק על ידי ״שאילת שאלות״ וניתוח התגובה שלו; אחרת, הזיכרון נשאר בלתי מוגדר. בנוסף במאמר נדונים גורמים המשפיעים על ביצועי LLM: גודל המודל, איכות/כמות הדאטה והארכיטקטורה. המחברים טוענים שהזיכרון של מודלים באותו הגודל מושפע מאופן האימון שלהם ואם המודל אומן על יותר דאטה איכותי אז הוא משתפר (אין הפתעות כאן).

ולבסוף נעשות הקבלות בין ארכיטקטורת LLM למבנה המודולרי של המוח האנושי (את זה פחות אהבתי אבל זרמתי).

https://arxiv.org/pdf/2409.10482

🚀המאמר היומי של מייק -01.10.24: ⚡🚀
Larger and more instructable language models become less reliable

שנה טובה, מתוקה ושקטה לעוקביי היקרים! אני חושד שהמאזן הקלורי של רובכם הופר בבוקר אז אני מביא לכם סקירה קלילה (פורמלית של אתמול). ודרך אגב הסקירה של היום תהיה אוסף של כל הסקירות עד עכשיו ואני אפרסם את זה מחר בבוקר.

המאמר שנסקור היום הוא לא מתמטי והוא דן ביכולות של מודלי שפה. המדד מתבונן ביכולות של מודלי שפה לפתור בעיות דרך הפריזמה של 3 מדדים שונים. השניים מהם הם די סטנדרטיים וברורים והם אחוז נכונות/אי נכונות של התשובה אך השלישי הוא אחוז הימנעות של מודל שפה מהתשובה. אכן בלא מעט מקרים מודלי שפה בוחרים להגיד לנו שלא יודעים את התשובה ולפעמים זה די מעצבן (אבל לפעמים ממש לא).

המחברים מצאו כי LLMs נכשלים ביצירת "אזורי פעולה אמינים לבעיות קלות": אפילו במשימות הנתפסות כפשוטות על ידי בני אדם, LLMs ממשיכים לעשות טעויות. כלומר אין "מקלט בטוח" ברור של באיזור קושי נמוך שבו המודלים מבצעים באופן עקבי ללא שגיאות.

שיפורי ביצועים (הנובעים מאימון דאטה יותר טוב, אימון משופר ויישור) מתרחשים בעיקר עבור בעיות מורכבות, בעוד lLLMs ממשיכים לטעות במקרים קלים: כלומר LLMS יותר חזקים מראים ביצועים משופרים במשימות מאתגרות. עם זאת, שיפור זה אינו מתרחב באופן אחיד למשימות פשוטות יותר, מה שיוצר חוסר התאמה בין ציפיות אנושיות לביצועי המודל.

אימון יעיל (המאמר קורא לזה shape-up) מפחיתים הימנעות אך מגבירים אי-נכונות של התשובות: המאמר מראה שמודלים חדשים וחזקים יותר פחות נוטים להימנע ממתן תשובות. עם זאת, הפחתה זו בהימנעות מלווה לעתים קרובות בעלייה בתשובות לא נכונות במקום תשובות נכונות.

בנוסף אחוז הימנעות לא עולה עם רמת הקושי של הבעיה: היינו רוצים כי Prob(הימנעות|קושי) יהיה קבוע, כלומר מודלים היו נמנעים מלענות לעתים קרובות יותר ככל שקושי המשימה עולה. אולם המחברים מראים ששיעורי ההימנעות נשארים יחסית קבועים בכל רמות הקושי.

המחברים גם בדקו את יציבות תשובות המודל לניסוחים שונים של הבעיה ומצאו כי מודלים חזקים יותר מפגינים יציבות גבוהה יותר לניסוח המשימה (פרומפט). כלומר תשובתם פחות תלויה בניסוח הבעיה. למרות שיפורים ביציבות, עדיין יש אזורים (של בעיות) שבהם הביצועים יכולים להשתנות משמעותית בהתאם לניסוח שנעשה בו

שימוש, אפילו עבור מודלים מעוצבים.

בנוסף השיפורים ביציבות התשובה לא מונוטוניים (מבחינת קושי הבעיה): חלק מהניסוחים (של הבעיה) מבוצעים טוב יותר במקרים מורכבים אך גרוע יותר במקרים קלים: הקשר בין יעילות הניסוח וקושי המשימה אינו תמיד פשוט. חלק מהניסוחים שעובדים היטב למשימות מאתגרות עשויים לבצע באופן גרוע במשימות קלות יותר, מה שמסבך את תהליך בחירת הניסוח.

עוד תוצאות מעניינות רבות במאמר הזה - ממליץ בחום להעיף מבט...

https://www.nature.com/articles/s41586-024-07930-y

🚀⚡ המאמר היומי של מייק -03.10.24🚀:
Transformers are Expressive, But Are They Expressive Enough for Regression?

שוב מאמר על הטרנספורמרים אבל קצת שונה מהמאמר הסטנדרטי על LLMs. המאמר הזה מציג חקירה מעמיקה לגבי expressiveness של הטרנספורמרים, תוך בחינה ספציפית של יכולתם בתור משערכי פונקציות אוניברסליים (כאלו שניתן לקרב איתם כל פונקציה חלקה בדיוק נתון). המחברים מאתגרים טענות קיימות לגבי expressiveness של הטרנספורמרים ומספקים הוכחות תיאורטיות ואמפיריות כאחד שתומכים בהשערתם שהטרנספורמרים מתקשים לקרב (לשערך) באופן מדויק פונקציות חלקות.

לפני 4 שנים הוכח שהטרנספורמר(האנקודר) מסוגל לשערך כל פונקציה רציפה אם יש בו מספיק שכבות (בלוקים של טרנספורמר). המשפט הוכח לפני כ 4 שנים והוא מראה שהטרנספורמר בעל שכבות מרובות למעשה יודע לשערך ופונקציה קבועה למקוטעין (piecewise constant) ועם הגודל המינימלי של אינטרוול הקביעות (=רזולוציה) δ הינו קטן מדי אז ניתן לשערך באמצעותו כל פונקציה חלקה בכל דיוק.

המאמר המסוקר מתמקד במחקר של הרזולוציה δ הנדרשת לשערוך בדיוק נתון של פונקציה חלקה. התרומה התיאורטית המרכזית של המאמר היא משפט 4.1, אשר קובע חסם עליון על גורם הרזולוציה δ עבור שמכיל מאפיינים שונים של פונקציה מקורבת f.

משפט זה משמעותי מכמה סיבות:

א) הוא קושר ישירות את גורם הרזולוציה δ לנגזרות של f. קשר זה מבהיר מדוע פונקציות חלקות עם נגזרות המשתנות במהירות מהוות אתגר קשה עבור טרנספורמרים.

ב) החסם מראה יחס הפוך בין δ לבין הנגזרות החלקיות של הפונקציה. עבור פונקציות עם נגזרות גדולות, δ חייב להיות קטן כדי לשמור על איכות הקירוב. זה אומר בעצם שאנו צריכים יותר שכבות של טרנספורמרים כדי לקרב בדיוק גבוה את f.

ג) המונח האקספוננציאלי 1/(p+md) בחסם מצביע על כך שככל שממד הקלט m או ממד האמבדינג d גדלים, גורם הרזולוציה δ חייב לקטון אקספוננציאלית כדי לשמור על אותה איכות קירוב.

ד״א המחברים מספקים הוכחה מפורטת למשפט זה, תחילה למקרה החד-ממדי ולאחר מכן בהכללה לממדים גבוהים יותר..

יתר על כן, המחברים מקשרים את התוצאה התיאורטית הזו להשלכות המעשיות על ארכיטקטורות טרנספורמר. הם מראים שמספר השכבות הנדרש לקירוב הולם גדל כ O(m(1/δ)^(dm)), מה שהופך ללא ישים מבחינה חישובית עבור δ קטן וממד הקלט בגודל בינוני m. כלומר צריך יותר מדי שכבות הטרנספומרים בשביל זה.

המחברים ביצעו ניסויים מקיפים על הטרנספורמר כדי להשלים את ממצאיהם התיאורטיים. הם עשו 2 ניסויים עם הבנצ'מרקים הבאים:

א) EXPT-I (רגרסיה): בדיקת יכולתם של טרנספורמרים לקרב ישירות פונקציות חלקות.

ב) EXPT-II ("סיווג מקוונטט"): בדיקת יכולתם של טרנספורמרים לקרב פונקציות קבועות למקוטעין.

התברר כי הטרנספורמרים מתפקדים באופן גרוע משמעותית ב-EXPT-I בהשוואה ל-EXPT-II, שזה תומך בהשערה שהם מתקשים בקירוב פונקציות חלקות.

הגדלת מספר השכבות, ראשי מנגנון ה-attention, או ממדי אמבדינג אינה משפרת באופן משמעותי את הביצועים על פונקציות חלקות. לעומת הטרנספורמרים מצליחים לקרב באופן הולם פונקציות קבועות למקוטעין עם רזולוציה δ לא קטנה במיוחד.

https://arxiv.org/pdf/2402.15478

🚀המאמר היומי של מייק -04.10.24: ⚡🚀
?Were RNNs All We Needed

המאמר הזה משך את תשומת ליבי כי יש לו "all we needed" בכותרת. מסיבה שאינה ב-100% ברורה לי מאמרים כאלו יוצרים בי דחף חזק לסקור אותם. אז ככה הגעתי למאמר הזה שאלולא השם כנראה שלא הייתי מגיע אליו.

המאמר מציע לשפצר את ה-RNN כך שנוכל להפעיל אותו בצורה מקבילית במהלך האימון. הסיבה העיקרית ש-RNN כמעט יצא מכלל שימוש היום הוא חוסר היכולת שלו להתאמן באופן מקבילי כלומר לבצע חיזוי של כמה טוקנים ממוסכים. הטרנספורמרים לעומת זאת כן ניחנים ביכולת הזו אך יש להם מגבלה בדמות סיבוכיות ריבועית במונחי אורך הסדרה (שכואבת לנו בעיקר באינפרנס כי מאמנים אותם פעם אחת) שמקשה על השימוש (לפחות הנאיבי שלהם) לסדרות מאוד ארוכות.

מצד שני ל-RNNs יש יכולת יותר טובה לעבד סדרות מאוד ארוכות כי כל ה"זיכרון" שלהם מקודד בכמה ווקטורים (1,2 או 3) והסיבוכיות החישובית שלהם פרופורציונלית לאורך הסדרה ולא לריבוע שלה (גם באימון וגם באינפרנס). כאמור הבעיה הגדולה של ה-RNNS שדי הרגה את הארכיטקטורה הזו היא אי יכולתה לאפשר חיזוי מקבילי באימון. זה שהופך את האימון על כמויות דאטה עצומות כמו שמקובל היום (עשרות טריליונים טוקנים) עם RNNs לארוך מדי ולא פיזיבילי.

חשוב להבין שהסיבה לחוסר יכולת לחזות בצורה מקבילי נובעת מהמעברים הלא לינאריים בין המצבים החבויים ב-RNN (גם ב-LSTM וגם ב-GRU).

לאחרונה SSMs (או State Space Models) ניסו לטפל בבעיה הזו דרך ארכיטקטורה שבה המעברים האלו כן לינאריים וארכיטקטורת ממבה (שסקרתי בהרחבה לפני כמה חודשים) ששכללה SSMs לרמת ביצועים קרובה לטרנספורמרים. בנוסף A21 Labs השתמשו בממבה כאבן בניין של הארכיטקטורה החדשה שלהם לפני כחודשיים(יחד עם הטרנספורמרים).

עכשיו אתם שואלים מה המאמר המסוקר עשה בנידון. כאמור הבעיה הגדולה ב-RNN היה מעברים לא לינאריים בין המצבים החבויים. המחברים פשוט הורידו את התלות הלא לינארית מהמשוואות של LSTM ו-GRU. מה שהתקבל כתוצאה מכך ניתן למקבול במהלך האימון (אבל דורש יותר זיכרון מהגרסאות הרגילות). יצא משהו די

דומה לממבה - גם כן המצב החבוי תלוי באופן ליניארי במצב החבוי הקודם ובאופן לא לינארי בייצוג האיבר הנוכחי של סדרת הדאטה.

מה שמפתיע אותי קצת כאן זה ביצועים טובים מדי - אני קצת חשדן אבל בואו נראה מה קורה עם הארכיטקטורה הזו בעתיד.

https://arxiv.org/abs/2410.01201v1

🚀המאמר היומי של מייק -06.10.24: ⚡🚀
CONTRASTIVE LOCALIZED LANGUAGE-IMAGE PRE-TRAINING

ממשיכים הפסקה בסקירות על מודלי שפה ועוברים לסקירות על מודלים מולטימודליים (שפה ותמונות). טוב, הפסקה למחצה. אתם בטח זוכרים את המודל שנקרא CLIP שעשה הרבה רעש לפני כמה שנים.

CLIP הוא אחד המודלים מולטימודליים הראשוניים שהצליח לייצר אמבדינגס חזקים ומיושרים (aligned) של טקסט ושל תמונות. מיושרים הכוונה של הייצוגים של תמונה וטקסט שמתאר את תוכנה קרובים אחד לשני בזמן שהייצוגים של תמונה וטקסט לא מתאימים רחוקים אחד מהשני (במקרה הזה ביחס למרחק קוסיין ביניהם).

המודל הזה אומן על דאטהסט ענק של תמונות והכותרות שלהם (או טאגים) מהאינטרנט כאשר אימנו אותו תוך שימוש בטכניקה למידה ניגודית (contrastive learning או CL). בגדול מאוד טכניקות CL מאומנות להפיק ייצוג סמנטי מדאטה (מסוגים שונים) כאשר המטרה היא לקרב את הייצוגים (אמבדינגס) של פיסות דאטה קרובות (או חיוביות) ולהרחיק ייצוגים של פיסות דאטה לא דומות (שליליות). במקרה של CLIP פיסות דאטה חיוביות הם הייצוגים של תמונה והכותרת שלה ואילו הזוגות השליליים בנויים מכותבות ותמונות שנבחרו באקראי.

המאמר שנסקור אחד כאמור משכלל את CLIP על ידי הקניה של יכולות לוקליזציה לייצוג. הכוונה כאן שהמחברים מאמנים ייצוגים של תמונה ושל טקסט באופן כזה שבהינתן ייצוג התמונה I וייצוג התיאור של פאץ' ב I המכיל אובייקט מסוים יהיה ניתן להפיק ב"קלות" את מיקום האובייקט בתמונה.

במילים פשוטות נניח שיש לנו אריה עומד ושואג בתמונה הנמצא ב-bounding box (המוגדר על ידי רביעיה של קואורדינטות שלו בתמונה) המסומן ב- B. המחברים מאמנים רשת אנקודר לתמונות f_I רשת אנקודר לטקסט f_T כך שייצוג התמונה R_I ייצוג "אריה עומד ושואג" R_T, המופקים על ידי שני האנקודר האלו (בהתאמה) כך שרשת רדודה יחסית (נקראת prompter במאמר), המקבלת אותם, תוכל לחזות את מיקום האריה B בתמונה. דרך אגב המיקום כאן לא חייב להיות מתואר על ידי bounding box אלא יכול להיות מוגדר (בערך) על ידי כמה ניקודת, תיאור כללי (נגיד חיה, בלי להזכיר שזה אריה) ובעוד צורות.

האימון נעשה כמו בלמידה הניגודית כמו ב-CLIP המקורי. אבל בנוסף ללוס הרגיל שלו יש כאן עוד לוס ניגודי המקרב את ייצוגים של כותרת הפאץ' בתמונה לייצוג המופק על Prompter מייצוג התמונה ומהמתאר של הפאץ' (נגיד BB) ומרחיק את הייצוגים האלו לפאצ'ים שונים. כמובן שה-Prompter גם מאומן תוך כדי,

המאמר משתמש במודלים מאומנים למטרת זיהוי אובייקטים בתמונה (OWLv2) ובמודלים מאומנים אחרים (VeCap) למתן כותרות לפאצ'ים האלו.

מאמר די חמוד וקליל...

https://arxiv.org/pdf/2410.02746

🚀המאמר היומי של מייק -08.10.24: ⚡🚀
CONTEXTUAL DOCUMENT EMBEDDINGS

מזמן לא סקרתי מאמר בנושא של Document Retrieval או DG. למעשה DG מהווה שלב של Retrieval Augmented Generated או RAG שמטרתו היא לאתר את המסמכים הרלוונטיים מסט המסמכים D. בדרך כלל זה נעשה על סמך קירוב של האמבדינגס(הנמדד על ידי מרחק קוסיין) של המסמכים ושל השאלה המופקים על מודל שפה כלשהו.

יש שכלולים למעטים לשיטה הזו, למשל לחלק כל מספר לצ'אנקים ומשתמשים בייצוג שלהם לחישוב הקרבה. יצא לא מזמן מאמר שהציע להוסיף תמצות לכל מסמך וכמובן קיימות עשרות או מאות אחרות.

אם יש בידינו דאטהסט של זוגות D_T המורכבים מ- (שאלה, מסמך רלוונטי) אנו יכולים לעשות פיינטיון לאמבדינגס כאלו, כלומר לאמו שני מודלים: הראשון לחישוב אמבדינג של המסמכים והשני לחישוב אמבדינג של השאלה. בד"כ זה נעשה עם למידה ניגודית שמאומנת לקרב את ייצוגי של כל השאלה לייצוג המסמך הרלוונטי לו ומרחיקה אותו מכל מהייצוגים של שאר מסמכים.

המאמר מציע שיטה שמשפרת את התהליך הזה על ידי הוספת קונטקסט לייצוגים (=אמבדינגס) האלו. אם יש לנו מסמך שניתן לשייך אותו לכמה תחומים (=דומיינים) אנו רוצים שהאמבדינג של המסמכים ישתנה בהתאם בדומיין של השאלות. כלומר אם השאלות צפויות להיות מהדומיין של רפואה אנו רוצים שהאמבדינגס ישקפו את האספקטים הרפואיים ועבור דומיין הספורט שיהיה יותר "מכוון" לספורט. כלומר אנו צריכים כאן contextualized embedding בתלות בשאלות מ-D_T ובסט המסמכים D בעצמו.

המאמר בוחר לעשות זאת על ידי אימון מודלי embedding למסמך או לטקסט בצורה הבאה. קודם כל אנו מחלקים את D לכמה קלסטרים לפי דומיינים (עם מודל embedding התחלתי). לאחר מכן המחברים ממקסמים את סכומי הלוסים הניגודיים על פני כל הקלסטרים האלו. כלומר אנו רוצים לבנות אמבדינג של שאלה ושל המסמך כך ש:

"אמבדינג של השאלה ושל המסמך הרלוונטי לה יהיו קרובים אחד לשני בתוך כל קלסטר (המדמה דומיין) ואילו ייצוג של השאלה יהיה רחוק מהכל המסמכים האחרים בקלסטר".

כלומר אנו מתאימים את האמבדינגס כפונקציה של דומיין השאלות. המאמר גם מציע שיטה לבניה של באצ'ים (ככה מאמנים רשתות היום) כך שהרשת תלמד על שילובי המסמכים הקשים ביותר(למשל מסמכים דומים סמנטית אבל מדומיינים שונים).

בנוסף המאמר מציע לשלב את ייצוגי המסמכים לבנייה אמבדינג של מסמך נתון 'D. כלומר ייצוג של מסמך 'D מורכב משרשור של ייצוגי כל המסמכים מהדאטהסט ואמבדינגס של כל הטוקנים מ 'D (שהם תלויי הקשר המסמך כמובן). בהמשך מאמנים אנקודר למסמך בצורה דומה למה שתואר לפני אבל עם כמה טריקים לייעול האימון.

אציין שהמאמר לא כתוב בצורה מאוד ברורה....

https://arxiv.org/abs/2410.02525

🚀המאמר היומי של מייק -10.10.24: ⚡🚀
DIFFERENTIAL TRANSFORMER

המאמר הזה עשה הרבה גלים ביומיים האחרונים וזו הסיבה שבחרתי אותו לסקירה היומית שלי. המאמר החזיר אותי 3-4 שנים אחורה לתקופה שבה על בסיס ימי יצאו מאמרים המציעים שכלולים שונים לליבה של הטרנספורמרים כה אהובים עלינו. כמובן אני מתכוון למנגנון ה-attention שמאפשר לנו לכמת קשרים בין הטוקנים השונים בטקסט.

המחברים הציעו להחליף את חישוב הסופטמקס הרגיל שיש לנו בטרנספורמרים בהפרש משוקלל (רק הסופטמקס השני משוקלל) של הסופטמקסים. כל סופטמקס מחושב עם מטריצת Q ו-K משלה כאשר המשקול λ של הסופטמקס השני מחושב באופן הבא: $\lambda = \exp(\lambda_{q1} \cdot \lambda_{k1}) - \exp(\lambda_{q2} \cdot \lambda_{k2}) + \lambda_{init}$ כאשר

$\lambda_{q1}, \lambda_{k1}, \lambda_{q2}, \lambda_{k2} \in R^d$ הינם נלמדים ו- $\lambda_{init} = 0.8 - 0.6 \times \exp(-0.3 \cdot (l-1))$, כאשר l זה מספר השכבה (של בלוק הטרנספורמר). אם הנוסחה עבור λ איכשהו מובנת ודי סטנדרטית הנוסחה עבור λ_{init} נותרת בגדר תעלומה (אלא אם כן זה ניסוי ותהיה רגרסיה של הערכים שהתקבלו עם פונקציה מצורה מסוימת).

המאמר טוען לשיפור תוצאת אבל הבדיקות נעשו בעיקר למודלים עם 3B פרמטרים. יש גם טענות לקנסול של רעש כלשהו שאני לא בטוח שאני מבין. בקיצר אני קצת סקפטי, מודה....

https://arxiv.org/abs/2410.05258

🚀המאמר היומי של מייק -11.10.24: ⚡🚀

SELECTIVE ATTENTION IMPROVES TRANSFORMER

היום נסקור מאמר המציג רעיון לשיפור הליבה של הטרנספורמרים, כלומר מנגנון ה-attention. להבדיל מהמאמר של סקרתי(Selective Transformer) הרעיון כאן די ברור לי מתמטית ולא ולא זיהיתי בו נוסחאות מתמטיות "מפתיעות". המאמר של היום מציע שיטה לשיפור ביצועים של הטרנספורמרים ועל הדרך מצליח להקטין את גודל הזכרון הנדרש עבורו.

המחברים טוענים (ובצדק) שלפעמים יש טוקנים שלא כדאי לטרוח ולחשב מקדמי attention עבור זוגות מסוימים של הטוקנים. בנוסף ניתן לדעת את זה על ידי הסתכלות על טוקנים ביניהם ואלו באים לפניהם (ההקשר).

המחברים נותנים את הדוגמא הבאה הממחישה את התופעה הזו. נניח שהטוקנים א, ב, ג הוזנו לטרנספורמר. בשכבה כלשהו עם מ תשומת לב סטנדרטי, טוקן ב מחליט "כמה הוא מעוניין לקחת" מטוקן א (מקדם attention), וטוקן ג יכול להחליט כמה לקרוא מטוקן א, **אבל טוקן ב אינו יכול להשפיע על כמה טוקן ג "לוקח" מטוקן א**. אם טוקן ב קבע שטוקן א אינו רלוונטי או אפילו מטעה לטוקנים עתידיים כמו ג, אין שום דבר שהוא יכול לעשות בשכבה הנתונה כדי לתקן זאת. השיטה המוצעת על ידי המחברים באה לתקן (להקל) את הבעיה הזו.

הרעיון המוצע הוא מאוד אינטואיטיבי ואלגנטי. המחברים מציע להחסיר מווקטור ה-attention (לפני חישוב הסופטמקס) של כל טוקן מטריצת מיסוך נלמדת F. איבר ij במטריצת F (עבור זוג טוקנים i- ו- j) מבטא עד כמה אנו רוצים להקטין את ה-attention בין טוקנים אלו. ערך גבוה של F_{ij} מסמן לנו שהמודל "מאמין" שצריך "להתעלם מהקשר בין טוקן i ל-j כלומר (אם $i > j$; מטריצה F הינה מטריצה קוזלית כלומר $F_{ij} = 0$ אם $i<j$) אפשר לא לדלג על חישוב מקדם ה-attention ביניהם.

אבל מה זה מטריצת F ואיך היא נבנית? גם בצורה מאוד אינטואיטיבית F) הינה שסכום של מטריצות מיסוך רכה S עבור כל הטוקנים בין j ל-i. כלומר טוקן j אינו משפיע על מקדמי מיסוך עבור ה-attention לטוקנים שקודמים ל-j. המחברים לא מסבירים למה הם בחרו לעשות את זה ככה (למישהו יש רעיון?). כמובן מטריצה S הינה אי שלילית (עושים ReLu).

השיטה המוצעת יכולה כאמור לעזור בהאצת האינפרנס על ידי הורדה של טוקנים עם מקדמי F הגדולים ביותר מחישוב ה-attention (לטוקן i נתון). למעשה זה סוג של pruning שהוא תחום מחקר די פעיל ברשתות הנוירונים. המחברים מציעים להגדיר "תקציב attention" לכל שכבה (בלוק של טרנספורמר) ובאופן הדרגתי להעיף מספר קבוע של טוקנים מחישוב ה-attention (נעשה באיטרציות). כל פעם מורידים טוקנים עם ערכי F הגבוהים ביותר ובוחרים שכבה שעבורה הורדה כזו משפיעה באופן המועט ביותר על ה-perplexity (כלומר log-likelihood).

בנוסף כבר במהלך האימון של מטריצות S אנו יכולים לגרום למודל "לבטל" יותר נוירונים על ידי הוספה של איבר לפונקציית הלוס הרגילה שלה(log-likelihood), הקונס את המודל על S בעלת ערכים נמוכים מדי.

יש לי תחושה שהמאמר הזה הוא התחלה של משהו מעניין...

https://arxiv.org/pdf/2410.02703

🚀⚡ המאמר היומי של מייק -12.10.24:⚡🚀

GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models

האם מודלי שפה גדולים מסוגלים לעשות ריזונינג? השאלה הזו מעסיקה חוקרים רבים לאור יכולות די מרשימות שמודלי שפה מפגינים בפתרון שאלות לא פשוטות (אבל רק בתנאים מסוימים 🙂). המאמר בוחן את יכולות החשיבה המתמטית(שזה תת-יכולת של ריזונינג כללי) של LLMs ומציג את GSM-Symbolic, בנצ'מרק חדש לבחינת יכולות אלו שהם פיתחו.

החוקרים מצאו שביצועי LLMs(נבחן מגוון רחב של מודלים: GPT-4, Llama3, Mistral, Phi, Gemma ו-o1) משתנים באופן משמעותי כאשר משנים מעט את השאלות המתמטיות, מה שמעלה ספקות לגבי אמינות המדדים הקיימים. הביצועים של רוב המודלים יורדים כאשר עוברים מ-GSM8K המקורי ל-GSM-Symbolic, מה שמרמז על אפשרות של זיהום דאטה (contamination) במהלך האימון (כתבתי על זה לא מעט).

בנוסף המודלים מראים רגישות גבוהה יותר לשינויים במספרים מאשר לשינויים בשמות עצם, מה שמעיד על חוסר יציבות ביכולות החשיבה שלהם. ככל שמספר המשפטים בשאלה עולה, הביצועים יורדים והשונות בביצועים עולה, מה שמצביע על קושי בטיפול בשאלות מורכבות יותר.

החוקרים יצרו בנצ'מארק GSM-NoOp, שבו נוספו משפטים לא רלוונטיים לשאלות, וגילו ירידה דרמטית בביצועים של כל המודלים. אפילו כאשר ניתנו למודלים דוגמאות של אותה שאלה או שאלות דומות, הם התקשו להתגבר על האתגרים של GSM-NoOp.

המחקר מצא שאימון נוסף על משימות קלות יותר וגם הגדלת כמות דאטה לאימון לא שיפרו את הביצועים במשימות מורכבות יותר.

קצת מנחם שלפחות מודלים חדשים יותר, כמו o1-preview ו-o1-mini, הראו ביצועים חזקים יותר, אך עדיין סבלו מהמגבלות שזוהו במחקר

הממצאים מעלים ספקות לגבי היכולת האמיתית של LLMs לבצע חשיבה מתמטית פורמלית. נראה כי המודלים מסתמכים יותר על התאמת תבניות מאשר על חשיבה לוגית אמיתית. המחקר מדגיש את הצורך בשיטות הערכה אמינות יותר ובמחקר נוסף על יכולות החשיבה של מודלי שפה גדולים.

https://arxiv.org/abs/2410.05229

⚡🚀המאמר היומי של מייק -14.10.24: ⚡🚀

LLMS KNOW MORE THAN THEY SHOW: ON THE IN-TRINSIC REPRESENTATION OF LLM HALLUCINATIONS

מאמר כחול-לבן זה מציג חקירה מקיפה של דפוסי השגיאות של LLMs והקשר שלהם עם הייצוגים הפנימיים של המודל. המחברים מבצעים סדרת ניסויים כדי לנתח כיצד LLMs מקודדים מידע על התשובה הנכונה וחוקרים את טבע השגיאות שהם מייצרים.

המחברים חקרו את הנושאים הבאים:

שיפור זיהוי שגיאות:

המחברים גילו כי ניתן להגיד האם המודל ייתן תשובה נכונה או לא מהסתכלות בטוקנים ספציפיים המכילים "תשובה מדויקת" בתוך פלט המודל. כלומר עבור השאלה "מה עיר הבירה של צרפת" האינדיקציה האם המודל נותן התשובה הנכונה ניתן לגזור מייצוגי הטוקנים המופק על ידי שכבות מסוימות של המודל. על ידי התמקדות בטוקנים אלה, המחברים הצליחו לשפר משמעותית את דיוק זיהוי השגיאות במגוון משימות ומודלים.

הכללה בין משימות:

המחקר בוחן האם יכולות זיהוי השגיאות ניתן להכללה בין משימות וסוגי דאטה שונים. התוצאות מראות הכללה מוגבלת, עם הצלחה מסוימת רק בין משימות הדורשות מיומנויות דומות (למשל, אחזור עובדתי או היסק שכל ישר). זה מרמז על כך של-LLMs יש מספר מנגנוני אמיתות "ספציפיים למיומנות" ולא מנגנון אוניברסלי אחד.

טקסונומיה של שגיאות:

המחברים מציעים טקסונומיה של שגיאות LLM המבוססת על התפלגות התשובות במספר דגימות. הם מזהים מספר סוגי שגיאות, כולל תשובות נכונות/שגויות באופן עקבי, תשובות נכונות/לא נכונות לסירוגין ומקרים עם תשובות מגוונות רבות. המחברים מדגימים שניתן לחזות סוגי שגיאות אלה מהייצוגים הפנימיים של המודל.

פער בין ייצוג פנימי להתנהגות חיצונית:

המחברים מראים פער זה באמצעות מערך ניסויי בו הם מייצרים מספר תשובות לכל שאלה ומשתמשים במודל מאומן(באמצעות probing) לבחירת התשובה הטובה ביותר על סמך ייצוגים פנימיים. הם הבחינו בשיפורים משמעותיים בדיוק עבור סוגי שגיאות מסוימים, במיוחד אלה בהם המודל אינו מראה העדפה ברורה לתשובה הנכונה בפלטים הרגילים שלו. לדוגמה, בקטגורית שגיאות "שגוי באופן עקבי אך מייצר את התשובה הנכונה לפחות פעם אחת", שיטת הבחירה מבוססת מודל הסיווג השיגה שיפורים של עד 40% בדיוק בהשוואה למצב הרגיל.

ממצא זה מרמז על כך שה-LLMs לעתים קרובות "יודעים" את התשובה הנכונה ברמה מסוימת, אך ידע זה לא תמיד משתקף בתהליך יצירת הפלט שלהם. פער זה מעלה שאלות חשובות לגבי טבע ייצוג הידע ב-LLMs והמנגנונים השולטים בתהליך יצירת הפלט שלהם. המחברים מציעים כי ממצא זה עשוי לשמש לפיתוח אסטרטגיות חדשות לשיפור דיוק ה-LLM, אולי על ידי שינוי תהליך יצירת הפלט כך שלוקח בחשבון גם את הייצוגים הפנימיים.

https://arxiv.org/abs/2410.02707

⚡🚀המאמר היומי של מייק -15.10.24: 🚀⚡
EFFICIENT DICTIONARY LEARNING WITH SWITCH SPARSE AUTOENCODERS

היום סוקרים מאמר קליל המשלב שני רעיונות די נחמדים שמשמים LLMs (במיוחד לאחרונה) והאמת השילוב שלהם נראה די טבעי. הרעיון הראשון הינו Mixture of Experts או MoE בקצרה.

MoE היא שיטה המאפשרת לנו להקל על האינפרנס על ידי שימוש רק בחלק ממשקלי המודל. בד״כ מטריצות משקלים ברשת feed-forward (יש שם 2 שכבות בסך הכל) בבלוק הטרנספורמרים (אחרי attention) מחוקלים לכמה קבוצות שכל אחת מהן נקראת מומחה או expert. באינפרנס המודל משתמש רק בחלק (לפעמים רק אחד) מהמומחים ובכך הוא מוריד את מחירו של האינפרנס. כלומר אותו המודל מופעל בצורה קצת שונה בהתאם לקלט (בנוסף ל-attention),

הקונספט השני הוא Sparse AutoEncoders או SAE בקצרה שהפך להיות די פופולרי אחרי החוקרים של אנטרופיק הציעו להשתמש בו למטרת חקר interpretability של מודלי שפה. לפני הבלוג הזה הסברה הרווחת (סוג של) היתה שבמודל שפה יש נוירונים שנדלקים חזק (מקבלים ערך גבוה) על קונספטים מסוימים כאשר כל נוירון כזה הינו מונו-סמנטי כלומר יש קונספט אחד בלבד שהוא ״אחראי״ עליו.

לעומת זאת החוקרים של אנטרופיק הציע להתבונן בכל נוירון כפולי-סמנטי כלומר ״אחראי״ על מספר קונספטים לא קשורים. לפי משנתם ניתן לגלות את הקונספטים האלו באמצעות SAE שבונה autoencoder דליל (הרוב אפסים) במימד גבוה הרבה יותר מגודל השכבה שבה נמצאים הנוירונים הפוליסמנטיים אלו. SAE כאן מורכב משתי שכבות בלבד, אחת לאנקודר ואחת לדקודר.

כאן כל רכיב שהוא לא אפס בווקטור אחרי שכבת ה-encoder של SAE הוא אחראי על קונספט מסוים כלומר מהווה נוירון מונוסמנטי. כך יוצא שכל נוירון בשכבה המקורית הוא שילוב לינארים של הנוירונים המונוסמנטיים אלו. SAE מאומן בצורה די סטנדרטית עם איבר רגולריזציה שאוכף את דלילות הייצוג אחרי האנקודר.

אז המאמר מציע לשלב את שני הקונספטים האלו כך שכל נוירון הוא צירוף לינארי אחר של הנוירונים המונוסמנטיים בשכבת ה-encoder. זה מאפשר גמישות נוספת ביחס לרעיון המקורי ובטח מאפשר לגלות קונספטים שונים המוסתרים בתוך ה-LLMs שלנו.

מאמר קליל - ממליץ להעיף מבט

https://arxiv.org/abs/2410.08201

🚀⚡ :המאמר היומי של מייק -16.10.24🚀⚡
EFFICIENT REINFORCEMENT LEARNING WITH LARGE LANGUAGE MODEL PRIORS

היום נסקור מאמר שהוא נראה די כבד מתמטית (הרבה נוסחאות ומלל שנראה מתמטי) אבל הרעיון מאחוריו הוא די פשוט וקל להסבר. אנחנו אוהבים למנף את עוצמתם של מודלי שפה למשימות רבות (ולא תמיד לכאלו שהם מסוגלים לבצע כמו שצריך לפחות כרגע).

המאמר מציע להשתמש במודל שפה כפריור עבור סוכנים במשימות בהם הם צריכים לבצע SDM או sequential decision making. המאמר נותן בתור דוגמא משחק overcooked כאשר הסוכן צריך לבצע משימות בישול שונות בהתבסס על מצב המטבח שבו הוא מבשל אותם. המטרה של הסוכן היא לחזות את הפעולה הבא (באמצעות תיאור טקסטואלי) כאשר התגמול הוא ביצוע נכון של המשימה (הכנה של מנה לפי המתכון :)).

כאמור המטרה כאן היא לחזות את הפעולה הבאה עבור הסוכן (המתוארת) על ידי הטקסט כאשר המצב (state) גם מתואר על ידי טקסט. בגדול מאוד אנו מתחילים ממודל אחד (הפריור P) עבור חיזוי המצב הבא (מהמצב הקודם והפעולה) ועבור חיזוי הפעולה הבאה בהינתן המצב (מתואר על ידי התפלגות Q_h). המטרה כאן היא ללמוד את Q_h כאשר ממקסמת התגמול הצפוי ושומרת את התפלגות Q קרובה לפריור P (זוכרים PPO שהתפרסם מאוד לפני שנתיים כאשר OpenAI השתמשו בו ל-RLHF לאימון מודלי שפה). המרחק כמובן ניתן על ידי ה-KL 🙂

אז הפעולה הבאה a_t (כלומר גנרוט התיאור הטקסטואלי שלה) מתבצע באופן הבא. דוגמים כמה גרסאות של a_t עם P מחשבים את הנראות שלהם לפי Q הנלמד, מנרמלים עם הסופטמקס ודוגמים את הפעולה הבאה כאשר מטרת התהליך מקסום של התגמול הצפוי (עם הרגולריזציה שהסברנו עליה קודם).

כמובן שניתן לעשות את זה בכמה אופנים: בצורה של online דרך שערוך של פונקציית Q של הזוג (מצב, פעולה) כאשר פונקציית Q קשורה להתפלגות Q_h של הפעולה הבא שנידונה בפסקה הקודמת (עניין של נרמול נכון). ניתן לעשות את זה גם באמצעות offline עם איזה פוליסי טוב ידוע של המומחים כאשר המטרה היא גם שערוך של פונקציית Q שבאמצעותה ניתן לשערך (לקבל) את Q_h עבור חיזוי הפעולה הבא. ניתן לעשות את זה גם באמצעות שיטה דומה ל-PPO אבל בכל המקרים הפריור הוא ההתפלגות המושרית על ידי מודל שפה נתון.

מאמר מעניין בקיצור...

https://arxiv.org/pdf/2410.07927

⚡🚀המאמר היומי של מייק -17.10.24: 🚀⚡
EQUIVARIANT CONTRASTIVE LEARNING

היום נסקור מאמר שפורסם לפני שנתיים וחצי בנושא למידה ניגודית (contrastive learning). הנושא עצמו תמיד עניין אותי וסקרתי לא מעט מאמרים אבל חייב להגיד שבזמן האחרון שטף המאמרים על CL די נחלש. כאמור המאמר הזה שראה אור לפני שנתיים מציע שכלול לשיטה הקלאסית לבנייה של ייצוג דאטה (אמבדינג) באמצעות CL.

בגדול CL היא שיטה לבניית ייצוג של דאטה כאשר העיקרון המוביל הוא לקרב ייצוגי פיסות דאטה דומות(זוגות חיוביים) ולהרחיק ייצוגים של פיסות דאטה לא דומות (שליליים). זוגות דוגמאות חיוביים (במקרה של דאטה לא מתויג) נבחרות כאוגמנטציות שונות של דוגמא (עבור תמונות זה יכול להיות הזזה, סיבוב וכדומה) ואילו זוגות השליליים נבחרים באקראי מהדאטהסט.

אולם יש לא מעט בעיות עם הגישה הזו הקשורות לבחירת זוגות של דוגמאות חיוביות - למשל שני פאצ'ים באותה התמונה עלולים להכיל תוכן סמנטי שונה שלא נרצה לקרב את ייצוגיהם (הוצעו מספר פתרונות לסוגיה זו בעבר וחלקן סקרתי). בנוסף אולי היינו רוצים לקבל ייצוגים שונים (ולא מאוד קרובים) של טרנספורמציות מסוימות של אותה התמונה (נגיד סיבוב או הזזה) למשימת downstream ספציפית.

כלומר היינו רוצים להשרות יחס נתון T_i בין ייצוגי התמונה ההתחלתית I ולייצוג התמונה אחרי טרנספורמציה T (נקרא לה I_T). כלומר אנו רוצים לבנות ייצוג p כך ש:

$$p(T(I)) = I\_T(p(I))$$

וזה בדיוק מה שנקרא equivariance. למעשה CL הסטנדרטי הוא מקרה פרטי של equivariance שעבורן T_i הינה טרנספורמצית זהות וזה נקרא אינווריאנטיות של הייצוג תחת טרנספורמציית T.

וזה בדיוק מה שהמאמר עושה. למעשה המחברים מציעים לאמן ייצוג ששומר על אינווריאנטיות עבור טרנספורמציות מסוימות (כמו בCL הסטנדרטי) ו אוכף בנוסף equivariance מוגדר לטרנספורמציות מקבוצה נתונה G המתאימה למשימת downstream שיש לנו ביד. כלומר לכל טרנספורמציה מ-G אנו מגדירים מראש את הטרנספורמציה ה-equivariant שלה (שיכולה להיות חברה ב-G גם כן) ומאמנים את הייצוג כך שהיחס ה-equivariance ביניהם יתקיים. מבחינה פרקטית הלוס הוא סכום משוקלל של הלוסים של CL הסטנדרטי ו ה-ECL.

מאמר חמוד - מחר או היום בערב אסקור את מאמר ההמשך שלו...

https://arxiv.org/abs/2111.00899

⚡🚀המאמר היומי של מייק -18.10.24: ⚡🚀

SimCSE: Simple Contrastive Learning of Sentence Embeddings

סקירה קצרה מאוד על איך ניתן לעשות למידה ניגודית (contrastive learning) כדי לבנות ייצוג חזק של הטקסט. הרי כבר הסברנו בסקירה הקודמת שהמטרה של CL היא לאמן ייצוג של דאטה כך שייצוגים קרובים סמנטית יהיו קרובים במרחב הייצוג ואילו ייצוגים של דוגמאות לא דומות יהיו רחוקות שם. מאמנים ייצוג כזה בדרך כלל דרך מזעור היחס שבין ייצוגי פיסות דאטה דומות (זוג חיובי) לבין אלו של הלא דומות (שליליים).

השאלה איך לבנות את הייצוגים האלו (במיוחד הזוגות החיוביים)? זה בעצם נושא מחקר פעיל מלפני שנתיים-שלוש. המאמר המסוקר מציע לבנות זוגות חיובים דרך dropouts שונים של רשת הנוירונים (שאותה מאמנים לבנות את הייצוג). כלומר עבור אותו הטקסט זוג דוגמאות חיובי נבנה עם עם הפעלת הרשת עליו עם שני dropouts שונים. נזכיר dropout מבטל באקראי קשרים בין נוירונים ברשת ומהווה כלי ידוע לשיפור יכולת ההכללה של הרשת. הזוגות השליליים נבנים עם דוגמאות שנבחרו בצורה אקראית.

לדאטהסטים המכיל משפטים מתויגים כמו למשל NLI (למשפט נתון הדאטהסט מכיל משפט אחד עם אותה המשמעות(entailment), משפט אחד בעל משמעות דומה ומשפט אחד בעל משמעות הפוכה או סתירה - contrary). באופן לא מפתיע המאמר מציע לבחור בתור זוג שלילי את שני המשפטים בעלי משמעות הפוכה ובתור זוג חיובי שניים עם אותה משמעות.

בנוסף המשפט הזכיר לי לייצוג דאטה טוב יש 2 תכונות מהותיות: קרבה בין ייצוגי הדאטה הדומה והתפלגות יוניפורמית של כלל הייצוגים של הדאטה - זה חשוב.

https://arxiv.org/pdf/2104.08821

⚡🚀המאמר היומי של מייק -19.10.24: ⚡🚀

DiffCSE: Difference-based Contrastive Learning for Sentence Embeddings

סקירה קצרה ואחרונה(כנראה) במיני-סדרה על איך לבנות ייצוג דאטה באמצעות שיטות למידה ניגודית. כבר הסברתי על הלמידה הניגודית בשתי בסקירות הקודמות. בקצרה, מאמנים מודל הבונה אמבדינג לדאטה המקרב ייצוגים של פיסות דאטה דומות ולהרחיק פיסות דאטה לא דומות. וכאמור הוצעו עשרות שיטות לעשות זאת לדאטה מדומיינים שונים.

המאמר מציע שיטת CL העושה זאת בצורה מתוחכמת יותר (לטעמי). הרי אחת המטרות של בניית ייצוג הדאטה היא שהוא ישקף את התכונות האינהרנטיות של הדאטה והמחברים הציעו דרך "לאכוף" את זה על הייצוג. הם

מאמנים מודל לבניית ייצוג טקסט כך שהמודל "יבדיל בין מה אמור ומה לא אמור להיות בתוך הטקסט".

איך הם עשו זאת? הם מיסכו כמה טוקנים בטקסט, ביקשו ממודל אחר לחזות את הטוקן הזה ואז אימנו את ייצוג כך שבעזרתו יהיה ניתן להבדיל בין הטוקנים שנחזו ואלו שלא. כלומר בנוסף למודל החיזוי (לא אומן) ומודל לבניית אמבדינג הם אימנו עוד מודל לסיווג בינארי שמטרתו להגיד האם טוקן נחזה או לא. וייצוג הטקסט מוזן למודל הסיווג הזה.

דרך אגב פונקציית הלוס למודל הסיווג דומה לזו של GAN אבל אין באמת קשר בין שני הדברים (זה טיפה בלבל אותי בהתחלה)....

https://arxiv.org/pdf/2204.10298

⚡🚀המאמר היומי של מייק -20.10.24: 🚀⚡
RL, BUT DON'T DO ANYTHING I WOULDN'T DO

אוקיי, אחרי כמה סקירות יחסית קלילות הגיע הזמן לסקור מאמר קצת כבד לפחות מהמבט הראשון. המאמר בנושא של אימון מודלי שפה עם השיטות מעולם למידה באמצעות חיזוקים או בקצרה RL. דרך אגב הטענות המתמטיות הרבות שהמחברים הוכיחו (לא אמפירית אלא הוכחות מתמטיות רציניות) לא מוגבלות רק לאימון LLMs עם RLHF.

בד"כ כאשר אנו מאמנים LLM על RLHF פונקציית הלוס שאנו מעוניינים לאפטם מורכבת מסכום של שני איברים (לפעמים מוסיפים עוד אבל אני מדבר כאן על כאלו המופיעים ברוב המאמרים על יישור (alignment) של LLMs עם שיטות RL.

האיבר הראשון אחראי על מקסום של פונקציית reward שזה מודל שמאומן לפני על דאטהסט המכיל זוגות של תשובות מועדפות יותר ומועדפות פחות(המתויג על ידי בני אדם) לסט של שאלות. מודל reward מאומן לתת ערך גבוה לתשובה טובה וערך נמוך לתשובה לא טובה לשאלה. אז האיבר הראשון מאפטם את משקלי ה-LLM המאומן כך שימקסמו את פונקציית ה-reward ובכך יגרמו ל-LLM להיות יותר מיושר (aligned)עם הציפיות שלנו (לפחות היינו רוצים להאמין בכך).

האיבר השני הינו איבר רגולריזציה השומר את המשקלים של המודל המאומן קרובים יחסית (במונחי מרחק KL בין התפלגויות הטוקנים) למשקלי המודל ההתחלתי (שלו אנו עושים פיין טיון). איבר זה נדרש כי בלעדיו המודל יעשה את מה שנקרא "reward hacking" ובמקום להתיישר עם ציפיותנו ימקסם את reward אבל כתוצאה נקבל עוד יותר גרוע ממה שהיה (או לפחות פחות טוב ממה שניתן לקבל עם איבר רגולריזציה זה).

אולם מחברים המאמר טוענים שאיבר זה לא מספיק ולא תמיד ימנע ממשקלי המודל להתכנס למצבים לא רצוים. הסיבה לכך היא שמודל בסיס שאנו רוצים לשמור את המודל המאומן קרוב אליו מהווה בעצמו קירוב של מודל "בטוח ומיושר עם ציפיותנו" (בדרך כלל אומן על התשובות הרצויות). ומתברר שגם אם מודל הבסיס שלנו קרוב מספיק ל"מודל הבטוח" והמודל שאנו מאמנים קרוב למודל הבסיס במונחי KL, עדיין לא ניתן להבטיח שהמודל המאומן יהיה קרוב מספיק ל"מודל הבטוח" (גם במונחי KL) - כלומר אי שוויון המשולש לא מתקיים כאן. גם נאמן את מודל בסיס על יותר דאטה ויותר משאבים, עדיין נתקשה להבטיח את קרבתו של המודל המאומן ל"מודל הבטוח"

הסיבה לכך היא קצת (מבחינה קונצפטואלית) דומה לכך למה דגימת Langevin בצורתה הקלאסית (ללא רעש) לא עובדות לדאטה בעלת מימד גבוה מאוד כמו (תמונות). בגלל המימד המאוד גבוה של המרחב הסמנטי של הדאטה מודל הבסיס יגיע למקומות ש"המודל הבטוח" לא היה מגיע בכלל ואז הוא יתקשה לתת שערוך אמין

להסתברויות הטוקנים. וזה יגרום למודל המאומן להיות לא אמין באותה המידה.

המחברים קוראים למאורעות אלו (הגעה למצב שהמודל הבטוח לא היה מגיע אליו) מצבי חסר תקדים (unprecedented events) וטוען שכאשר הם קורים מודל הבסיס ייטה לתת תשובה "פשוטה מדי" ולרוב לא נכונה וכך יעשה המודל המאומן. הפשטות הזו נובעת כנראה (איליה סלוצקבר מדבר על זה רבות) בגלל ה algorithmic-information-theoretic inductive bias שעוזר לרשתות נוירונים להגיע ליכולת הכללה טובה עקב נטייתם להתכנס לפתרונות פשוטים בעלות סיבוכיות תכנותית נמוכה (שזה בעצם סיבוכיות kolmogorov) כלומר הן פעולות לפי עקרון התער של אוקם (זו ההנחה כמובן). זה גורם למודלים להפגין התנהגות פשוטה(מדי) ולא טובים במקרה שהם נתקלים במאורעות חסרי תקדים האלו. והמודל המאומן על RLHF "יורש מהם" את הפגם זה.

מאמר מאוד עמוק, דורש זמן בשביל להפנים אבל שווה קריאה בהחלט...

https://arxiv.org/abs/2410.06213

⚡🚀המאמר היומי של מייק -22.10.24: ⚡🚀
Sample what you can't compress

לא היה לי הרבה זמן להקדיש לסקירה אז בחרתי במאמר הזה שניתן לסקור אותו די בלקוניות בלי לפגוע בחוויית הקוראים. המאמר מציע שיטה נחמדה לבניית ייצוג דאטה ויזואלי (קרי תמונות) באמצעות שכלול של אוטו-אנקודר. מכיוון שהייצוג הזה בד"כ במימד נמוך יותר מהדאטה עצמו אז ניתן להתייחס אליו בתור דחיסה של דאטה. ד"א ניתן לאמן ייצוגים שלאו דווקא "מעבירים" את הדאטה למרחב בעל מימד נמוך יותר ב-denoising AE ולפעמים ב-sparse AE.

אוטו-אנקודר זו דרך לבנות ייצוג מקומפרס של דאטה עם השילוב של האנקודר והדקודר כאשר האנקודר ממפה את הדאטה למרחב הייצוג והקודר משחזר את הדאטה המקורי מייצוגו הדחוס. מאמנים AE דרך מזעור של לוס השחזור (עד כמה טוב הצלחנו לשחזר את הדאטה מייצוגו הלטנטי) ולפעמים מוסיפים רגולריזציה במטרה לגרום לייצוג להיות בעל תכונות מסוימות (כגון דליל).

כמובן שלא תמיד מצליחים להגיע לייצוג חזק (ששומר את כל התכונות האינהרנטיות של פיסת דאטה) עם AE והמחברים מציעים לשכלל אותו על ידי הוספתו של מודל הדיפוזיה לסיפור. כזכור (או שלא ואז אני אזכיר) מודל דיפוזיה מאומן להסיר רעש מפיסת דאטה ואם מאמנים אותו טוב אז מקבלים מודל שיודע לגנרט דאטה מרעש טהור(על ידי הסרת רעש הדרגתית).

המחברים מציע לקחת את מודל הדיפוזיה (המחברים משתמשים במודל דיפוזיה המקורי שבונה את התמונה עצמה בתהליך דיפוזיה ולא ייצוגה הלטנטי). המודל הזה מורכב מסדרת של U-Nets (ולא טרנספורמרים כמו שאנו רואים היום במודלי דיפוזיה) שקודם מקטינים את מימד התמונה (כלומר ניתן לראות את זה כאוטו-אנקודר) ולאחר מכן בונים מהייצוג הזה את התמונה.

המחברים מזינים את התמונה המשוחזרת אחרי הדקודר של AE יחד עם התמונה המורעשת(המקורית) למודל דיפוזיה שמאומן כאמור להסיר רעש מהדאטה (יחד עם AE). הלוס מורכב מסכום משוקלל של הלוס הרגיל של מודל הדיפוזיה, הלוס הרגיל והלוס ה-perceptual ששניהם מופעלים לתמונה המשוחזרת אחרי השלב הראשון של ה-AE (לפני מודל הדיפוזיה). הלוס ה-perceptual בודק עד כמה התמונה המשוחזרת נראית "טבעית למבט האנושי" (משווים את האקטיבציות שלה ברשת מאומנת עם אלו של התמונות הטבעיות).

הייצוג הסופי של פיסת דאטה מתקבל אחרי ה"אנקודר" של מודל דיפוזיה (ה-bottleneck). וכמובן יש טענות

לדחיסה טובה יותר משיטות SOTA עם הגישה המוצעת...

https://arxiv.org/abs/2409.02529

⚡🚀המאמר היומי של מייק -23.10.24: ⚡🚀

Predicting from Strings: Language Model Embeddings for Bayesian Optimization

המאמר מהסוג שנסקור היום אני לא סוקר בדר״כ - אולי מתוך 300 מאמרים שסקרתי יש 1-2 כאלו (לא בטוח). לא בגלל שהנושא לא מעניין אלא שיש פחות מאמרים בו והוא נחשב פחות ״באזזי״ למרות חשיבותי. כמו שמשתמע משם המאמר הנושא הוא אופטימיזציה בייסיאנית.

בגדול אופטימיזציה בייסיאנית היא אחד הכלים לפתרון בעיות תכנון ניסוים ולמה שנקרא black-box optimization כאשר היא למעזר את מחיר של תהליך החיפוש הפתרון הממקסם פונקציית המטרה. פונקציית המטרה יכולה להיות יעילות התרופה (כאשר המטרה למצוא את הרכבה האופטימלי) או אופטימיזציה של הייפר-פרמטרים של רשת גדולה. בשני המקרים כל אבלואציה של פונקציית המטרה הינה יקרה מאוד ויש צורך למזער את כמות הפעמים שמחשבים אותה (לבדיקה הרכב של תרופה או אבלואציה של ביצועים עבור שילוב הייפר-פרמטרים מסוים של הרשת).

קיימות לא מעט שיטות לאפטם את בחירת הנקודות x לאבלואציה של פונקציית המטרה שמצד אחד בוחרת איזורים בהם לא בדקנו (exploration) ומצד שני גם מנצלת את הידע שלנו על ערכי פונקציית המטרה באיזורים שכבר ביקרנו (exploitation) במטרה למצוא נקודת מקסימום טובה במאמץ מינימלי. רוב השיטות מנסות לבנות מה שנקרא surrogate objective או פונקציית מטרה דמה הזולה להפעלה כדי למצוא את x הבא בהינתן תוצאות הפעלה הקודמות (כלומר זוגות x ו- y=f(x)). הדרך הפופולרית ביותר היא להשתמש בתהליכי גאוס כדי למדל את פונקציית מטרה דמה ובעזרתה בוחרים את ה-x האופטימלי.

המאמר מציע לרתום את ה-LLMs לסיפור הזה במטרה לשערך את התוחלת ואת השונות של f(x) עבור x נתון. בשלב הראשון הופכים את הזוגות של x ו-y הידועים לפורמט של string (נגיד לjson המכיל את שמות הפיצ'רים והערכים שלהם). לאחרי מכן מזינים אותם לאנקודר מבוסס LLMs המפיק את ייצוגי הזוגות האלו. בשלב האחרון מכניסים את ייצוגים אלו לדקודר כדי יחד עם הערך של x שעבורו אנו רוצים לחשב את f(x) (תוחלת ושונות). מאמנים את הדקודר (האנקודר לא מאומן) על סדרות ״זהב״ של זוגות x ו- f(x) למספר משימות שונות. במהלך האימון בהינתן k הזוגות הראשונים מנסים לחזות את ערך הפונקציה עבור x_k+1 ל k-ים שונים.

מעניין שהמאמר מניח כי את באינפרנס ערכי ה- x-ים לבדיקה מתקבלים דרך איזה אלגוריתם אבולוציוני נתון.

https://arxiv.org/pdf/2410.10190

⚡🚀המאמר היומי של מייק -24.10.24: ⚡🚀

HOW MANY VAN GOGHS DOES IT TAKE TO VAN GOGH? FINDING THE IMITATION THRESHOLD

מאמר מעניין שנטלו בו חלק חוקרים ישראלים מאוניברסיטת בר-אילן. הם חקרו נושא די חשוב שקשור להפרת זכויות יוצרים אפשרית על ידי מודלים גנרטיביים לתמונות. הרי יש מודלים שאומנו בחלקם על דאטה שהוא פרטי, מוגן על ידי זכויות יוצרים ואם המודל יתחיל לגנרט לנו תמונות דומות מדי להם זה עלול להוות עבירה על החוק. אבל איך להבטיח (או לפחות לתת הערכה כלשהי) לכך שזה לא יקרה?

המאמר בחר בגישה די אינטואיטיבית לכך. הרי כישורי העתקה של קונספט מסוים על ידי המודל קשורים קשר סיבתי (אמנם לא ב 100% מובן כרגע) במספר פיסות דאטה (= תמונות) המוכלות בדאטהסט שהמודל אומן עליו. אבל איך נדע זאת? הרי אז נצטרך לאמן הרבה מודלים כדי לבדוק מתי התמונות המגונרטות על ידי המודל יהיו דומות מדי קונספט T מסוים (עם פרומפט מתאים).

כמובן שזה לא בר עשייה והמאמר מציע שיטה יחסית פשוטה לעשות את זה כאשר הוא מניח הנחה מהותית אחת: מספר התמונות המכיל קונספט T מספיק לכך שהמודל יהיה מסוגל להעתיקו איננו תלוי ב-T. אני מניח שזה נכון בגבולות הסביר זאת אומרת המספר הזה נע באינטרוול יחסית צר לכל הסגנונות. יש עוד הנחה שניה (גם חשובה) שאין איזה confounded בין מספר התמונות ליכולת המודל להעתקה (גם די סביר).

עם הנחה כזו המאמר מציע לאמן מודל על הדאטהסט שיש בו שונות גדולה בין כמות ההופעות של כל קונספט. לאחר מכן המאמר מגנרט תמונות מכל T שהופיע בטקסט ובודק כמה מהם קרובים סמנטית (משווים אמבדינגס) ל T. זה נעשה עם הסף שנקבע דרך השוואה בין דמיון האמבדינגס של תמונות שונות של אותו הקונספט מול תמונות של מכילות את הקונספט הזה (כדי למזער FP יחד FN).

לאחר מכן מגנרטים תמונות עבור כל הקונספטים T שיש בדאטהסט ומחשבים כמה מהם (היחס) מכילים את T. זה נקרא imitation score. בסוף אנו נקבל imitation score עבור כל קונספט T ובגלל שיש לנו שונות גדולה בין הופעה של כל קונספט בדאטהסט ניתן לזהות איפה יש עלייה מובהקת ב- score הזה מבחינת מספר ההופעות של קונספט T בתמונה. זה קצת דומה לזיהוי elbow ב-k-means ויש אלגוריתמים מעולים (כמו PELT) שיודעים לעשות זאת. ככה נקבל את הסף של מספר ההופעות של קונספט בדאטהסט שממנו המודל יידע להעתיקו ופוטנציאלית לגרום לתביעות.

אהבתי - המאמר גם כתוב יפה וברור.

https://arxiv.org/pdf/2410.15002

⚡🚀המאמר היומי של מייק -25.10.24: 🚀⚡

Amortized Planning with Large-Scale Transformers: A Case Study on Chess

מאמר די מעניין שגרם לדיונים רבים בנושא יכולות ריזונינג של מודלי שפה. אחרי שהעניינים קצת נרגעו הגעתי לסקורו בלי להתייחס יותר מדי לסוגיה הזו. המאמר למעשה אימן מודל שפה די צנוע מבחינת פרמטרים (עם הטרנספורמרים בפנים) לשחק שח. אזכיר שהמכונות הגיעו לרמת של בני אנוש בשחמט די מזמן (לדעתי לפני 30 שנה כאשר deep blue השאיר אבק לאלוף העולם דאז גארי קספרוב).

אז מה המחברים עשו בעצם? הם הורדו 10 מיליון משחק שחמט מאתר LiChess והשתמשו בכלי הנקרא StockFish לשערוך הסתברות ניצחון עבור מצב לוח נתון s. לאחר מכן הם הפכו את מצב הלוח ותיאור המהלך לטקסט (נראה די טבעי בסך הכל) ואימנו מודל שפה "לשחק שח". המחברים ניסו לעשות זאת בכמה דרכים:

1. אימנו את המודל לחזות את הסיכוי לניצחון בהינתן מצב הלוח s ומהלך a. כדי לעשות זאת הם חילקו סיכויי הניצחון לכמה בינים (זרים) ואימנו את המודל לחזות את הבין שבו נמצא הסיכוי ה-ground-truth. הם עשו את זה לא בצורה הרגילה (עם one-hot encoding של כל בין) אלא על ידי "ריכוכו" כלומר כל בין מקבל הסתברות משלו כאשר הבין ה-GD מקבל את ההסתברות הכי גבוה (נעשה לפי התפלגות גאוס ונקרא HL-Gauss)
2. אימנו את המודל את סיכוי הניצחון עבור מצב לוח נתון s באותה הצורה כמו ב 1.
3. אימון מודל לחזות את המהלך ה-GD של המשחק

בסוף המהלך נבחר כזה עם סיכוי לניצחון הגבוה ביותר. ויש תוצאות לא רעות.

האם זה מצביע על כך שהמודלים יודעים לעשות ריזונינג - לא יודע, מבטיח לחשוב על זה לעומק....

https://arxiv.org/pdf/2402.04494v2

⚡🚀המאמר היומי של מייק -26.10.24: ⚡🚀
Efficient Vision-Language Pre-training by Cluster Masking

היום סוקרים מאמר נחמד בנושא של למידה ניגודית (contrastive learning) אבל הפעם עבור מקרה מולטימודלי. כלומר הפעם מאמנים מודל בסגנון של CLIP הידוע המיועד לבניית ייצוג דאטה ויזואלי (תמונות) והשפה. הרעיון העיקרי בלמידה הניגודית הוא לאמן מודל הממפה קרוב (במרחב האמבדינג) פיסות דאטה דומות ורחוק (באותו המרחב) פיסות דאטה לא דומות.

אבל הפעם מדובר בדאטה מולטימודלי. ב-CLIP המקורי אימנו את המודל לקרב ייצוג של אוגמנטציות שונות של תמונה עם הכותרת שלה ולהרחיק אותן (האמבדינגס של האוגמנטציות השונות של התמונה) מהייצוגים של כותרות שנבחרות באקראי. דבר דומה נעשה דומה לייצוג כותרת של תמונה: מקרבים לאמבדינג של אותה התמונה (עם אוגמנטציות) ולהרחיקו מהייצוגים של השאר.

נציין ש-CLIP המקורי אימן שני מודלי ייצוג שונים(למיטב זכרוני) לתמונות ולשפה אבל יצאו גם שדרוגים שאימנו שני מודלי ייצוג עם הרבה משקלים משותפים (אותה הארכיטקטורה).

הכל טוב ויפה אבל נשאלת השאלה האם ניתן לשפר כאן משהו? מתברר שכן ומהמאמר מציע שכלול קליל ל CLIP.ל-CLIP. כתבתי שאחד הדברים החשוב ב-CL הינה בחירה של הזוגות של פיסות דאטה לא דומות(זוגות שליליות). ככל שיהיה יותר מגוון בזוגות השליליות הייצוג שייבנה יהיה חזק יותר (כי ראה יותר דברים לא דומים ואז יבין יותר טוב איך "צריך להיראות ייצוג טוב".

אז המחברים מציעים לקלסטר פאצ'ים בתמונה לקלסטרים וכל פעם לא לבחור את הזוגות החיוביים בצורה אקראית אלא לאפשר בחירה של פאץ' אחד מתוך כל קלסטר. כלומר, לכל באץ' בוחרים רק פאץ' אחד מהקלסטר. כלומר פאצ'ים דומים מדי לא נכנסים לזוגות השליליים ב-CL. הקליסטור יכול להתבצע על הערכים של הפיקסלים בשילוב עם מודל אמבדינג כלשוה.

מאמר פשוט - לקח לי איזה דקה להבין ו 10 דקות לכתוב סקירה. אוהב כאלו...

https://arxiv.org/pdf/2405.08815

⚡🚀המאמר היומי של מייק -28.10.24: ⚡🚀
HEAVY-TAILED DIFFUSION MODELS

המאמר עם השם הקצר הזה משך את עיניי כי יש לי חיבה גם למודלי דיפוזיה גנרטיביים וגם להתפלגויות בעלות תכונות מעניינות למשל זנבות כבדים. בגדול התפלגות נקראת בעלת זנב כבד או ארוך כאשר התפלגות לזנב שלה (כלומר המסה ההסתברותית מימין לנקודה) מקשל (הסתברות) הינה גבוה יותר מאשר להתפלגות מעריכית. נשמע קצת מסובך אבל במילים פשוטות ניתן להגיד כי להתפלגויות בעלות זנב כבד(HT) יש יותר מסה בקצוות.

למשל התפלגות נורמלית אינה בעלת זנבות כבדים והתפלגות סטודנט t וגם התפלגות קושי הן כן. אוקיי, למה אני בכלל מדבר על זה? הסיבה היא די פשוטה - ההנחה שנוכל להניח התפלגות גאוסית על כל סוג של דאטה אינה

נכונה. יש סוגי דאטה שלא ניתן לאפיין אותם בצורה טוב עם התפלגות בעלות זנבות קלים. עקב גם אנו נתקשה לגנרט דאטה מהתפלגויות אלו אם נמדל אותו (הדאטה) עם מודלי הבנויים על הנחות גאוסיות גם אם המודלים האלו הם בעלי expressiveness גבוהה כמו מודלי הדיפוזיה. עדיין יהיה מאוד בעייתי ליצור באמצעותם דאטה בעלת התפלגות HT במיוחד בקצוות ההתפלגות.

אז המאמר, שהוא אחד הכבדים ביותר מתמטית מאלו שראיתי לאחרונה, מציע להחליף את התפלגויות גאוסיות שיש לנו במודלי דיפוזיה בהתפלגות סטודנט שהיא התפלגות HT. כלומר כל מה שהיה בעלת התפלגות גאוסית במודל דיפוזיה מקורי יהיה מהתפלגות t. דרך אגב אחד הפרמטרים של התפלגות t (שהיא כמובן וקטורית עבור מודלים אלו כי אנו רוצים לגנרט דאטה בעלת מימדים רבים) שהוא שולט ב"כבדות הזנב" שלה וכאשר היא שואפת לאינסוף אנו מקבלים את ההתפלגות הגאוסית האהובה עלינו. כלומר המודלים המוצעים במאמר הם הכללה של מודלי דיפוזיה גאוסיים שאנו מכירים ואוהבים.

כמובן שלא מספיק סתם להחליף התפלגות גאוסית במודל דיפוזיה בהתפלגות t - זה דורש להגדיר לא מעט התפלגויות מותנות הנדרשות לנו להגדרת הלמידה של תהליך denoising. זה די לא טריוויאלי אבל העקרון נשאר דומה -מאמנים את המודל להסיר רעש (שהוא מפולג עם t) באופן הדרגתי. במקום KL divergence המוכר לנו ממודלי דיפוזיה המחברים משתמשים ב-γ-Power divergence כדי למדוד מרחק בין ההתפלגות הדאטה אחרי הסרת רעש לזה של הדאטה האמיתי (לכל איטרציה).

גם תהליך הגנרוט מוגדר דומה עקרונות למודלי דיפוזיה גאוסיים אבל כמובן כל ה-hyperparameters מותאמים להתפלגות t. יש גם רפרמטריזציות שאנו כה אוהבים במודלי דיפוזיה, ייצוג באמצעות משוואות דיפרנציאליות חלקיות, גם באמצעות טכניקה חדשה הנקראת flow matching (הבונה מסלול מיטבי בין ההתפלגות ההתחלתית והתפלגות הדאטה). כאמור מאמר די כבד מתמטית ומקווה שהצלחתי להסביר לכם את העקרונות לפחות.
https://arxiv.org/pdf/2410.14171

⚡🚀המאמר היומי של מייק -29.10.24: 🚀⚡

Global Lyapunov functions: a long-standing open problem in mathematics, with symbolic transformers

אתם אולי שמתם לב שיש לי נטייה לא להתלהב יותר מדי מיכולות של מודלי שפה בטח בתחומים של ריזונינג ופתרון בעיות מתמטיות קשות. אז היום אני מודה שאני קצת (ממש טיפה) מתלהב מהמאמר שאני הולך לסקור. המחברים אימנו מודל המסוגל למצוא פתרונות של בעיה מתמטית קשה שאין דרך כללית למציאת פתרונה.

מדובר בבעיית חיפוש של פונקצית ליאפונוב למערכת דינמית. מערכת דינמית היא מתוארת על ידי מערכת משוואות דיפרנציאליות במישור במישור הזמן. ידוע שאם קיימת פונקציית ליאפונוב למערכת דינמית אז ניתן להגיד שהיא (המערכת) יציבה. המערכת יציבה אם הפתרון שלה לא מתבדר בזמן כלומר נמצא בתחום מסוים סביב 0 עבור כל זמן t (או לפעמים שואף ל 0).

לפונקציית ליאפונוב V(x) תכונות מסוימות (כי כמובן תלויה בפתרון x(t) של מערכת הדינמית (למשל V(0)= 0 והיא שואפת לאינסוף כאשר x שואף לאינסוף. למיטב זכרוני V(x) קשורה לאנטורפיה של המערכת (תקנו אותי אם אני מתבלבל כאן).

כאמור אין דרך כללית למצוא V(x) עבור כל מערכת דינמית אבל למערכות דינמיות מצורה מסוימת (פולינומיאלית) ניתן למצוא אותה. המחברים למעשה אימנו טרנספורמר שבהינתן מערכת דינמית יודע למצוא את V(x) עבורו. הם

בנו דאטהסט של מערכות משוואות דיפרנציאלית עבור מערכות דינמיות ו$V(x)$ עבורן ואימנו טרנספורמר לחזות את פונקציית ליאפונוב שלהם וזה גם עבד במקרים שלא ניתן לעשות זאת בדרך מתמטית ריגורוזית.

הדאטה מועבר לטרנספורמר בצורה סימבולית כלומר כל נוסחה מתוארת על ידי שכל קודקוד בו הוא או פונקצייה מתמטית או משתנה ואילו הקשתות מקודדת פעולות מתמטיות שונות. עץ זה מוזן לטרנספורמר בסדר מסיום (קבוע לכולם).

חייב להגיד שזה די מרשים אך מסייג את זה בהבנתי הרדודה בנושא המערכות הדינמיות.
https://arxiv.org/abs/2410.08304

⚡🚀המאמר היומי של מייק -30.10.24: 🚀⚡
Beyond Preferences in AI Alignment

היום סקירה של מאמר ללא נוסחאות אבל קשה לי לקרוא לה קלילה. יש בה דיונים פילוסופיים לא פשוטים וזה מה שהבנתי מהם (תקנו אותי אם אני טועה).

המאמר מציג ביקורת מקיפה על הגישה המבוססת-העדפות(preference based) ליישור(alignment) של AI. המחברים טוענים שהגישה הנוכחית, המתמקדת בהעדפות אנושיות כיחידה הבסיסית של ערכים אנושיים, היא בעייתית ומוגבלת.

הם מציעים מסגרת חלופית המכירה בכך שהעדפות אנושיות הן מורכבות, משתנות לאורך זמן, ותלויות בהקשר חברתי. המאמר מציע גישה חדשה המבוססת על קריטריונים נורמטיביים ספציפיים לתפקיד (של המודל), במקום על העדפות גולמיות.

המאמר גם דן בצורך במערכות AI שמסוגלות להבין ולכבד את המורכבות של ערכים אנושיים, במקום לנסות לפשט אותם למודל של העדפות פשוטות. הם מציעים גישה חוזית (contractualist) ליישור AI, המבוססת על הסכמה הדדית בין בעלי עניין שונים. יש שם (במאמר) ביקורת על התיאוריה הקיימת של בחירה רציונלית (שהיא סוג של preference-based שיש לנו כרגע) ומציע חלופות המתחשבות במגבלות הקוגניטיביות האנושיות.

הכותבים מתייחסים לשאלה כיצד לטפל במצבים בהם העדפות שונות מתנגשות זו בזו. הם מציעים מודל חדש הנקרא Evaluate, Commensurate, Decide המתאר כיצד ערכים אנושיים משפיעים על העדפות. המאמר מציע כמה דרכים ליישום גישות אלו לאימון מודלי AI (בצורה די כללית אני חייב להגיד). המאמר מציע מסגרת (תיאורטית) לפיתוח מערכות המסוגלות להתמודד עם שינויים בהעדפות אנושיות לאורך זמן.

המאמר מדגיש החשיבות של פיתוח מערכות AI שיכולות לתפקד כ"כלים"(מתוחכם אבל מתמחה אך עם "מרחב פעולות צר ומוגדר") ולא כסוכנים אוטונומיים.

ניתן למצוא במאמר גם(איך לא) דיונים בחשיבות של שמירה על פלורליזם בפיתוח AI, כך שמערכות, משלבות AI, יוכלו לשרת מטרות שונות תוך כיבוד נורמות מוסכמות המשתנות לקבוצות שונות ולפעמים תלויות גם בנסיבות.

יאללה, עכשיו תגידו האם הבנתי נכון....

https://arxiv.org/abs/2408.16984

⚡🚀המאמר היומי של מייק -31.10.24: 🚀⚡
Understanding Transformers via N-gram Statistics

מאמר די נחמד ולא רגיל מבית גוגל. המאמר מחזיר אותנו לתקופה שלא מידלנו את השפה הטבעית באמצעות מודלים סטטיסטיים עם עשרות ומאות מיליארדי פרמטרים. פעם ניסינו להשתמש ב- n-grams כדי לשערך את ההתפלגות של המילים בטקסט. כמובן גישות כאלו לא יכולות לעבוד עבור דאטהסטים בעל עשרות טריליוני טוקנים כמו שיש לנו היום אבל אולי אפשר לקחת LLMs גדולים ולבדוק האם ניתן לקרב את חיזויהם באמצעות סטטיסטיקות על n-grams. כדי לא לסבך המאמר לא בודק את זה על למידת in-context.

וזה בדיוק מה שהמאמר הזה (שיש לו רק מחבר אחד שזה די נדיר בימינו) עושה. הוא בודק האם ניתן לחזות את הטוקן הבא שמודל שפה מאומן חוזר באמצעות סטטיסטיקה של n-grams שבאים לפניו בטקסט. במקרה הזה n-grams בנויים לא ממילים אלא מטוקנים. דרך אגב הסטטיסטיקה של n-grams אינה חייבית לכלול את כל n הטוקנים הבאים לפני הטוקן הנחזה אלא עשויה "להכיל חורים"(כלומר יכולה לקחת טוקן i i-1, i-2 ו- i-4 עבור gram-3 - נצטרך למצע מעל טוקן i-3 בשביל כך).

המחבר מצא כמה דברים מעניינים. ניתן לשערך את החיזוי של מודל שפה עם gram-7 (עבור דאטהסטים שהם בחרו) בלא מעט מקרים. בנסוף נמצא כי לטוקנים בעל שונות נמוכה (של ההתפלגות שלהם) n-grams מצליחים יותר מאשר לטוקנים בעל שונות חיזוי גבוהה. מעניין שככל שמאמנים מודל שפה יותר יותר קשה לקרב אותה עם n-grams (צריך להגדיל את n או לא משנה מה ה-n דיוק הקירוב יורד).

אהבתי...

https://www.arxiv.org/abs/2407.12034

⚡🚀המאמר היומי של מייק -01.11.24: ⚡🚀

LLMs Are In-Context Reinforcement Learners

אני אוהב מאמרים שמשלבים כמה שיטות של ML. אסקור היום אחד כזה המציע לשדך למידת in-context עם למידה באמצעות חיזוקים או בקצרה RL. למידת in-context היא יכולת של מודל שפה ללמוד משהו חדש מכמה דוגמאות בפרופמט ללא צורך בפיין טיון. יש לא מעט הסברים ליכולת די מפתיע זו ולפעמים יכולת זו נקראה emergent capabilities.

עכשיו נשאלת השאלה: איך נוכל לבחור דוגמאת להדגמה שאנו מראים למודל שפה בפרומפט למקסום ביצועיי המודל? השאלה הזו לא מאוד טריוויאלית ואין עליה כרגע תשובה חד משמעית. המחברים מציעים לגשת לבעיה זו דרך למידה עם חיזוקים (סוג של). השיטה הנאיבית היא פשוט לצבור דוגמאות עד שנגמר לנו את אורך חלון ההקשר של המודל. לכל דוגמא בהדגמה אנו שומרים בבאפר את השלישיה המכילה את הדוגמא (שאלה עצמה), תשובת המודל ומשערך של איכות התשובה (או פשוט האם התשובה נכונה או לא). ואז באינפרנס פשוט לוקחים את הדוגמאות האלו בתור פרומפט.

לטענת המחברים הגישה הנאיבית הזו לא עובדת משתי סיבות עיקריות. קודם כל שילוב מתמשך של אותם הפרומפטים לדוגמאות שונות מוביל לשונות גדולה בפלט של LLM (לפי המחקרים הקודמים עלולה להוביל לביצועים ירודים). הסיבה השניה טמונה בכך ששלישיות (שאלה, תשובה, לא נכון) מסבכות את המודל ולא מספקות לו מספיק מידע על איך היה צריך לענות נכון (ד"א בלמידה ניגודית יש בעיה דומה המצריכה כמות מאוד גדולה של דוגמאות שליליות בכל באץ' - כתבתי על זה לא מעט בסקירותיי).

עקב כך המחברים הציעו להכניס קצת "אקראיות" לבניית הפרומפטים (המחברים קוראים לזה אפיזודה בהתאם לטרמינולוגיה של RL - כל אפיזודה מורכבת מכמה שלישיות של שאלה, תשובה, נכונות התשובה) וגם להשתמש באפיזודות שקיבלו ציון "נכון". לכל דוגמא הם הציע קודם לדגום באקראי מהבאפר של אפיזודות בצורה אקראית

ולהשתמש לכל דוגמא במדגם שונה של אפיזודות. כאמור שומרים רק את האפיזודות שבהם המודל צדק. כך פרומפט לכל שאילתה הופך להיות לא קבוע ומכיל רק דוגמאות עם תשובות נכונות. זה נקרא Explorative ICRL במאמר.

כמובן ש Explorative ICRL לא יעיל חישובית כי כל פעם צריך לחשב את הפרומפט מחדש (מה שלא צריך לעשות בגישה הנאיבית אך לא עובדת). המחברים שכללו את זה עם מנגנון קאשינג המאפשר לשמור מספר קבוע של פרומפרטים (מערך של אפיזודות) ולכל אפיזודה נתונה להחליט לאלו מהם להוסיף אותה. זה מקל על העלות החישובית.

מאמר חמוד למרות שמשום מה לקח לי קצת זמן להבין אותו...

https://arxiv.org/pdf/2410.05362

🚀⚡ :המאמר היומי של מייק -02.11.24🚀⚡

Learning to Compress: Local Rank and Information Compression in Deep Neural Networks

היום סוקרים מאמר כחול לבן למחצה (אחד המחברים משניים הוא ישראלי רביד שוורץ זיו) והם חוקרים נושא שמעניין אותי מאוד באופן אישי. הנושא הוא דחיסה של דאטה באמצעות רשתות נוירונים והוא גם מאוד קשור לעבודות של נפתלי תשבי האגדי בנושא צוואר בקבוק מידעי (information bottleneck או IB) וגם השערת יריעה (manifold hypothesis או MH) בנוגע לרשתות נוירונים עמוקות.

MH טוענת שדאטה מהעולם האמיתי (כגון תמונות או טקסט) אינם מפוזרים באופן אחיד במרחב בעל מימד גבוה, אלא שוכנים על יריעה בעל מימד נמוך יותר. רשתות נוירונים עמוקות מצליחות היטב עם הדאטה הז כי הן לומדות לזהות ולנצל את המבנה של אותה יריעה, מה שמאפשר להן לבצע הכללה טובה למרות המורכבות העצומה של המרחב המקורי.

כמובן שזה קשור לדחיסה כי ניתן לראות במיפוי ממרחב בעל מימד גבוה למרחב בעל מימד נמוך שהרשתות עושות בהתאם ל MH סוג של דחיסה. ניתן לראות "שהמימד האמיתי" של מרחב הפיצ'רים של שכבה ברשת נוירונים קשורה לראנק(=דרגה) של היעקוביאן שלהם (הפיצ'רים) ביחס לקלט. למה זה קורה בעצם? הרי מרחב האפס של היעקוביאן מייצג כיוונים שבהם האקטיבציות של השכבה לא משתנות (כפונקציה של הקלט). ככל שמימד של מרחב האפס גדול יותר הדרגה של יריעת הפיצ'רים בשכבה נמוכה יותר. משמעות הדבר היא שהתרחשה יותר "דחיסה" או הפחתת מימדי הקלט.

נציין שמטריצות עם דרגה לא מלאה מהוות מרחב בעל מידה אפס במרחב של כל המטריצות (כמו הסתברות של כל מספר עם דוגמים יוניפורמית בין 0 ל 1). עקב כך המאמר מגדיר robust local rank או RLR שזה מספר ערכים סינגולריים (הכללה של הערכים העצמיים) של היעקוביאן שהם גדולים ממספר קטן אפסילון אך חיובי (נזכור עבור דרגה אמיתית צריך להחליף אפסילון ב 0).

אוקיי, מקווה ששרדתם את זה אז עכשיו מגיעים שני המשפטים העיקריים של המאמר. הם טוענים שברשתות עמוקות (מספר שכבות גבוה) בבעיות סיווג תמיד יהיה שכבה l שה-RLR יהיה נמוך מ-(פרופורציונלי לאפסילון בחזקה מינוס 2 ובנורמת אופרטור של מטריצת השכבה l (נורמת אופרטור זה הערך הסינגולרי הגבוה ביותר). הכוונה כאן לרשת שעושה התאמה מושלמת לדאטה האימון (עם מרג'ין 1 כלומר מצליחה להפריד בין הקטגוריות השונות בבטחה). משמעות המשפט היא שהרשת המאומנת דוחסת את הדאטה בשכבה l באופן אפקטיבי.

המחברים מוכיחים משפט דומה בנוגע לבעיות רגרסיה.

תמיד כיף לצלול למתמטיקה חמודה עם המאמרים של רביד 🙂

https://arxiv.org/abs/2410.07687

⚡🚀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-PAtenttion 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

⚡🚀המאמר היומי של מייק -04.11.24: ⚡🚀

Refusal in Language Models Is Mediated by a Single Direction

מאמר מעניין החוקר איך ניתן לגרום למודל שפה לתת תשובות רצויות יותר ורצויות פחות. מתברר שאפשר לגרום למודל להסביר לנו איך מכינים הרואין או שודדים בנק ונמלטים מהעונש עם אם מזיזים פלט של שכבה אחת

במודל שפה. וגם ניתן למנוע ממודל "לא מרוסן" לתת תשובות לא פוגעניות ולפעמים להימנע מלענות על שאלות מסוכנות אם מזיזים את הפלטים של כל השכבות של מודל, כל אחת עם וקטור r_l כאשר l זו מספר השכבה.

איך בעצם מוצאים את הוקטורים האלה? עבור דאטהסט המכיל שאלות ותשובות רצויות מחשבים את ההפרש r הממוצע (על כל התשובות) בין האקטיבציות של כל שכבות המודל ועבור כל הטוקנים של חלון ההקשר. כלומר יש לנו מטריצה LxI של וקטורי ההפרש כאשר L זה מספר השכבות ו I זה מספר הטוקנים בחלון ההקשר.

כדי לגרום למודל להיות "פחות מרוסן" אנו בוחרים שכבה שהוספתן של מורידה ממנו את בלמים בצורה המשמעותית ביותר (יש מדדים לא רעים לכך). כלומר משאירים I וקטורי הפרשים שחישבנו. כדי לגרום למודל להיות יותר מנומס צריך להחסיר את "כיוון הגסות" מכל השכבות של המודל בצורה שתעביר אותם ממרחב אורתוגונלי ל r (כל שכבה ולכל טוקן בחלון ההקשר). בפרט מכל אקטיבציה x בכל שכבה ובכל טוקן : x* r^T * r קל לראות שהווקטור המתקבל כתוצאה מכך יהיה אורתוגונלי ל r.

עושים זאת לווקטור האקטיבציה לפני residual connection בכל בלוק של טרנספורמר. כמובן (מכיוון שיש הרבה מכפלות של מטריצות)ניתן להזיז גם את המשקלים שלהם כדי לקבל את אותם האפקטים. מאמר די מגניב וקל להבנה.

https://arxiv.org/abs/2406.11717

⚡🚀המאמר היומי של מייק -05.11.24: ⚡🚀

RETHINKING SOFTMAX: SELF-ATTENTION WITH POLYNOMIAL ACTIVATIONS

מאמר די לא רגיל והוא מדבר על חלופה פוטנציאלית של מנגנון ה-attention שאנו כה אוהבים בטרנספורמים. אתם בטח זוכרים שמשקלי attention בשנאים מחושבים עם softmax שהוא מנרמל וקטורי משקלים לנורמה 1 ובנוסף כל רכיביו הינם בין 0 ל- 1 כלומר הוא מהווה התפלגות הסתברותית. המחברים טוענים שתכונות אלו של המשקלים לא קריטיות לפונקציונאליות של השנאים ומציעים להחליף אותם בקרנל אחר שהוא פולינומיאלי כפי שאתם בטח ניחשתם מהשם של המאמר.

אבל למה זה עובד בכלל? המחברים טוענים (באופן די מפתיע, אני חייב להגיד) שהביצועים הנפלאים של הטרנספורמרים נובעים בחלקם מיכולתה של פונקציית סופטמקס לכפות רגולריזציה מסוימת על נורמת פרובניוס של מטריצה המשקלים וגם של היעקוביאן שלה (ביחס לקלט של הסופטמקס) במהלך האימון הוא מסדר sqrt(n) כאשר n הינו מימד לקלט.

נורמת פרובניוס או NF מוגדרת בתור שורש של סכום הריבועים של כל הערכים במטריצה והיא גם שווה לשורש של סכום הריבועים הערכים הסינגולריים (הכללה של ערכים עצמיים למטריצות לא ריבועיות). ד"א סופטמקס מחושב במנגנון ה-attention של מערך של וקטורים אז היעקוביאן תיאורטית הוא טנזור תלת מימדי (המאמר מפרט איך מחשבים את NF במקרה הזה).

אז בגדול המאמר מוכיח שני משפטים. בראשון מהם טוענים ש NF של מנגנון attention פולינומיאלי (כולל הלינארי) מתנהג לפי O(n) אם המטריצות שם, K ו-Q וגם ייצוגי הטוקנים מפולגים גאוסית כמובן). אז אם מנרמלים את ה-attention הפולינומיאלי עם n^(-0.5) מקבלים את sqrt(n) שהיה לנו עבור מנגנון ה-attention הרגיל. בנוסף NF של היעקוביאן לפי Q, המנורמל לפי n^(-0.5) (לא זה שמתנהג לפי sqrt(n) ב-attention הרגיל) גם מתנהג לפני sqrt(n).

המחברים טוענים שזה מספיק כדי לטעון שניתן להחליף סופטמקס בפולינומים שיותר קלים מבחינה חישובית, מקבלים תוצאות מעודדות אבל אני עדיין לא השתכנעתי...

https://arxiv.org/abs/2410.18613

⚡🚀המאמר היומי של מייק -07.11.24: 🚀⚡
Cross-layer Attention Sharing for Large Language Models

אתם בטח יודעים הרצה של מודלי שפה עלול להיות דבר די יקר מבחינת משאבי חישוב וגם הזכרון. בטח כאשר יש לכם מודלים עם עשרות מיליארדי פרמטרים על עשרות רבות של שכבות של טרנספורמרים. אחד הדברים הכבדים שמצריכים לא מעט זיכרון הוא KV-Cache, שבו נשמרים המכפלות של ייצוגי (אמבדינגס) של הטוקנים במטריצות K ו- V לכל השכבות ולכל הטוקנים שכבר גונרטו (כולל הפרומפט - מדובר במודלי הדקודרים).

כמובן שכאשר המימדים של וקטורי הייצוג והמטריצות לא קטנים וגם אורך ההקשר נמדד בעשרות ומאות אלפים KV-Cache דורש הרבה מאוד זיכרון. בעבר יצאו לא מעט מאמרים שניסו לדחוס אותו על ידי ניתוח וזיהוי יתירויות אבל זה בד״כ נעשה פר שכבה (= בלוק הטרנספורמר). המאמר המסוקר מציע להתבונן בדחיסת KV-cache מפרספקטיבה רחבה יותר ולנסות לדחוס אותו דרך ניצול התלויות של ה-KV-cache בין השכבות השונות.

המחברים חקרו דמיון בין החלקים השונים בבלוק הטרנספורמרים (מכפלות של המטריצות השונות בוקטורי ייצוג, מקדמי attention וכדומה) והגיעו למסקנה שניתן ״להסיק״ את מקדמי ה-attention של שכבה n מהדאטה של שכבה n-1 בצורה חסכונית חישובית. כלומר עם הרבה פחות משקולות מהטרנספומר הרגיל. כלומר ההצעה היא לעשות סוג של LoRa אבל למקדמי ה-attention.

בצורה קצת יותר קונקרטית המאמר החליף מטריצות W_Q ו-W_K במטריצות בעלות ראנק נמוך (מכפלה של שתי מטריצות מלבניות כאשר המימד הפנימי של המכפלה נמוך - כלומר (M x k * k x N) כאשר k קטן הרבה יותר מ- M ו- מ-M. מחשבים את הקלט לסופטמקס עם המטריצות האלו. לאחר מכן משרשרים אותם עם הקלט לסופטמקס מהשכבה הקודמת, מפעילים FFN והנה יש לנו קלט לסופטמקס בשכבה n. ושימו לב שאנו צריכים לשמור הרבה פחות דאטה ב- KV-cache כי יש לנו מטריצות בעלות ראנק נמוך.

איך מאמנים את הסיפור הזה? משלבים את הלוס הרגיל של מודל שפה עם לוס distillation שמטרתה לקרב את מקדמי -attention המחושבים בדרך המוצעת עם אלו שמחושבים עם מודל רגיל (עם attention ו- KV Cache רגילים).

מאמר די מעניין - אבל קצת ארוך מדי לדעתי אז תמצתתי לכם אותו 🙂

https://arxiv.org/abs/2408.01890

🚀⚡ :המאמר היומי של מייק -08.11.24🚀⚡
Occam's Razor for Self Supervised Learning: What is Sufficient to Learn Good Representations?

סקירה קצרה של מאמר המציע גישה חדשה ללמידה self-supervised או SSL בקצרה. אזכיר כי שיטת SSL מניחה שיש לנו דאטה לא מתויג ומתרטנו לאמן מודל מסוגל להפיק ייצוג חזק של דאטה. מה זה ייצוג חזק של דאטה, אתם שואלים? בד״כ הכוונה לכזה שניתן למנף אותו בצורה קלה (נגיד רק עם תוספת של שכבה לינארית) לבניית מסווג בעל ביצועים טובים.

כלומר כזה שיודע להפריד בין הקטגוריות השונות של דאטה בלי לדעת אותן בצורה מפורשת (למשל אנו יכולים לאמן מודל בצורת SSL על התמונות של ImageNet בלי להשתמש בתיוגים ואז לבדוק האם המודל הצליח ללמוד להפריד בין הקטגוריות השונות).

בד״כ SSL מבוצע עם שיטות של למידה ניגודית (contrastive learning) כאשר מטרתו מאוד בגדול היא לקרב ייצוגים של פיסות דאטה דומות (חיוביות) ולהרחיק את הייצוגים של פיסות דאטה לא דומות (שליליות). לרוב זוגות חיוביים נבחרים בתור אוגמנטציות שונות של אותה הדוגמא כאשר הזוגות השליליים הן דוגמאות שנבחרות באקראי. שיטות כאלו נחלו הצלחה די גדולה אבל דרשו דאטהסטים מאוד גדולים וגם משאבי אימון די משמעותיים (כי נדרש שם גודל באץ' די גדול כדי שהשיטה תעבוד טוב).

המאמר המסוקר מציע שיטה מאוד פשוטה ואינטואטיבית ל-SSL(תער אוקם). במקום לעבוד עם הייצוגים המאמר מאמן מודל לחזות את המספר של הדוגמא בדאטהסט. כלומר אם יש לנו 1000 דוגמאות מהדאטהסט יש לנו 1000 קטגוריות ומטרתנו לחזות קטגוריה של דוגמא מהייצוג הלטנטי שלה (המופק על ידי המודל המאומן). כלומר אחרי השכבה האחרונה של המודל מוסיפים שכבה עם מטריצה הממפה את הייצוג לקטגוריות (כלומר המספרים הסידוריים של הדוגמאות). ובסוף של לוס cross-entropy הסטנדרטי.

אז המאמר מוכיח שהשיטה עובדת לא רע לדאטהסטים יחסית לא גדולים (מעניין איך זה יעבוד לדאטהסט בגודל 10 מיליון). כמובן יש כמה טריקים באימון כמו soft labels אבל בגדול הרעיון די נחמד.
https://arxiv.org/pdf/2406.10743

⚡🚀 :המאמר היומי של מייק -09.11.24🚀⚡
CROSS-ENTROPY IS ALL YOU NEED TO INVERT THE DATA GENERATING PROCESS

מאמר המשך של המאמר שסקרתי אתמול שהציע שיטה חדשה ל-SSL או Self-Supervised Learning באריכות. מטרת SSL היא לבנות מודל המפיק ייצוג דאטה עוצמתי שיהיה קל לבנות ממנו מודלים downstream לביצוע משימות שונות על הדאטה הזה בתור backbone (למשל על ידי הוספת שכבות, LoRA, אדפטרים או שיטות פיין טיון אחרות הבנויות על ה-backbone הזה). כלומר הייצוג הזה צריך להיות מסוגל לזקק את כל התכונות המהותיות של הדאטה הזה כלומר לדחוסו בצורה יעילה.

משימת downstream הפשוטה ביותר היא משימת סיווג ובמקרה הזה מודל ייצוג טוב צריך להיות מסוגל להבדיל בין דאטה שייך לקטגוריות שונות (למרות שהמודל עצמו מאומן על דאטה לא מתויג). המאמר של אתמול הציע לאמן מודל שיודע לזהות פיסת דאטה מהייצוג שלה. כלומר כל פיסת דאטה מקבלת קטגוריה משלה (כלומר אם יש לנו דאטהסט עם 10L דוגמאות אז יש לנו 10K קטגוריות). בגדול מאמנים שכבה לינארית בנוסף לאנקודר (מודל הייצוג) שממפה (השכבה הלינארית) את וקטור הייצוג לקטגוריות עם לוס cross-entropy.

אז המאמר של אתמול טען שניתן להגיע לייצוגים חזקים עם השיטה הזו (למשימות downstream מסוג סיווג) והמאמר המסוקר הוכיח כמה טענות לגבי הרעיון שנדון במאמר (טוב זה לא בדיוק אבל קרוב) שסקרנו אתמול תחת הנחות די הגיוניות. המאמר די מתמטי ואנסה להסביר את הרעיון העיקרי בלי לצלול לנוסחאות וללא התעמקויות יתר לפרטים מתמטיים לא מהותיים.

המחברים מניחים כמה הנחות שעוזרות להם לחקור את הגישה הזו. ההנחה הראשונה מניחה שיש תהליך גנרטיבי המגנרט פיסות דאטה השייכים לכמה קטגוריות (מספרם ידוע). בפרט היא מדברת על כך שקיים מודל גנרטיבי g המגנרט דאטה מייצוגו הלטנטי z. המשתנה הלטנטי z בהינתן קטגוריה C מוגרל מהתפלגות von

Neumann-Fisher או vMF בקצרה. vMF היא התפלגות רב מימדית על ספירה בעלת רדיוס אחת המוגדרת על ידי וקטור z_c תוחלת ופרמטר ריכוז (סקלר המגדיר את מידת המריחות של ההתפלגות).

עכשיו המשפט הראשון במאמר טוען אם מאמנים ייצוג f (האנקודר) עלי ידי מקסום פונקציה שדומה לזאת מהמאמר הקודם רק שהקטגוריות יהיה קטגוריות של הדאטה(המיוצגות במרחב הלטנטי) ולא כל פיסת דאטה שייכת לקטגוריה משלה(נכון זה לא אותו הדבר אבל עדיין), יש פירוש די יפה לוקטורים w המרכיבים מטריצת W שהיא המיפוי הלינארי שאנו לומדים מהמרחב הלטנטי למרחב הדאטה.

במקרה הפשוט - משפט אחד מגדיר 4 מקרים, התלויים האם וקטורים w (המרכיבים את W) ווקטורי ייצוג אחרי f(x), וקטורי w מהווים טרנספורמציה אורתוגונלית של מרכזי הקטגוריות z_c שממנו הוקטורים הלטנטיים מוגרלים (כלומר זה אותם הווקטורים תחת סיבוב רב מימדי כלשהו). כלומר קיבלנו w_i עם מאוד קשורים למבנה של הדאטה. בנוסף במקרה הזה ההרכבה של האנקודר f (מה שאנו מאמנים) והדקודר g הינה לינארית כלומר הצלחנו למצוא את ההופכית של הגנרטור g - וזה תוצאה די חזקה (משפט 2 מנסח את זה בצורה די טובה).

ההוכחות לא פשוטות בכלל ועם זאת המאמר הזה מאוד חשוב ואני מקווה שהצלחתי לפחות להסביר לכם את מהותו.
https://arxiv.org/abs/2410.21869

⚡🚀המאמר היומי של מייק -10.11.24: ⚡🚀
WHAT MATTERS IN TRANSFORMERS? NOT ALL ATTENTION IS NEEDED

סקירה קצרה של מאמר די נחמד החוקר איזה חלקים במודלי טרנספורמרים (או שנאים) שלנו פחות נחוצים מהחלקים האחרים (או בכלל מיותרים). כמו שאתם זוכרים בכל בלוק של שנאי יש לנו מנגנון ה-attention, כמה שכבות MLP (שזה שכבה וחצי של fully-connected) וכמה שכבות נרמול (אותם לא בודקים).
המחקרים הקודם שחקרו את הנושא הזה התמקדו בזיהוי בלוקים שלמים של שנאים העשויים להיות לא נחוצים אך המחקר הזה החליטו לרדת לרזולוציה של אבן הבניין של השנאי עצמו (כלומר attention ו-MLP).

איך בודקים האם תת-בלוק לא נחוץ? בודקים את הקלט את הפלט של תת הבלוק הזה ואם אין כמעט הבדל בינם כנראה שלא צריך אותו. כדי לבדוק את הדמיון משתמשים כמובן בדמיון קוסיין (cosine similarity). בודקים את זה על כמויות גדולות של דאטה ומתחילים להוריד שכבות ולבדוק ביצועים.

מה התברר? באופן קצת מפתיע לרוב מנגנוני ה-attention הרבה פחות נחוצים מה-MLP וניתן לוותר עליהם בלי פגיעה רצינית בביצועים במיוחד במודלים הגדולים. אז אולי זו הדרך להקטין את העומס החישובי ששימוש המודלים האלו גורם? בואו נחכה ונראה....

https://arxiv.org/abs/2406.15786

⚡🚀המאמר היומי של מייק -11.11.24: ⚡🚀
Stealing Part of a Production Language Model

מזמן לא סקרתי מאמר על איזה ניתן לפרוץ למודלים עמוקים. יש תחום שלם שנקרא adversarial learning שבו חוקרים מפתחים מנגנוני הגנה נגד התקפות שמנסות לגנוב משהו מהמודל או דרך המודל (למשל דאטה שהוא אומן עליו). המאמר שנסקור היום מציע שיטה שבאמצעותה ניתן לזהות המימד הפנימי (החבוי) של המודל (מימד

ייצוגי הטוקנים) וגם את המטריצה בשכבה האחרונה של המודל. שכבה זו הממפה את האמבדינגס של כל הטוקנים ללוגיטים שלאחר מכן מוזנים לסופטמקס שממנו יוצרים "ההסתברויות של הטוקנים.

נתחיל מכך שמימד המטריצה W בשכבה האחרונה הוא N_voc x N_emb, כאשר N_emb זה המימד הפנימי של המודל (אלפים בודדים) ו- N_voc הוא מספר הטוקנים במילון (בד"כ כמה עשרות אלפים ולפעמים מגיע מעל 100K). כלומר N_voc > N_emb וזה בדיוק מה שמחברי המאמר מנצלים. מכיוון שהראנק של מטריצה W הוא N_emb כל המכפלות בה ממפות את הוקטור לתת מרחב במימד N_emb של מחרב הלוגיטים שהוא בעל מימד N_voc. כלומר אם ניקח מספר וקטורי לוגיטים ונשים אותם לעמודות של המטריצה (נקרא לה V) המספר המקסימלי לי וקטורים בלתי תלוים שיהיה לנו יהיה בדיוק N_emb.

זה בדיוק מה שמחברי המאמר עשו. אולם מכיוון שהחישובים בטרנספורמרים הם לא בדיוק המלאה (FP16 גג) אז קשה לתפוס מתי העמודות הופכות להיות בלתי תלויות. במקום זה הם חישבו את הערכים הסינגולריים(ע"ס) של V (דרך מה שנקרא SVD - מי שלא מכיר ממליץ לקרוא על זה) ומסתכלים מתי היחס של ע"ס העוקבים (הם ממוינים) צונח משמעתית.

למה זה חשוב? כי במקרה האידאלי ע"ס של V צריכים להתאפס אחר שעברנו את הראנק של או N_emb. אז בגלל אי דיוקים נומריים במודל כמובן שלא נראה ממש אפסים שם אלא ערכים מאוד נמוכים ואיפה שזה מתחיל לקרות זה בדיוק במימד N_emb + 1. אז עושים את הטריק הזה על הרבה מאוד דאטה ומגלים את המימד החבוי של המודל שלכם.

כמובן שבעולם האמיתי אין לכם גישה לכל הלוגיטים אלא רק ל-topK ואז המאמר מנצל את העובדה שניתן לקנפג חלק מהמודל להוסיף מרג'ין לטוקן נתון במילון. ואחרי מספיק משחקים מקבלים את כל הלוגיטים (זה די יקר חישובית).

מימד של W זה נחמד אבל מה עם מטריצה W עצמה. המאמר מציע התקפה כדי לגלות אותה (סוג של) גם. בכללי המאמר מלא ברעיונות יפים להתקפות על המודלים ומי שמתעניין מוזמן להעיף מבט.

⚡🚀המאמר היומי של מייק -12.11.24: ⚡🚀

OccamLLM: Fast and Exact Language Model Arithmetic in a Single Step

זהו מאמר שממש אהבתי, אהבתי גם את הרעיון וגם כתוב בצורה מאוד ברורה. למה כה אהבתי את הרעיון? אני כבר זמן מה טוען שבמקום להשקיע מאמצים גדולים באימון מודלי שפה לפתור בעיות מתמטיות יחסית מורכבות (שלדעתי מאוד קשה כי הם לא "בנויים" לזה באופן טבעי) כדאי להשתמש בכלים חיצוניים ייעודיים לכך (למשל כלים סימבוליים). מטרה של מודלי שפה במקרה הזה היא לזהות מתי הקלט שמוזן אליו (הפרומפט) מצריך פתרון בעיה מתמטית, "לתרגם" את הבעיה לשפה של הכלי הייעודי הזה, להעביר את הבעיה המתורגת לשפתו אליו לפתרון ולפענח את הפלט שלו.

וזה בדיוק מה שהמאמר הזה עושה. המחברים לקחו מודל שפה ופתחו מודל נפרד לפתרון בעיות מתמטיות. למעשה המודל לפתרון בעיות מתמטיות שפותח במאמר הוא גרף חישובי דינמי שכל צומת בו היא פונקציה או פעולה מתמטית (נדיג סימן + ו- *, או cos ו-exp). יש גם צמתים למשתני קלט השונים כדי שהמודל יוכל לחשב פונקציות על כמה משתנים (multivariate). למעשה גרף כזה הוא DAG או בשמו המלא Directed Acyclic Graph ומאמנים אותו לבחור את "נתיב החישוב" בו ("מסלול הצמתים") בהינתן הייצוגים (אמבדינגס של הטוקנים) המוחשבים על ידי מודל שפה (ד"א מודל שפה לא מאומן ונותר קבוע לכל אורך אימון המודל).

המחברים מאמנים שני מודלים: הראשון מזהה האם יש צורך בהפעלת המודל לחישובים מתמטיים לכל טוקן בהינתן ההקשר (כלומר כל הטוקנים לפניו). המודל השני מאומן לבנות נתיב חישובי בגרף החישובי שתיארתי בפסקה הקודמת. את שני המודלים האלו מאמנים בנפרד.

מעניין כל שכבה של רשת ה-DAG הזה מורכבת משני חלקים: בחלק בראשון יש לנו צמתי החלטה: כל צומת כזה הוא וקטור "המחבר" אותו לצמתים פונקציונליים שכל אחד מהם הוא בעצם פעולה או פונקציה מתמטית (מקבוצת פעולות ופונקציות שבחרנו). הוקטור הזה הוא למעשה סופטמקס שממנו נדגם לאיזה צומת פונקציונלי/פעולה נחבר אותו. כל צומת פונקציונלי שנבחר מחובר עם כל צמתי ההחלטה מהשכבה הבאה ואליהם מועבר הייצוג משכבת ההחלטה הקודמת יחד עם ייצוג הפעולה (כנראה האם נבחרה או לא). כך נבנה גרף חישובי מייצוגי הטוקנים המחושבים על ידי מודל שפה (הם מחוברים לשכבת ההחלטה הראשון במודל החישובי). ד"א כל פעולה וכל פונקציית בסיס בגרף משוכפלת בכמה צמתית כדי להקנות למודל יכולת לקרב פונקציות מורכבות יותר.

מכיוון שאנו דוגמים את הגרף החישובי כל פעם מחדש עבור כל פלט של מודל השפה, לא ניתן לאמן אותו בקלות על שיטות קלאסיות של למידת מכונה (supervised learning). המחברים בחרו בשיטה קלאסית מעולם למידה עם חיזוקים (RL) הנקראת reinforce כאשר פונקציית reward היא עד כמה התשובה המחושבת באמצעות הגרף החישובה קרובה לתשובה ground truth. דרך אגב ניתן לייצג רוב הפונקציות עם עם יותר מאחד נתיבי חישובי.

מאמר די נחמד אבל כתוב לא מאוד ברור (או שהיה חסר לי קצת רקע)...

https://arxiv.org/abs/2406.06576

⚡🚀המאמר היומי של מייק -16.11.24: 🚀⚡

NON-NEGATIVE CONTRASTIVE LEARNING

מאמר מעניין בנושא הלמידה הניגודית (contrastive learning) או CL בקצרה. נזכיר שמטרת CL היא לבנות ייצוג יעיל לדאטה לא מתויג שנוכל להשתמש בו לאחר מכן לאימון מודלי לשמישות downstream שונות (למשל על ידי הוספה של כמה שכבות ייעודיות למשימה למודל שבונה את הייצוג). השיטה הפופולרית ביותר ל-CL (שלה יש וריאציות ושכלולים רבים) היא InfoNCE הוצעה לראשונה במאמר של Oord et al כבר בשנת 2018 הרחוקה.

השיטה מנסה לקרב ייצוגים של דוגמאות דומות (כגון אוגמנטציה של אותה התמונה) מבחינה דמיון קוסיין (מכפלה פנימית מנורמלת) ובאותו הזמן היא מנסה להרחיק ייצוגים של דוגמאות לא דומות (הנבחרות בד"כ באקראי). זה נעשה (בגדול) עלי ידי אימון מודל שממזער את היחס בין מרחקי הקוסיין (מעלים אותו באקספוננט) של זוגות דוגמאות שליליים (כלומר לא דומים) לזה של זוגות דוגמאות חיוביים (דומים). נציין שבכל באץ לוקחים מספר גבוה של זוגות שליליים (את הסיבות הסברתי בסקירות הקודמות בנושא).

המאמר מציע שיטה המשפרת את איכות הייצוגים הנלמדים, למשל כאלו שבהם הקטגוריות השונות של דאטה (אזכיר שמדובר באימון עם דאטה לא מתויג) יהיו מרוכזות ב"חלקים מסוימים" (תת-וקטורים) של וקטורי הייצוג כאשר שאר הערכים יהיו אפסים או מאוד קרובים ל-0. וקטורים כאלו יהיו נוחים יותר משימות downstream הקשורים לסיווג דאטה. המאמר טוען ששיטת CL עם פונקציית לוס בסגנון InfoNCE לא מצליחות להפיק ייצוגים עם תכונות כאלו והסיבה העיקרית היא האינווריאנטיות שלהם לסיבוב הנובעת מהצורה של פונקציית הלוס שלהם (הסבר מפורט בפרק 2.1 במאמר).

המחברים מציעים שני חידושים עיקריים. קודם כל הם מציעים לאמן ייצוגים שהם לא שליליים (ב-InfoNCE אין שום מגבלה כזו). החידוש השני הוא פונקציית לוס שאכן מכילה מכפלות פנימיות של וקטורי ייצוג הדאטה אבל בלי

אקספוננטים ויחסים (כבר הוצע קודם אבל ללא אי שליליות). הפעם פונקצית הלוס היא הפרש בין המרחק הריבועי בין הדוגמאות השליליות לבין המרחק בין הדוגמאות החיוביות.

מהחברים מצטטים מאמר שהראה שהייצוגים המופקים על ידי המודל הממזער לוס זה ללא הגבלה של אי שליליות הינם שקולים לאלו המתקבלים מפקטוריזציה סימטרית (מייצגים מטריצה כמכפלה של מטריצה F והשחלוף שלה) של מה שנקרא מטריצת co-occurrence A. לקח לי קצת זמן להבין מה זה בדיוק אבל בגדול זה מטריצה המכילה סוג של "הסתברויות" של שתי דוגמאות יהיו חיוביות (אוגמנטציה של אותה הדוגמא).

כלומר אם יש לנו דאטהסט של 1000 דוגמאות ו-10 אוגמנטציות שונות פר דוגמא מטריצה A בגודל 10K x 10K מכילה 1/10 לזוגות חיוביים (כאשר תמונות i ו- j הן אוגמנטציות של אותה התמונה) 0 בשאר המקומות. מדובר כאן בפקטוריזציה למטריצה F שהיא low-rank כלומר אחד המימדים שלה (מימד הייצוג של דאטה) הוא הרבה יותר קטן מהמימדים של מטריצה A (שהיא עצומה לדאטהסטים בגודל רציני, מיליוני תמונות).

אז המאמר משתמש באותו הלוס אבל מחפש וקטורי ייצוג שהם אי שליליים (מפעילים עליהם פונקציות כגון ReLU, sigmoid, softplus וכדומה). בנוסף המחברים שמו לב כי בייצוגים המתקבלים יש נוירונים מתים כלומר כאלו שמאוד קרובים ל-0 עבור כל הדוגמאות). המחברים משתמשים בטריקים נחמדים כמו stop-gradient כדי להתמודד עם התופעה הזו.

בסוף מקבלים ביצועים משופרים כאשר הייצוגים המתקבלים הינם יותר disentangled ויותר קרובים לאורתוגונליות לדאטה מקטגוריות שונות.

https://arxiv.org/abs/2403.12459

⚡🚀המאמר היומי של מייק -18.11.24: 🚀⚡

Knowledge Editing in Language Models via Adapted Direct Preference Optimization

היום סוקרים מאמר כחול לבן בנושא פיינטיון(=טיוב, ככה אמרו לי) של מודלי שפה באמצעות טכניקות מבוססות על למידה עם חיזוקים או בקצרה RLHF. למיטב ידיעתי השימוש הראשון ב-RLHF היה במאמר במאמר שפיתח מודל הנקרא InstructGPT שמשמו כבר ברור כי אומן לעקוב אחרי הוראות המשתמשים. זה נעשה באמצעות טכניקת RL הנקראת Proximal Preference Optimization או PPO. מעניין ש-PPO הומצאה על ידי לא אחר אלא ג'ו שולמן שהיה תקופה די ארוכה CTO של OpenAI. בגדול מאמנים את המודל למקסם את פונקצית התגמול של תשובותיו תוך שמירתו (התפלגות הטוקנים) קרוב יחסית להתפלגות ההתחלתית (דרך KL divergence).

החיסרון העיקרי של PPO היה צורך באימון מודל תגמול (reward) שבהינתן שאלה ותשובה נותן ציון המשקף את איכות התשובה מנקודת ראיה של בני אדם (לפחות אלו שמאמנים מודלי שפה). לשמחתנו זמן קצר לאחר מכן הוצעה גישה הנקרא DPO או Direct PO שאפשרה לטייב (או ליישר כמו align) מודלי שפה ללא צורך בלהשתמש במודל תגמול בצורה מפורשת (מניחים צורה אופטימלית של התגמול נפטרים ממנו). כדי לאמן מודל שפה בשיטת DPO צריך דאטהסט המורכב מתשובות רצויות יותר ורצויות פחות ואנו מאמנים מודל.

המאמר למעשה פיתח שיטה שהתאימה את DPO לבעיה של עריכת ידע (knowledge editing) של מודל שפה. כלומר אנו רוצים שהמודל יענה אחרת על שאלות מסוימות (נגיד מתאימים אותו לדומיין מסוים). בעיה זו שקולה לבעיית יישור מודל שפה שניתן לפתור עם DPO. המחברים הציעו 3 שכלולים עיקריים ל-DPO:

- במקום סט שאלות ותשובות(חיובית ושלילית) התשובות השליליות נוצרות על ידי המודל במהלך האימון
- התשובות השליליות(המודל הנוכחי) מג'ונרטות עם מה שנקרא teacher-forcing. כלומר עד טוקן שחוזים משתמשים בטוקנים של התשובה החיובית (שאותה אנו מצפים לקבל מהמודל לאחר עריכת ידע)

- האופטימיזציה עם DPO מבוצעת עם ה-teacher forcing הזה (נשמע מאוד הגיוני עם 2)

ויש תוצאות לא רעות כמובן...

https://arxiv.org/abs/2406.09920v1

⚡🚀המאמר היומי של מייק -20.11.24: 🚀⚡

Adaptive Decoding via Latent Preference Optimization

היום סוקרים מאמר ששוב שכנע אותי שלא משנה כמה מאמרים אקרא עדיין אפספס רעיונות מעניינים גם בתחומים שאני מתמחה (סוג של) ומתעניין. כמובן מדובר בשיטות לג'נרוט דאטה ממודלי שפה? המאמר הזה מציע שיטה המתאימה את הייפר-פרמטרי הג'נרוט שלה כפונקציה של הקונטקסט. למשל המאמר שנסקור היום עוסק בהתאמה של טמפרטורת דגימה לגנרוט דאטה. אזכיר לכם שטמפרטורת הדגימה T שולטת באקראיות דגימה של טוקן הבא - ככל שהיא גדולה יותר טוקנים עם "הסתברות דגימה" (מותנית בהקשר) נמוכה יותר מקבלים יותר סיכוי להידגם.

מתברר שקו מחקרי זה (התאמת הייפר-פרמטרי ג'נרוט) קיים כבר איזה 4 שנים ויצאו לפחות 10 מאמרים בנושא (שלא ידעתי). אז המאמר הזה הוא המשך של כמה מאמרים שלא סקרתי בזמנו). אוקיי אז כאמור המאמר מנסה לאפטם את T בהינתן ההקשר. המחברים מניחים שאנו בוחרים T מסט טמפרטורות סגור T_1,..., T_k המחברים מציעים לאמן רשת M_t(נקרא Adaptive Decoder במאמר) החוזה את T האופטימלי בהתבסס על ייצוגי טוקני ההקשר. כלומר הרשת פולטת התפלגות מעל T_1,..., T_k (כלומר סופטמקס).

למעשה התפלגות כזו היא ממשקלת (משנה לפי התפלגות הטמפרטורות הנוצרת על ידי M_t) את התפלגות הסופטמקס מעל מילון הטוקנים שממנו מודל שפה מגנרט טקסט. כמובן ניתן לאמן M_t בכמה דרכים על דאטהסט נתון במטרה למקסם את הנראות(likelihood) של הדאטה (לדעתי נעשה במאמרים קודמים). המאמר מציע לעשות את בשיטת DPO הלקוחה לעולם למידה עם חיזוקים עם RL (קראו סקירה מ 18.11.24 כדי לרענן מה זה). רק אזכיר שבשיטה זו מבצעים יישור (alignment) של מודל שפה על דאטהסט של תשובות רצויות ופחות רצויות.

אז המחברים מציעים להכליל את השיטה הזו עבור המקרה שאנו לא רק מאמנים את המודל אלא גם המודל לקביעת התפלגות טמפרטורה. הדאטהסט של תשובות וטמפרטורות רצויות נבנה על ידי דגימה של מודל שפה בטמפרטורות שונות ובחירה של התשובה הטובה ביותר והגרועה ביותר או עלי ידי מודל אחר או על ידי מתייגים אנושיים. ואז בדומה ל-DPO בונים פונקציית לוס שמעדכנת את מודל השפה וגם M_t יחד. הרי ניתן לראות ב-M_t מודל דגימה ממילון הטוקנים כאשר כל טוקן הוא טמפרטורה T_k. אז זה הכללה די מתבקשת. המחברים גם מציעים פונקציית לוס שמעדכנת רק את M_t באותה הצורה.

לבסוף המאמר מציע פונקצית לוס המאפטמת מודל שפה יחד עם M_t כאשר התפלגות של הטוקנים (של מודל השפה) מבוטאת דרך מרגינליזציה שלה מעל התפלגות הטמפרטורות דרך נוסחת בייס. כלומר מיישרים את המודל לתעדף **רק** תשובות רצויות באופן ישיר אבל יחד עם זאת גם M_t מתעדכן.

https://arxiv.org/abs/2411.09661

⚡🚀המאמר היומי של מייק -21.11.24: 🚀⚡

Unfamiliar Finetuning Examples Control How Language Models Hallucinate

מאמר של סרגיי לווין האגדי מאוניברסיטת טורונטו שידוע יותר בתרומתו האדירה לפיתוח שיטות מבוססות למידה עם חיזוקים (RL) ליישומי רובוטיקה. הפעם הוא עם קבוצתו חוקר את תופעת הזיות (hallucinations) של מודלי

שפה. הזיה זה מושג מאוד רחב בהקשר מודלי שפה ובגדול (מאוד) ניתן להגדירו בתור מתן תשובה לא נכונה (בעיקר עובדתית) על ידי מודל שפה.

מאז שמודלי שפה נכנסו לחיינו בשנים האחרונות תופעה זו נחקרה באופן נרחב בעשרות (אם לא מאות) מאמרים. המאמר שנסקור היום חוקר סיבות לתופעה זו וגם מציע דרכים להתמודד איתה. החוקרים טוענים הסיבה להזיות טמונה בניסיון להקנות למודל ידע חדש במהלך טיוב (finetuning). המחברים טוענים שהמודל נוטה ללמוד פחות טוב את העובדות הנמצאות בדאטהטס של FT (נקרא לו D_FT) שלא מיוצגות מספיק טוב בדאטהסט הגדול ששימש את המודל לאימון מקדים (נקרא לדאטהסט זה בתור D_PR). עובדות (ושאלות עליהם) נקראות לא-מוכרות במאמר.

בפרט המאמר משער (ומראה אמפירית) שעבור שאלה על עובדה לא q המודל מוציא תשובה שהיא סוג של תשובה ממוצעת עבור כל השאלות הלא מוכרות מ-D_FT. כלומר
כזו שממזערת את פונקציית הלוס הממוצעת על כל השאלות הלא מוכרות האלו מ-D_FT. ומכיוון שרוב התשובות ב-D_FT מנוסחת היטב ובאנגלית רהוטה אנו מקבלים תשובות יפות אך לא נכונות בהחלט ממודל שפה לשאלות לא מוכרות.

בגדול הרעיון העיקרי שהמחברים מציעים לתיקון המצב הזה הוא ללמד את המודל להגיד "לא יודע" בצורה ברורה על שאלות לא מוכרות (כלומר במקרים שהוא אכן לא יודע). אחת הדרכים לעשות זאת היא קודם לזהות שאלות לא מיוצגות מספיק ב-D_FT (על ידי ניתוח שכיחותם או אנטרופיה של הלוגיטים של תשובה המודל לשאלות אלו - ד"א שניהם לא אידאליים באספקט הזה). לאחר מכן במקום לאמן מודל לענות תשובות נכונות לשאלות אלו (שהוא לא מסוגל ללמוד), תשובות אלו מוחלפות ב-D_FT על ידי תשובות נייטרליות בסגנון "אני לא יודע". כמובן אפשר להוסיף ל-D_FT מלא שאלות הלא מוכרות ב-D_PR עם תשובות אלו.

הדרך השנייה היא לאמן מודל עם שיטות של RLHF עם שינוי של פונקציית תגמול (reward) המקטין קנס על תשובות נייטרליות ומשאיר את שאר התגמולים כמו שהם. המחברים מראים (אמפירית) שבמקרה זה המודל יותר "שמח" לתת תשובות נייטרליות לשאלות לא מוכרות. המאמר מציע שיטה המורכבת מ-4 שלבים לאימון RLHF לשיפור יכולת המודל להגיד "לא יודע":

1. עושים FT רגיל
2. דוגמים את המודל עם שאלות מוכרות ולא מוכרות
3. בונים פונקצית תגמול הקונסת את המודל יותר על תשובות לא נכונות לשאלות לא מוכרות (וקנס מאוד נמוך או 0 על תשובות מתחמקות)
4. אימון RLHF עם פונקצית התגמול מסעיף 3.

מאמר נחמד שהשאיר בי טעם לראות את ההמשך.

https://arxiv.org/abs/2403.05612

⚡🚀המאמר היומי של מייק -22.11.24: ⚡🚀

The Unreasonable Ineffectiveness of the Deeper Layers

מאמר קליל שלא יקשה עליכם יותר מדי בסופ"ש. המאמר מציע דרך מאוד פשוטה לקצץ שכבות במודלים המבוססים על ארכיטקטורת הטרנספורמרים. אתם בטח זוכרים שמודלי שפה שלנו וגם לא מעט מודלים בדומיינים אחרים מבוססים על טרנספורמרים שמורכבים מבלוקים שכל אחד מהם מורכב ממנגנון attention ושתי שכבות feed-forward (השנייה מהן לינארית). בנוסף יש שכבות נרמול וחיבורי residual (כלומר הפלט של כל שכבה מחובר יחד עם הפלט של השכבה הקודמת).

מודלי שפה מודרניים מכילים עשרות רבות של בלוקי טרנספורמרים שכמובן משליך על כמות הזמן והמשאבים הנדרשים להפעלתם, בעיקר במשימות גנרוט. כאמור המאמר שנסקור היום מציע דרך לקצץ כמה בלוקי טרנספורמרים רצופים שכמובן יקטין את זמן חישוב שנדרש ליצירה הפלט. אבל איזה בלוקים לבחור כך שהפגיעה בדיוק המודל תהיה מינימלית.

מכיוון שהגרף החישובי של הטרנספורמר מורכב מלא מעט חיבורי residual טבעי לבחור בלוקים רצופים שלא מוסיפים הרבה לפלט הבלוק הנמצא לפניהם במודל. כלומר אם הדלתא שנותנים הבלוקים האלו זניחה אז ניתן להעיף אותם בלי פגיעה רצינית בביצועים.

האבל איך ניתן לבדוק את זה? האמת יש לא מעט דרכים לבדוק את זה ומאמר בחר להשוות את הפלט של הבלוק l עם הפלט של הבלוק l+n (אנו מוחקים n בלוקים רצופים) באמצעות מודיפיקציה קטנה של מרחק קוסיין (החליפו cos ב-arccos וחילקו ב-pi כדי לגרום למדד הזה להיות בין 0 ל 1). באופן הגיוני n בלוקים עם דמיון גבוה מאוד לבלוק שקודם להם (מבחינת הפלט) נבחרים בתור מועמדים טובים לקיצוץ (כלומר בוחרים בלוק התחלתי l ומספר בלוקים לקיצוץ n עם הדמיון הגבוה ביותר). הדמיון מחושב על ייצוג הטוקן האחרון עבור כמות דאטה גדולה.

לאחר המחיקה ניתן לעשות למודל פיין טיון קליל ולטענת המחברים ניתן למחוק ככה על חצי שכבות טנרספורמים (במודלי שפה) בלי פגיעה רצינית בביצועים).

https://arxiv.org/abs/2403.17887

⚡🚀המאמר היומי של מייק -23.11.24: 🚀⚡

Table Meets LLM: Can Large Language Models Understand Structured Table Data? A Benchmark and Empirical Study

היום אני סוקר מאמר בנושא שמזמן לא נגעת בו(בסקירות) והוא דאטה טבלאי. המאמר בוחן שאלה מרתקת - האם מודלי שפה גדולים (LLMs) כמו GPT באמת מבינים מידע מובנה בטבלאות?

קצת: רקע

בשנים האחרונות, LLMs הפכו לכלי חשוב בעיבוד שפה טבעית. אבל בעוד שהם מצוינים (סוג של) בהבנת שפה טבעית (בצורה של טקסט), יכולתם להבין מידע בצורה של טבלאות עדיין לא נחקרה לעומק וזה בדיוק מה שהחוקרים מנסים לעשות במאמר המסוקר

מה החוקרים עשו?

החוקרים פיתחו מדד חדש שנקרא SUC (Structural Understanding Capabilities) שבוחן את היכולות של מודלים להבין מבנה של טבלאות. המדד כולל שבע משימות שונות:

1. זיהוי גבולות טבלה
2. איתור תאים ספציפיים
3. חיפוש הפוך (מיקום לערך)
4. אחזור עמודות
5. אחזור שורות
6. זיהוי גודל טבלה

7. זיהוי תאים ממוזגים

הם בדקו את GPT-3.5 ו-GPT-4 במשימות אלו תוך שימוש בפורמטים שונים של קלט (HTML, JSON, CSV ועוד).

מה הם גילו?

התוצאות מפתיעות! הנה הנקודות העיקריות:

- HTML מתגלה כפורמט "הנוח" ביותר להצגת טבלאות ל-LLMs
- המודלים הראו יכולות טובות במשימות יחסיות מורכבות (זיהוי גבולות טבלה, זיהוי תאים ממוזגים) אך נכשלו במשימות פשוטות (זיהוי גודל טבלה, אחזור שורה פשוט, חיפוש תא בודד)
- הביצועים השתפרו משמעותית עם דוגמה אחת (one-shot) לעומת אפס דוגמאות

החידוש המרכזי: Self-augmented Prompting

החוקרים פיתחו שיטה חדשה שנקראת "self-augmented prompting" שמשפרת את ביצועי המודלים. השיטה מבקשת מהמודל תחילה לזהות מידע קריטי בטבלה (כמו טווחי ערכים) ואז משתמשת במידע הזה כדי לשפר את התשובה הסופית. זה מאפשר שיפור די רציני במספר בנצ'מארקים)

סיכום:

אני חייב להגיד שהמאמר הזה מרתק. הוא מראה שלמרות ההתקדמות העצומה ב-LLMs, יש עדיין פערים משמעותיים ביכולת שלהם להבין מידע מובנה. זה מזכיר לנו שלמרות שהמודלים האלה מרשימים, הם עדיין רחוקים מהבנה אנושית אמיתית של מבנים ויחסים בין דאטה.

החוקרים עשו עבודה לא רעה בפיתוח מדדים ושיטות שיעזרו לקהילה להמשיך לשפר את היכולות האלה. השיטה החדשה שלהם ל-prompting היא פשוטה אבל אפקטיבית, וזה בדיוק מה שאנחנו צריכים - פתרונות פרקטיים שאפשר ליישם מיד.

מילה אחרונה

אם אתם עובדים עם טבלאות ו-LLMs, המאמר הזה הוא חובה. הוא מספק תובנות מעשיות וכלים שימושיים. הקוד והדאטה זמינים ב-GitHub, אז אתם יכולים להתחיל לשחק עם זה ישר.

מעניין במיוחד יהיה לראות איך הממצאים האלה ישפיעו על הדור הבא של מודלי שפה. האם נראה מודלים שמתוכננים במיוחד להבנת מידע מובנה?

⚡🚀המאמר היומי של מייק -26.11.24: 🚀⚡
Is DPO Superior to PPO for LLM Alignment? A Comprehensive Study

המאמר מציג ניתוח מעמיק של 2 שיטות מרכזיות ליישור מודלי שפה גדולים עם העדפות אנושיות: (Direct Preference Optimization (DPO) ו- Proximal Policy Optimization (PPO).

1. **רקע ומוטיבציה:**

- קיימת סתירה מעניינת: יישומים מסחריים מצליחים כמו ChatGPT משתמשים ב-PPO, בעוד שבספרות האקדמית DPO משיג תוצאות מובילות.

- מחקר זה בודק האם DPO אכן עדיף על PPO ומה גורם לביצועים הנמוכים של PPO במדדים אקדמיים.

2. ממצאים תיאורטיים:

- DPO סובל ממגבלות מהותיות הקשורות להטיה כלפי תשובות מחוץ להתפלגות הדאטה (out-of-distribution או ODD)

- הביצועים של DPO מושפעים משמעותית מהמרחק בין ההתפלגות בין ההתפלגות ההתחלתית של המודל לדאטה המשמש לאימון RLHF (העדפות אנושיות)

3. **שיפורים ב-PPO:**

החוקרים זיהו 3 גורמים קריטיים לשיפור ביצועי PPO:

- נרמול של פונקציית היתרון (Advantage Normalization) - משמש לעדכון של משקלי המודל ב-PPO

- אימון עם באצ'ים גדולים

- עדכון הדרגתי של המודל המאומן באמצעות ממוצע נע מעריכי של משקלי המודל מהאיטרציות עדכון הקודמות

4. **תוצאות ניסיוניות:**

- PPO משיג ביצועים עדיפים בכל המשימות שנבדקו

- במשימות מאתגרות של יצירת קוד, PPO משיג תוצאות state-of-the-art

- מודל PPO עם 34B פרמטרים משיג שיפור של 10% בהשוואה ל-AlphaCode-41B באחד הדאטהסטים

5. מסקנות עיקריות:

- למרות הפופולריות הגוברת של DPO, השיטה סובלת ממגבלות מהותיות

- עם היישום הנכון של הטכניקות שזוהו, PPO יכול להשיג ביצועים מצוינים

- המחקר מספק תובנות חשובות לגבי האופן שבו יש ליישם PPO ביעילות

6. חשיבות המחקר:

המאמר תורם תרומה משמעותית להבנת היתרונות והחסרונות של שיטות יישור שונות, ומספק הנחיות מעשיות ליישום מוצלח של PPO. התוצאות מאתגרות את ההנחה הרווחת ש-DPO עדיף, ומדגישות את החשיבות של יישום נכון של PPO.

סיכום:

לסיכום, זהו מחקר חשוב המספק תובנות מעשיות ותיאורטיות חשובות לתחום יישור(alignment) של מודלי שפה גדולים עם העדפות אנושיות.

https://arxiv.org/abs/2404.10719

⚡🚀המאמר היומי של מייק -27.11.24: ⚡🚀
The Illusion of State in State-Space Models

מאמר חשוב זה בוחן את המגבלות התיאורטיות של State Space Models או (SSMs), אשר צמחו כארכיטקטורה חלופית לטרנספורמרים עבור מודלי שפה גדולים. המחברים מדגימים שלמרות עיצובם שנראה Recurrent ובעל מצב (כלומר stateful), למעשה SSMs (כמו טרנספורמרים) מוגבלים באופן בסיסי ביכולתם לבטא חישוב "רציף", מכיוון שאינם יכולים לחשב דבר מחוץ למחלקת המורכבות TC0. משימות ממחלקת TC0 מוגדרות ככאלו שניתן לייצגן עם שרשראות בוליאניות בסיסיות (וחישובי סף ו- majority vote) בעומק סופי (למשל חיבור של מספרים, מכפלה או מיון של n מספרים). מדובר במחלקה הכי "פשוטה" בהיררכיה של תורה סיבוכיות circuit (כלומר circuit complexity).

משמעות הדבר היא ש-SSMs אינם יכולים לפתור בעיות מסוג permutation composition ש- RNNs בעלות שכבה אחת מסוגלות לפתור.

תרומות מרכזיות של המאמר:

1. **ניתוח תיאורטי:**
   - מוכיח שגם SSMs לינאריים וגם SSMs בסגנון Mamba מוגבלים למורכבות חישובית TC0
   - מראה ש-SSMs אינם יכולים לפתור בעיות שלמות-NC1 (משימות שניתן לייצג אותן עם פעולות בוליאניות בעומק לוגריתמי ממימד הבעיה - מספר משתנים בגדול) כמו הרכבת תמורות. כלומר לא עומק סופי כמו ב- TC0.
   - מדגים ש-SSMs אינם יכולים לעקוב במדויק אחר מהלכי שחמט, לכתוב קוד מורכב, או לעקוב אחר ישויות בנרטיבים.

2. **בדיקות אמפיריות שבוצעו על ידי מחברים המאמר:**
   - מספק ראיות ניסיוניות המראות ש-SSMs בסגנון Mamba וטרנספורמרים מתקשים במשימות permutation composition.
   - מראה ש-SSMs דורשים עומק גדל כדי "לטפל" ברצפים ארוכים יותר למידול פעולות קבוצה "תמורתיות"
   - מדגים ש-RNNs בשכבה יחידה יכולים לפתור משימות אלו ש-SSMs אינם יכולים (כנראה בגלל לינאריות בין המעבירים של המצבים החבויים ב-SSMs).

3. **שכלולי ארכיטקטוניות המוצעים במאמר:**
   - מציע 2 דרכים להרחיב SSMs מעבר למגבלות TC0: הוספת אי-ליניאריות (RNN-SSM) והפיכת מטריצות המעבר לתלויות בקלט (WFA-SSM) - שכלול של ממבה המוסיף אי לינאריות למטריצה A שנותרה קבועה בממבה.

השפעה והשלכות של המאמר:

- מאתגר הנחות לגבי יתרונות SSMs על פני טרנספורמרים
- מצביע על גישות היברידיות פוטנציאליות המשלבות ארכיטקטורות שונות

- פותח כיוונים חדשים לפיתוח ארכיטקטורות עם יכולת ביטוי משופרת ליישומי עיבוד שפה טבעית ועבור דומיינים נוספים
- מדגיש את חשיבות הניתוח התיאורטי של התמאת של ארכיטקטורת מודל למשימה ספציפית שהוא מתוכנן לפתור

סיכום:

מאמר תורם הן מבחינה תיאורטית והן מבחינה מעשית להבנת ארכיטקטורות של רשתות נוירונים. הניתוח התיאורטי הקפדני, בשילוב עם ראיות אמפיריות תומכות, מספק תובנות חשובות לגבי המגבלות הבסיסיות של SSMs.. בעוד שחלק מהתוצאות התיאורטיות מסתמכות על הנחות תיאורטיות של מורכבות, ההשלכות המעשיות נתמכות היטב בראיות אמפיריות.

https://arxiv.org/abs/2404.08819

🚀⚡ :המאמר היומי של מייק -28.11.24🚀⚡
Parameter-Efficient Fine-Tuning with Discrete Fourier Transform

רקע: PeFT:

נתחיל את הסקירה ברענון קצרצר לגבי שיטות טיוב (fine-tuning) חסכוניות של מודלי שפה. PeFT הינה משפחה של שיטות המאפשרות טיוב של מודלים גדולים (בפרט מודל שפה) תוך שימוש במספר מצומצם של פרמטרים, מה שחוסך משמעותית במשאבי חישוב וזיכרון.

רקע: LoRA:

אחת השיטות הפופולריות ביותר ב-PeFT, הנקראת LoRA, מקפיאה את משקולות המודל ומאמנת מטריצות תוספת לכל שכבה של הטרנספורמטורים. כל מטריצת תוספת נלמדת הינה בעלת בדרגה נמוכה (low-rank), כך שניתן לייצגה על ידי מכפלה של שתי מטריצות קטנות (במימד האמצעי של המכפלה).

היתרון המרכזי של LoRA הוא שהיא מאפשרת להתאים מודלים גדולים למשימות ספציפיות תוך אימון של חלק קטן (נגיד 1% מכלל הפרמטרים שלו), מה שהופך אותה ליעילה במיוחד. שיטה זו הוכיחה את עצמה כאפקטיבית במיוחד בהתאמת מודלי שפה גדולים למשימות ספציפיות. בנוסף, LoRA מאפשרת החלפה מהירה בין גרסאות שונות של המודל המטויב, מכיוון שניתן לשמור את המטריצות הקטנות בנפרד מהמודל המקורי.

שיטה מוצעת:

הרעיון המרכזי הוא להסתכל על שינויי המשקולות של רשת הנוירונים כמו על תמונה או אות, ולייצג אותם בציר התדר במקום ערכים ישירים. כשאנחנו רוצים לטייב את המודל, במקום לשנות את כל המשקולות באופן ישיר (שדורש המון פרמטרים), אנחנו:

1. מגדירים מראש כמה נקודות דגימה במרחב התדרים שבהן נרצה להתמקד. זה כמו לבחור אילו תדרים אנחנו רוצים לשמור בייצוג הדחוס שלנו. זה נעשה על ידי בחירת מטריצת תדרים קבועה (לא נלמדת) E בגודל 2xn המשמשת לבניית ייצוג של מטריצת תוספת. מטריצה זו היא קבועה לכל השכבות של הטרנספורמרים.

2. לומדים וקטור c בגודל n (לכל שכבה) כאשר דרך שילובו עם E בונים את מטריצת התוספות בתחום התדר F (הסבר לאיך זה נבנה לא נראה ברור במאמר)

3. מעבירים את F דרך Gaussian bandpass filter (כלומר דוגמים בעיקר תדרים נמוכים, הנמצאים קרוב למרכז המטריצה).

4. מעבירים את מטריצת F לתחום הזמן (הרגיל) ומשתמשים בה בדיוק כמו ב-LoRA

יתרונות השיטה המוצעת:

היתרון הגדול הוא שתדרים הם דרך מאוד יעילה לייצג מידע (צריך 2n+ Ln משקלים כאשר L מספר השכבות במודל). בדיוק כמו שאפשר לדחוס תמונה או מוזיקה על ידי שמירת התדרים החשובים ביותר, כאן אנחנו יכולים לייצג שינויים מורכבים במשקולות באמצעות מספר קטן מאוד של תדרים.

זה עובד טוב(כנראה):

- שינויים במשקולות נוטים להיות "חלקים" יחסית, כלומר יש בהם מבנה שאפשר לתפוס טוב עם תדרים

- הבסיס המתמטי של פורייה הוא אורתוגונלי, מה שאומר שכל תדר מוסיף מידע ייחודי

- אנחנו יכולים לבחור מראש כמה תדרים אנחנו רוצים לשמור, ובכך לשלוט ישירות בכמות הפרמטרים

סיכום:

בניגוד לשיטות אחרות שמנסות להקטין את כמות הפרמטרים על ידי הגבלת הדרגה של המטריצות (כמו LoRA), הגישה הזו מסתכלת על הבעיה מזווית שונה - דרך עדשת התדרים, ומצליחה להשיג דחיסה משמעותית יותר.

https://arxiv.org/abs/2405.03003

⚡🚀המאמר היומי של מייק -29.11.24: ⚡🚀
In-Context Learning with Long-Context Models: An In-Depth Exploration

המאמר מציג מחקר אמפירי מקיף של למידה in-context או ICL עם מודלי שפה בעלי חלון הקשר ארוך. אזכיר שעם ICL המודל מקבל כמה דוגמאות המדגימות פעולות מסוימות ולאחר מכן המודל מתבקש לבצע פעולה זו על דוגמאות חדשות.

ממצאים חדשים על התנהגות של ICL ל-LLMs בעלי חלון הקשר ארוך:

1. שיפור ביצועים מתמשך: עלייה משמעותית בביצועים כאשר מעלים את מספר הדוגמאות בהדגמה מ-10 ל-1000 דוגמאות

2. רגישות פחותה לסדר: השפעת סדר הדוגמאות יורדת ב-50% ב-1000 דוגמאות לעומת 10(עבור סידור אקראי)

3. ירידה ביתרון ה-RAG: היתרון של RAG פוחת משמעותית עם יותר דוגמאות

4. השפעת קיבוץ דוגמאות לפי קטגוריות: מיון דוגמאות לפי קטגוריות פוגע יותר בביצועים ככל שחלון ההקשר גדל

5. יעילות אורכי attention קצרים: ניתן להשיג ביצועים דומים עם מנגנון attention קצר יחסית המשתרע ל-50-75 דוגמאות

6. השוואה לטיוב (fine-tuning): למידת in-context לאורכי חלון הקשר ארוכים לרוב משתווה או עולה על טיוב עם מעט דוגמאות אולם הטיוב מנצח כאשר יש מספיק דוגמאות.

https://arxiv.org/abs/2405.00200

⚡🚀המאמר היומי של מייק -30.11.24: 🚀⚡

Fishing for Magikarp: Automatically detecting under-trained tokens in large language models

מאמר מעניין מבית חברת cohere, אחת החברות שמפתחות מודלי שפה foundational.

רקע:

המאמר חוקר סוגיה מעניינת של טוקנים לא מאומנים מספיק (under-trained) כלומר שלא נמצאים (או נמצאים בכמות מזערית) בדאטהסט אימון של המודל. סיבה אפשרית לקיום טוקנים כאלו נעוצה בעובדה שמילון הטוקנים לא תמיד נבנה על בסיס הדאטהסט שהמודל מאומן עליו.

מילון הטוקנים בנוי על דאטהסט קטן הרבה יותר מדאטהסט אימון העצום של המודל בשלב אימון מקדים (pretraining): הרי בניית מילון טוקנים עם אלגוריתמים קיימים על דאטהסט של עשרות טריליוני טוקנים איננה פיזיבילית חישובי. בגדול מאוד בוחרים תת-מילים "השכיחים ביותר" בדאטהסט (כולל סימני פיסוק וכדומה) לפי שיטה מסוימת (היום השיטה הפופולרית היא Byte-Pair Encoding או BPE, שיטה טוקניזציה נוספת נקראת WordPiece). וההבדלים בסט לטוקניזציה לבין זה לאימון המודל עלול להוביל ליצירת טוקנים מוזרים כמו _TheNitrome.

הנוכחות של טוקנים שלא אומנו מספיק במודל מובילה למספר בעיות, כולל בזבוז קיבולת בטוקנייזר ופגיעה ביעילות המודל. בנוסף הם עלולים לגרום לפלט לא רצוי ולשבש אפליקציות downstream במיוחד בעידן שבו מודלי שפה משתמשים יותר ויותר בנתונים חיצוניים. כמובן שטוקנים כאלו "מזמינים" jailbreaks למיניהם. למרות שנעשתה עבודה מסוימת בזיהוי טוקנים בעייתיים אלה, עדיין חסרות שיטות אוטומטיות אמינות ומוסברות היטב שנבדקו על מגוון רחב של מודלים.

פרטי מחקר:

המאמר מציע לזיהוי טוקנים undertrained כאלו באמצעות טרנספורמציה מסוימת של מטריצה unembedding U כלומר המטריצה הממפה את ייצוג הטוקן לווקטור המכיל התפלגות הסתברותית עבור כל הטוקנים במילון.

המחברים מציינים כי פונקציית הלוס באימון ממוזערת כאשר ההסתברות של טוקנים שאינם בשימוש נחזית כ-0, ללא קשר לקלט, מה שגורם ללוגיטים שלהם להתכנס למינוס אינסוף. המאמר משער שהמודל יכול להשיג חיזוי כזו (לא תלוי בקלט) באמצעות חיסור של וקטור קבוע c משורות של U, מה שמוביל לתרומה שלילית קבועה לערכי הלוגיטים של טוקנים שאינם בשימוש.

המחברים מציעים את האלגוריתם הבא לזיהוי טוקנים undertrained:

- מגדירים קבוצה S של טוקנים חשודים ל-undertrained (כלומר אינדקסים של שורות ב-U)
- חשב את הרכיב העיקרי(principal component) הראשון c של U כאומדן לרכיב קבוע c. מכיוון שפונקציית הסופטמקס אינה משתנה להסטות קבועות, יש להקפיד להסיר רכיב קבוע כזה כדי למקסם את ההפרדה של טוקנים שאינם בשימוש.
- הסר אותו כדי לקבל $U' = U - (c^T*U)U$.
- חשב את וקטור האמבדינגס הממוצע של הטוקנים שאינם בשימוש $u\_oov = U'_i, i \in S$.
חשב את מרחקי הקוסיין (או מרחק L2) בין u_oov לבין שאר השורות ב-U'.

הטוקנים שהמרחק הזה קטן יחסית לאחרים (באחוזון 2 נגיד) חשודים להיות טוקנים שאומנו מספיק. המאמר מצליב הסתברויות של הטוקנים החשודים ל-undertrained ומראה שהן קטנות מאוד ומשתנים לאט מאוד (בעיקר בגלל weight decay) באופן עקבי לאורך האימון (ללא קשר לקלט).

https://arxiv.org/abs/2405.05417

⚡🚀המאמר היומי של מייק -02.12.24: ⚡🚀
Autoregressive Model Beats Diffusion: Llama for Scalable Image Generation

היסטוריה:

סקירה היום אעשה חזרה קטנה בזמן (מבחינתי) ואסקור מאמר בנושא הראיה הממוחשבת. פעם הייתי סוקר אותם בתדירות גבוהה יותר אולם לאחרונה רוב המאמרים שאני סוקר שייכים לדומיין הטקסטואלי כלומר NLP. לא אגלה לכם סוד אם אגיד לכם שהיום מודלי דיפוזיה (לרוב לטנטיים) די השתלטו על תחום גנרוט דאטה ויזואלי (כלומר תמונות ווידאו).

אולם לפני 3-4 שנים המצב בדומיין הויזואלי (בחלקו הגנרטיבי) היה די שונה. היו בו גם VAE שזה Variational AutoEncoders, גם זרימות מנורמלות (Normalized Flows) אבל מי ששלט בו באופן די מוחלט היה כמובן GANs כלומר Generative Adversarial Networks. וכמובן היו שילובים די מעניינים של השיטות הנ״ל שהגיעו לביצועים די יפים כמו VQGAN שזה שילוב של VAE ו-GAN.

רקע:

המאמר שנסקור היום מחזיר לחיים את VQGAN וטוען שניתן להגיע לתוצאות טובות יותר איתו (עם שכלול קל) ממודלי דיפוזיה גנרטיביים באותם הגדלים (= מספר פרמטרים). זו הצהרה די חזקה שמצריכה להבין מה המחברים שכללו ב-VQGAN שהוצע לפני 4 שנים.

קודם כל אסביר בקצרה איך עובד VQGAN (סקרתי אותו בעבר הרחוק בהרחבה) אז תוכלי לקפוץ לשם להסברים מפורטים יותר. בגדול VQGAN מורכב מאנקודר שמטרתו לקודד (במרחב הלטנטי) את הפאצ'ים של תמונה, codebook, המורכב ומספר גדול של וקטורים המקודדים את הפאצ'ים האלו ודקודר שלמעשה הופך את ייצוגי פאצ'ים אלו (וקטורים) לפאצים המרכיבים תמונה.

אחרי הקידוד של פאץ' על ידי האנקודר הווקטור הכי קרוב (לפי מרחק L2 לדעתי) נבחר מה-codebook והוא מוזרם לדקודר (יחד עם עם הוקטורים הפאצ'ים האחרים). האנקודר וה-codebook מאומנים להחזיר וקטורים כמה שיותר קרובים אחד לשני (יש שם stop-gradient גם) והדקודר מאומן לשחזר את התמונה (נבדק לכל פאץ' בנפרד וגם יחד) בצורה המיטבית (נמדד על ידי דמיון perceptual נקרא LPIPS וגם יש לוס של גאן בפנים עם הדיסקרימנטור).

מה המאמר עשה:

אבל איך נשתמש בכל לגנרוט? לאחר סיום אימון של VQGAN, לוקחים את כל הייצוגים הלטנטיים של התמונות מהדאטסט ומאמנים דקודר של הטרנספורמר לחזות ייצוג של פאץ' בהינתן הפאצ'ים הקודמים. ופה נכנסים לנו LLM שאנו כה אוהבים כי המחברים מאמנים אחד הלמות (LLAMA) למשימה הזו. הרי יש לנו מילון (codebook) כמו בשפה טבעית רק שבמקום הטוקנים הרגילים יש לנו טוקנים ויזואליים.

וזה עובד לא רע (לפי הבדיקות שהם עשו)...

https://arxiv.org/abs/2406.06525

⚡🚀המאמר היומי של מייק -04.12.24: ⚡🚀
KAN: Kolmogorov–Arnold Networks

האמת שזה די מחדל שב 7 חודשים מאז שהמאמר הזה התפרסם, לא סקרתי אותו. יש לו כרגע כבר 400 ציטוטים והיד עוד נטויה. אני באופן אישי מאוד אוהב מאמרים המבוססים על טענה מתמטית מוכחת ולצערי אין לנו הרבה כאלו בתקופה האחרונה.

המאמר הדי מדובר הזה מציג ארכיטקטורה חדשה המבוססת על משפט קולמוגורוב ארנולד שטוען שכל פונקציה רבת משתנים רציפה ניתנת לייצוג כסכום (כפול) של פונקציות של משתנה אחת. במילים פשוטות כל פונקציה ניתן לייצג בתור סכום של סכומים של פונקציות שכל אחת מהן היא של משתנה אחת בלבד.

משפט זה הוא "מקביל" ל- Universal Approximation Theorems (יש כמה כאלו) שאומרת שניתן לייצג כל פונקציה (המקיימת תנאי לא מגבילים במיוחד) על יד רשת נוירונים בעלת עומק 2 או יותר שכבות. רשתות נוירונים של היום בנויים בהתבסס על UAT (בגדול) והמאמר המסוקר מציע לבנות אותם בהתבסס על משפט KA. באופן די טבעי זה קיבל שם כן.

המודל KAN בנוי משכבות שכל אחד מהן סכום של פונקציות נלמדות (כלומר הפרמטרים בהם הם אלו שנלמדים על הדאטהסט). כל פונקציה נלמדת כזו מורכבת מצירוף לינארי של כמה b-splines (עוד פונקציה ללא פרמטרים הנקראת silu(x).

ב-ספליין B זה פונקציה המוגדרת באינטרוול, המחולק לכמה מקטעים (נקרא grid) שמהווים פרמטרים של הבי-ספליין. B המורכב מכמה פולינומים (מדרגה 3 בד"כ) כך שלכל מקטע יש פולינום משלו. בי-ספליין משמשים לקירוב של פונקציות כאשר המקדמים לפולינום בכל מקטע נקבעים כדי למקסם את דיוק הקירוב. אז ב-KAN לומדים את את פרמטרי הגריד במטרה למזער את פונקציית הלוס של הבעיה.

וזהו זה - היה זמנו לא מעט התלהבות סביב הארכיטקטורה החדשה הזו אבל התברר שהאימון של KAN הוא לא פשוט בכלל ולא תמיד מתכנס. אבל זה לא הפריע לא לקבל 400 ציטוטים בחצי שנה עם עשרות רבות מאמרים המשך שכנראה אסקור כמה מהם. בינתיים אני לא איבדתי תקווה ב-KAN...

https://arxiv.org/pdf/2404.19756

⚡🚀המאמר היומי של מייק -05.12.24: ⚡🚀
Memory3: Language Modeling with Explicit Memory

א. רעיון כללי:

המאמר מציע זיכרון מפורש (explicit memory או EP) כתוספת לארכיטקטורה של מודלים לשוניים. בניגוד לאופי הסטטי של פרמטרי המודל או הזיכרון הזמני (משקלי K ו- V), הזיכרון המפורש פועל כמחסן ידע מובנה ודינמי, הניתן לאחזור מחוץ למודל שפה.

זיכרון מפורש מיועד ל"שיפור טרייד-אוף" בין גודל של LLMs לבין ביצועיהם. באמצעות החצנת ידע פחות מופשט (כמו עובדות, נתונים, חוקים ספציפיים לתחום) אל תוך EM, המודל נמנע מהגדלה משמעותית של פרמטרי המודל, תוך שמירה או אף שיפור של ביצועים. חידוש זה לא רק משפר את היעילות החישובית, אלא גם הופך את המערכת למודולרית. עדכוני ידע אינם מחייבים אימון מחדש של כל המודל, מה שמדמה תהליך למידה אנושי שבו מידע חדש נשמר מבלי לשנות את הפונקציות הקוגניטיביות הבסיסיות.

ב. היררכיית זיכרון מוצעת

היררכיית הזיכרון שהוצעה במאמר שואבת השראה ממערכות קוגניטיביות אנושיות, שבהן הזיכרון לטווח ארוך מסווג לפי נגישות ותדירות שימוש. המחברים מעצבים מסגרת זו כדי להקצות ידע אסטרטגית ב- 3 רמות:

1. **טקסט פשוט (עלויות קריאה גבוהות, עלויות כתיבה נמוכות)**:
   - מתאים למידע שניגשים אליו באופן נדיר, אחסון טקסט פשוט שומר על קלילות המערכת הכוללת. אחזור מזיכרון זה פחות יעיל אך משמש כגיבוי לשאילתות נדירות.

2. **זיכרון מפורש (עלויות מאוזנות)**:
   - ידע הנמצא בשימוש תדיר יותר אך לא קריטי (כמו ידע מופשט על השפה) נשמר ב-EM, המאזן בין מהירות האחזור(retrieval) לעלויות האחסון. האינטגרציה שלו עם מנגנוני attention דלילים מבטיחה שרק חלקי הזיכרון הרלוונטיים ביותר יופעלו, מה שמשפר את יעילות האינפרנס.

3. **פרמטרי מודל (עלויות קריאה נמוכות, עלויות כתיבה גבוהות)**:
   - שמור לידע מופשט המהווה ליבה ליכולות האינפרנס הבסיסיות של המודל. עדכונים בשכבה זו מתבצעים באימון, מה שהופך אותם ליקרים חישובית.

היררכיה זו מאפשרת ל-Memory3 לתעדף הקצאת משאבים בצורה דינמית, ומבטיחה שהעלויות החישוביות יישארו ניתנות לניהול תוך שמירה על ביצועים גבוהים. עיצוב זה רלוונטי במיוחד ליישומים הדורשים עדכוני ידע בזמן אמת, כגון מערכות תמיכת לקוחות או בוטים מותאמים לתחום ספציפי.

ג. ארכיטקטורה

ארכיטקטורת Memory3 היא אבולוציה משמעותית של מודלים סטנדרטיים מבוססי טרנספורמרים, תוך שילוב **זיכרון מפורש** באופן חלק.

חידושים עיקריים:

1. **מנגנוני attention דלילים**:
   - באמצעות שילוב הזיכרון המפורש במנגנון attention, הגישה המוצעת נמנעת מעסקיילינג הריבועי של attention (היו בעבר טרנספורמרים שעשו משהו דומה). attention דליל מפחית כמות חישובים על ידי התמקדות רק בתת-קבוצות של זיכרון הרלוונטיות ביותר לשאילתה.

2. **אחזור זיכרון יעיל**:
   - המודל משתמש בחיפוש מבוסס דמיון קוסינוס כדי לאחזר זוגות מפתח-ערך(KV) רלוונטיים. אמבדינגס של חלקי הזיכרון הרלוונטיים מחושבים מראש שמבטיח אחזור מהיר וסקיילבילי, כך שמהירות האינפרנס לא נפגעת גם כשהזיכרון גדל.

3. **דילול(sparsification) זיכרון**:
   - כדי לשמור על יעילות הזיכרון, המחברים מציעים טכניקות כמו **בחירת טוקנים מדורגת (Top-k)**, שבהם נשמרים רק הטוקנים האינפורמטיביים ביותר. זאת בשילוב עם **קוונטיזציה של וקטורים**, שמכווצת את אמבדינגס של הזיכרון מבלי לאבד משמעותית מכוח הייצוג שלהן.

4. **גמישות בעדכוני ידע**:
   - בניגוד לאחסון מבוסס פרמטרים, זיכרון מפורש מאפשר עדכונים מודולריים. לדוגמה, הוספת ידע חדש כרוכה רק בהוספת זוגות KV במקום אימון מחדש של המודל, מה שהופך את Memory3 למותאם ומתאים לעתיד.

ד. פרדיגמת האימון

המחברים מאמצים פרדיגמת אימון בשני שלבים אשר מותאמת לשילוב זיכרון מפורש:

1. **שלב אימון warm-up**:
   - המודל עובר אימון בסיסי ללא EM. שלב זה מבטיח פיתוח של יכולות הפשטה חזקות והבנה לשונית בסיסית. שלב זה דומה לאימון מקדים במודלים טרנספורמריים מסורתיים.

2. **שלב אימון continual**:
   - המודל לומד לכתוב ולקרוא מ-EM. מטרות האימון מתרחבות כדי לכלול משימות ספציפיות לזיכרון כמו:

     - **כתיבת זיכרון**: אופטימיזציה של אחסון ידע בתור זוגות KV.
     - **אחזור זיכרון**: שיפור היכולת לאחזר מידע רלוונטי באופן יעיל ומדויק במהלך האינפרנס.

סיכום:

שילוב EM ב-Memory3 ממחיש דרך חדשנית לבניית מודלים לשוניים יעילים, ניתנים להתאמה ומודולריים. הגישה הזו עשויה (למרות שב-5 החודשים מאז יציאת המאמר לא ראיתי ניצנים לכך) להוות בסיס לדור הבא של LLMs, במיוחד בתחומים הדורשים עדכונים שוטפים של ידע ו-interpretability גבוה (בגלל שיש זיכרון מפורש).

⚡🚀המאמר היומי של מייק -07.12.24: ⚡🚀

Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering

1. תמצית המאמר

המאמר מציע שיטה המציידת RAG עם מערכת מבוססת **גרפי ידע (KG)** המותאמת לשירות לקוחות. המערכת, שפותחה על ידי צוות המחקר של LinkedIn, מעשירה LLMs בידע מבני שמקורו בפניות שירות היסטוריות. על ידי שילוב יחסים שונים בין פניות השירות (טיקטים) בגרף, השיטה משפרת באופן משמעותי את דיוק האחזור(retrieval), איכות התשובות והיעילות, עם שיפורים ניכרים במדדים כמו MRR, BLEU ומקטין זמני הטיפול בפניות.

2. תרומות מרכזיות

א. שילוב KG במערכות RAG

- **שימור מידע מבני**:
כל טיקט מיוצג כעץ (יחסים פנימיים בתוכו) ומקושרת לפניות אחרות דרך יחסים סמנטיים או מפורשים. עיצוב זה משמר את ההיגיון הלוגי של הטיקט, כולל תיאור הבעיה והפתרון. כל טיקט מהווה צומת בגרף.
- **שיפור באחזור ויצירת תשובות**:
המערכת מנווטת בגרף כדי לזהות תתי-גרפים רלוונטיים, המוזנים ל-LLMs לצורך יצירת תשובות איכותיות.

ב. בניית גרף הידע:

1. **עץ פנימי לטיקט**:
צמתים מייצגים חלקים כמו סיכומים או שורשי בעיה, וקשתות מציינות יחסים היררכיים.
2. **קשרים בין פניות**:
   - קשרים מפורשים: יחסים כמו ("e.g., "clone of" or "caused by").
   - קשרים סמויים: מחושבים על בסיס דמיון קוסיין בין אמבדינגס.

ג. שלבים בתהליך אחזור ותשובות

המערכת פועלת ב 3 שלבים:

- **זיהוי ישויות(entity) וכוונות**:
המערכת הופכת שאילתות משתמש לישויות וכוונות(intents) באמצעות LLMs וניתוח ותבניות YAML.
- **אחזור תת-גרפים**:
מתבצע חישוב דמיון בין אמבדינגס לשאילתה לצמתים בגרף לזיהוי תת-הגרפים הרלוונטיים ביותר.
- **יצירת תשובות**:
המערכת יוצרת תשובות בהתבסס על תת-הגרפים רלוונטיים לשאילתת המשתמש.

4. **קצת פרטים על השיטה**

השיטה המוצעת כוללת 3 שלבים עיקריים:

a. **זיהוי ישויות בשאילתה וזיהוי כוונה(intent):**

- המערכת מעבדת שאילתות משתמש על ידי חילוץ ישויות מוגדרות וכוונות באמצעות ניתוח תבניות YAML ו-LLMs. ישויות מוגדרות מייצגות אופיינים מהותיים (למשל, "תקציר בעיה" או "תיאור בעיה"), בעוד כוונות(intents) מכילות את מטרת השאילתה (למשל, "פתרון תיקון"). לדוגמה, בהינתן השאילתה

"כיצד לשחזר את בעיית ההתחברות כאשר משתמש לא יכול להתחבר ל-LinkedIn?", המערכת מזהה את הישויות כ"בעיית התחברות" ו"משתמש לא יכול להתחבר" ואת הכוונה כ"פתרון תיקון."

b. **אחזור מבוסס אמבדינגס (ייצוג):**

- **זיהוי פניות רלוונטיות:** מחשבים עד כמה הישויות שחולצו משאילתת המשתמש (למשל, "בעיית התחברות") תואמות את הצמתים ב-KG. עבור כל יישות בשאילתה, השיטה משתמשת בדמיון קוסיין למדידת קרבה בין ייצוג הישות לייצוגים של צמתים בגרף. הציונים מצטברים על פני כל הצמתים השייכים לטיקט מסוים. ככל שלטיקט יש מספר ישויות קרובות לשאילתה, הציון שלו עולה, מה שהופך אותו לסביר יותר להיבחר כרלוונטי.

- **חילוץ תת-גרף רלוונטי:** לאחר זיהוי טיקטים הרלוונטיים ביותר, הם משמשים לבניית שאילותות למסד נתונים (DB) בשפת שאילתות גרפים הנקראת Cypher. שאילתות אלה מאפשרות למערכת לחלץ תת-גרפים מקושרים, כגון תיאורים קשורים או שלבים לשחזור בעיה. תהליך האחזור המובנה הזה מבטיח(סוג של כמו תמיד) שהמערכת אוספת מידע מדויק ורלוונטי מבחינת ההקשר מגרף הידע.

c. **יצירת תשובה:**

- מגנרטת תשובות על ידי קישור נתוני הגרף שאוחזרו עם השאילתה המקורית. LLM מנסח מחדש את השאילתה באופן דינמי ומייצר תשובות מובנות. לדוגמה השאילתה "שגיאת העלאת csv בעדכון אימייל משתמש" מנוסחת מחדש ל-Cypher לאינטראקציה עם DB, מאחזרת פתרונות צעד-אחר-צעד.

6. סיכום

המאמר מציג דרך פורצת דרך לשילוב גרפי ידע במערכות RAG עבור מענה לשאלות בשירות לקוחות. על ידי לכידת יחסים פנימיים וחיצוניים בין פניות, המערכת משפרת משמעותית את דיוק האחזור ואיכות יצירת התשובות, ומציבה כיוון מעניין ביישומים פרקטיים של LLMs.

https://arxiv.org/abs/2404.17723:

המאמר היומי של מייק - 09.12.24:
Scaling Synthetic Data Creation with 1,000,000,000 Personas

תמצית המאמר ותרומות מרכזיות:

1. **השקת Persona Hub:**

- מאגר של מיליארד פרסונות מגוונות שנוצרו באמצעות טכניקות הניתנות להרחבה
- פרסונות אלו מגלמות ידע, תחומי עניין, התנסויות ומקצועות ייחודיים, המייצגים כ-13% מאוכלוסיית העולם

2. **יצירת דאטה סינתטי מבוסס פרסונות:**

- שילוב פרסונות בפרומפטים מאפשר למודלי שפה גדולים (LLMs) לייצר נתונים סינתטיים מגוונים במיוחד
- מדגים יישומים במגוון תחומים כגון בעיות מתמטיות, חשיבה לוגית, הוראות, טקסטים עתירי ידע, דמויות NPC במשחקים וממשקי כלים

3. **שיטות ליצירת פרסונות:**

טקסט-לפרסונה:
- מייצר פרסונות ישירות מנתוני רשת- מנתח הקשר טקסטואלי כדי להסיק את הפרסונה שסביר שקשורה אליו (למשל, "מי עשוי לכתוב או לחבב טקסט זה?")
- מפיק תיאורי פרסונה גסים או מדויקים (למשל, "מדען מחשב" לעומת "חוקר למידת מכונה המתמקד בארכיטקטורות נוירונים")
- מתרחב בקלות באמצעות LLMs ומאגרי נתונים ציבוריים ענקיים

פרסונה-לפרסונה:
- מרחיב פרסונות באמצעות קשרים יחסיים (למשל, ילד הקשור לאחות ילדים, או קבצן הקשור לעובד מקלט)
- משתמש בפרומפטים מבוססי יחסים כמו "מי נמצא בקשר קרוב עם פרסונה זו?"
- העשרת פרסונות נוספת על ידי איטרציה של שש דרגות הפרדה

4. **תהליך הסרת כפילויות פרסונות:**

- MinHash Deduplication: מסיר פרסונות דומות על בסיס חפיפת n-gram טקסטואלית

- Deduplication מבוסס אמבדינג: מסנן פרסונות באמצעות דמיון סמנטי (מרחק קוסיין) המחושב דרך אמבדינגים. ספי הדמיון הותאמו בהתאם לשיקולי איכות מול כמות
- לאחר ניקוי והסרת כפילויות, המאגר כלל 1,015,863,523 פרסונות ייחודיות

5. **יישומים:**

א. סינתזת בעיות מתמטיות:
- יצר 1.09 מיליון בעיות מתמטיות ייחודיות באמצעות פרסונות
- מודל 7B שעבר טיוב (fine-tuning) עדין עם בעיות אלו השיג דיוק של 79.4% על סט בדיקה סינתטי תוך-התפלגות ו-64.9% על MATH, תוצאה המשתווה ל-GPT-4-turbo-preview
- הדגים יכולת הרחבה - הוספת פרסונות שיפרה את גיוון הבעיות והבטיחה כיסוי רחב של מושגים מתמטיים

ב. בעיות חשיבה לוגית:
- סינתז חידות לוגיות מאתגרות (למשל, חשיבה מרחבית או זמנית) המותאמות למאפייני פרסונה
- כלל בעיות בסגנון Ruozhiba שובבי לבדיקת יכולות לוגיות מעודנות

ג. יצירת הוראות:
- יצר שאילתות משתמש המשקפות פרסונות מגוונות מהעולם האמיתי (למשל, כימאי עשוי לבקש מערכי ניסוי; אמן עשוי לבקש טכניקות ציור)
- אפשר סימולציות של שיחות רב-שלביות בין משתמש ל-LLM על ידי שרשור פרומפטים של פרסונות

ד. טקסטים עתירי ידע:
- יצר מאמרים ותוכן חינוכי המתואמים עם מומחיות הפרסונות (למשל, גנן כתב מדריכים על צמחים עמידים לבצורת)
- כיסה כמעט כל נושא באמצעות הרוחב של הפרסונות

ה. פיתוח כלים (פונקציות):
- חזה כלים שפרסונות עשויות להזדקק להם (למשל, נהג מונית הזקוק ל-API של תנאי תנועה)
- יצר הגדרות כלים עם קלטים, פלטים ותלויות ברורים

6. תוצאות מרכזיות:

- מודלים קטנים יותר (למשל, Qwen2 7B) שעברו כוונון עדין באמצעות נתונים סינתטיים השיגו רמות ביצועים שבדרך כלל דורשות מודלים גדולים יותר
- הוכיח שגיוון פרסונות מוביל לפלטים מגוונים ויצירתיים משמעותית יותר
- הדגים שפרסונות יכולות לדמות התנהגויות משתמש מגוונות, ולפעול ביעילות כנושאות מבוזרות של זיכרון ה-LLM

7. סיכום

המאמר מסמן קפיצת מדרגה (לא ברור עד כמה משמעותית) בגנרוט דאטה סינתטי. המתודולוגיה המוצעות נראית מבטיחה וניתנת ליישום עבור מגוון משימות, ויוצרת הזדמנויות לטיוב חכם של LLM, פיתוח יישומים, ואפילו סימולציות חברתיות.

https://arxiv.org/abs/2406.20094

המאמר היומי של מייק - 10.12.24:
LLM2LLM: Boosting LLMs with Novel Iterative Data Enhancement

1. מבוא ומוטיבציה

המאמר מציג את LLM2LLM, מסגרת חדשנית לשיפור ביצועי LLMs במצבים של מחסור בדאטה. בעוד שאימון נוסף של מודלים כאלה דורש בדרך כלל דאטה מתויג רב, מה שדורש עבודה ידנית מרובה, LLM2LLM מציע אסטרטגיית העשרת דאטה איטרטיבית המבוססת על פרדיגמת מורה-תלמיד(student-teacher) כדי לשפר את הדאטה בעייתיות (שהמודל הקטן, תלמיד, מתקשה להתמודד איתם) באופן דינמי

2. מתודולוגיה

LLM2LLM מורכב מ-3 שלבים עיקריים:

1. אימון מודל התלמיד: מודל התלמיד מאומן על כמות דאטה קטנה.
2. זיהוי שגיאות: הביצועים נמדדים על נתוני האימון באמצעות המודל הגדול (מורה), ודוגמאות שבהן המודל הקטן שוגה מזוהות.
3. העשרת דאטה ממוקדת: מודל המורה מייצר דוגמאות סינתטיות חדשות בתור אוגמנטציות שונות של הדוגמאות בהם מודל התלמיד טועה. דוגמאות אלו משתלבות מחדש במערכת לצורך איטרציות אימון נוספות.

מאפיינים מרכזיים:

- העשרה איטרטיבית: הדאטהסט לאימון מודל התלמיד משתפרות לאורך מספר סבבים במקום להיווצר מראש.
- מיקוד בטעויות: הדגש הוא על דוגמאות מאתגרות המדגישות את חולשות המודל הקטן.
- המחברים מציינים כי מודל המורה אינו חייב להיות חזק יותר, אלא רק להפיק דוגמאות קונספטואליות דומות לטעיות של המודל הקטן.

3. תוצאות

המסגרת הוכיחה שיפורים משמעותיים במדדים במצבי מחסור בדאטה תוך שהיא מתעלה על שיטות העשרה אחרות. דוגמאות לביצועים:

- GSM8K (הסקה מתמטית): שיפור של 24.2% בדיוק.
- CaseHOLD (הסקה משפטית): שיפור של 32.6%.
- SNIPS (זיהוי כוונות): שיפור של 32.0%.
- TREC (סיווג שאלות): שיפור של 52.6%.
- SST-2 (ניתוח רגשות): שיפור של 39.8%.

4.סיכום

LLM2LLM מציעה מסגרת להעשרת הדאטה באימון LLMs במצבים של מחסור בדאטה. על ידי התמקדות איטרטיבית בדוגמאות מאתגרות ושימוש בשיתוף פעולה בין מורה לתלמיד, היא משיגה שיפורי ביצועים משמעותיים. שיטה זו מסמנת כיוון מבטיח לשיפור היעילות והשימושיות של מודלים לשוניים בסביבות מוגבלות משאבים.

https://arxiv.org/pdf/2403.15042

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

המאמר היומי של מייק - 19.12.24:
Large Concept Models: Language Modeling in a Sentence Representation Space

מאמר שני (גם הוצג ב-NeurIPS 2024) של מטה המציע קונספט די מהפכני למודלי שפה. במאמר שסקרתי אתמול הם הציע לוותר על הטוקנייזר הסטנדרטי במודלי שפה ובמאמר שנסקור היום הם הציע לוותר על חיזוי של טוקן הבא שהתרגלנו אליו כל כך ב-LLMs.

כמו שאתם בטח זוכרים LLMs מאומנים (באימון מקדים וב-SFT) באמצעות מקסום הנראות (likelihood) של דאטהסט אימון D, כלומר מקסום של הסתברות גנרוט של D עם המודל המאומן. כדי לעשות את זה אנו ממקסמים (ביחס לפרמטרי מודל השפה שלנו) הסתברות של כל הפיסת דאטה. מכיוון שכל פיסת דאטה מורכב מטוקנים ניתן לבטא אותה באמצעות חוק בייס כמכפלה של הסתברויות מותנות שכל טוקנים בהינתן הטוקנים הקודמים (כלומר הקונטקסט). וככה אני מגיעים לחיזוי של טוקן בהינתן הקונטקסט גם אימון וגם כמובן באינפרנס.

המאמר מציין כי "חשיבה טוקן טוקן" אלא בקונספטים כאשר אנו בונים את הדיבור שלנו (תוך כדי הדיבור). המאמר מציע להטיל את הגישה הזו למודל שפה כאשר קונספט מוגדר בתור משפט. כלומר המחברים מציעים לאמן מודל לחזות את המשפט הבא במקום חיזוי טוקן הבא שאנו רגילים אליו במודלי שפה סטנדרטיים.

אבל איך נחזה משפט, הרי זה משהו דיסקרטי ועבור אורך די צנוע של המשפט מספר הערכים האפשריים שהוא יכול להיות הינו מעריכי והופך להיות גבוה מדי כדי לבצע את החיזוי בו (כלומר סופטמקס בגודל עצום). אז המאמר מעביר אותנו למישור הרציף ומציע לאמן מודל, שקיבל שם Large Concept Model או LCM לחיזוי אמבדינג של המשפט בהינתן האמבדינגס של המשפטים הקודמים לא בחלון הקונטקסט. המאמר בוחן כמה פונקציות לוס שהפשוטה מהם היא L2 בין האמבדינג ה-ground-truth לבין החזוי (יש עוד כמה מעניינים בפרק 2.4.1 במאמר).

הדרך הנוספת שהמאמר הציע לבנות את האמבדינג של המשפט הבא הוא אימון מודל דיפוזיה מותנה (רעיון יפה מאוד לטעמי) לחיזוי האמבדינג שלו.

האמבדינג נבנה על ידי מודל embedder שהוא נשאר קבוע במהלך האימון. בנוסף ל-embedder (שהוא encoder) יש לנו גם דקודר שהופך את הקונספט (האמבדינג שלו) לטקסט.

מאמר די יפה, כתוב די ברור רק קצת ארוך מדי לדעתי...

https://arxiv.org/abs/2412.08821

המאמר היומי של מייק - 20.12.24:
FAN: Fourier Analysis Networks

היום סוקרים קצרות מאמר המציע שכבה ארכיטקטונית חדשה לרשתות נוירונים. שכבה זו משלבת פונקציות מחזוריות כמו סינוס וקוסינוס. פונקציות מחזוריות אינן חיה חדשה בטריטוריה של הרשתות; כבר ראינו אותם במאמרי Neural radiance fields או NERF שהן משמשים לבניית מודלי 3D של אובייקטים וסצנות. למיטב זכרוני היה מאמר שבנה ייצוג של תמונה באמצעות רשת המערבת אקטיבציות מחזוריות.

אולם המאמר של היום מציע לבנות שכבה המכילה פונקציות מחזוריות אלא מציע לשלב אותן עם פונקציות אקטיבציות קלאסיות יותר כמו סיגמויד כאשר השילוב הוא לינארי. אז השכבה בגדול בנויה מצירוף לינארי של סינוסים וקוסינוסים עם מקדמים נלמדים יחד עם פונקציות אקטיבציות סטנדרטיות. השכבה הזו טובה למידול פונקציות מחזוריות כאשר ביצועיה על פונקציות לא מחזוריות אינן ברורות (המאמר טוען שיש שיפור גם שם),

המאמר גם מציע להחליף ב-FAN את שכבות ה-FFN בטרנספורמרים וגם שכבות gating ב-LSTM (אותו סכום ממשוקל את סינוסים וקוסינוס יחד עם הסיגמואיד) ומדווח שיפור בביצועים בכמה משימות.

רעיון מעניין...

https://arxiv.org/abs/2410.02675

המאמר היומי של מייק - 22.12.24:
Reasoning in Large Language Models: A Geometric Perspective

מאמר זה חוקר את יכולות החשיבה של LLMs מנקודת מבט גיאומטרית, תוך התמקדות בקשר בין הממד הפנימי(intrinsic dimension או ID) של ייצוגי הקלט לבין עוצמת expressiveness של מודלים אלה. החוקרים בוחנים כיצד ארכיטקטורות טרנספורמר מחלקות את מרחבי הקלט וכיצד חלוקה זו קשורה ליכולות ההנמקה שלהן (reasoning). העבודה מציעה תובנות חשובות לגבי האופן שבו ארכיטקטורת המודל ואורך ההקשר משפיעים על ביצועי LLM במשימות הנמקה.

רעיונות מרכזיים:

מסגרת גיאומטרית לכוח ביטוי של מדוך (אקספרסיביות)
הרעיון המרכזי סובב סביב צפיפות גרפי מנגנון self-attention והשפעתה על הממד הפנימי של הקלטים לשכבות MLP בתוך הטרנספורמרים (כלומר FFN). המימד הפנימי, בהקשר זה, מודד את מספר דרגות החופש האפקטיביות הנדרשות לייצוג אמבדינג של הקלט.

מנגנון self-attention כגרף:
הפלט של שכבת מנגנון self-attention מתואר כגרף, בו טוקנים הם צמתים ומקדמי attention מגדירים קשתות משוקללות. צפיפות הגרף קובעת את מספר החיבורים האפקטיביים, המשפיעים ישירות על הממד הפנימי של הייצוגים המועברים לבלוקי MLP.

חלוקת מרחב הקלט:
ממדים פנימיים גבוהים יותר מאפשרים לשכבות ה-MLP לחלק את מרחב הקלט לאזורים עדינים יותר. זה מאפשר למודל לבנות מיפויים מורכבים יותר ולתפוס קשרים לא-לינאריים ביעילות. כתוצאה מכך, יכולת ההנמקה של ה-LLM משתפרת עם כוח הביטוי המוגבר הנובע מחלוקות עדינות אלה.

יכולות קירוב:
על ידי אפשור חלוקה עדינה יותר, ממדים פנימיים גבוהים יותר מפחיתים שגיאות קירוב, מאפשרים ל-MLP לייצג פונקציות מורכבות בדיוק רב יותר. זה מתקשר ישירות למשימות הנמקה, בהן מיפויים מדויקים ותלויי הקשר הם קריטיים.

הסברים מעמיקים על הרעיונות:

חלוקה וקירוב

החוקרים משתמשים בניסוח piece-wise affine של רשתות נוירונים עמוקות (DNNs) כדי להסביר כיצד מרחב הקלט מחולק. הרעיון המרכזי של חלק "החלוקה והקירוב" הוא לתאר כיצד DNNs מחלקות את מרחב הקלט למספר אזורים, כל אחד נשלט על ידי כלל ליניארי ספציפי משלו.

חלוקת מרחב הקלט:

רשתות נוירונים (באמצעות פונקציות אקטיבציה בשכבותיה), מחלקות את מרחב הקלט למספר אזורים מובחנים. אזורים אלה מוגדרים על בסיס האופן שבו הנוירונים מופעלים בתגובה לנתוני הקלט. חשבו על מרחב הקלט כמפה, והרשת יוצרת "אזורים" על מפה זו כאשר לכל אזור יש כלל ייחודי משלו.

קירוב לינארי בתוך כל אזור:

בתוך כל אזור כזה, הרשת מתנהגת כמו פונקציה לינארית. זה למעשה מאפשר קירוב פונקציות מורכבות יותר על ידי שילוב חלקים פשוטים אלה.

יכולת הרשת לקרב פונקציות מורכבות תלויה ביכולתה לחלק את מרחב הקלט ו"להגדיר" חוקים לכל אזור. יותר חלוקות מאפשרות קירוב טוב יותר, שהוא קריטי למשימות מורכבות כמו הנמקה. מסגרת זו עוזרת להבין כיצד רשתות משתמשות באבני בניין פשוטות (מודלים לינאריים באזורים ספציפיים) כדי להתמודד עם בעיות מורכבות מאוד. ניסוח זה מדגיש את היכולת של DNNs לחלק באופן אדפטיבי את מרחב הקלט על בסיס דאטה האימון, כאשר מספר האזורים מתואם ישירות עם כוח הקירוב של המודל.

עבור טרנספורמרים, תורה זה ניתנת ליישום למנגנון multi head self attention או MHST שבו צפיפות האינטראקציות בין טוקנים משפיעה על החלוקה המושרית של מרחב הקלט ברמת שכבות ה-MLP שלו.

משפט מרכזי:
סכום מינקובסקי מסביר כיצד הפלטים של שכבת MHST מובנים גיאומטרית וקשורים למושג המימד הפנימי. בטרנספורמרים, MHST מפצלת את מנגנון attention למספר "ראשים", כאשר כל ראש מתמקד בהיבט ספציפי

של הקלט. ראשים אלה עובדים במקביל כדי לתפוס יחסים שונים בתוך הדאטה. המשפט מראה ניתן לפרש את הפלט של MHST כשילוב של אזורים שנוצרו על ידי כל ראש בודד. כל ראש מגדיר "צורה" (טכנית, מעטפת קמורה) המבוססת על הטרנספורמציות שהוא מחיל על הקלט.

סכום מינקובסקי:
סכום מינקובסקי הוא פעולה מתמטית המשמשת לשילוב צורות אלה. באופן אינטואיטיבי, זה אומר שהפלט של שכבת MHST הוא מרחב הכולל את כל השילובים האפשריים של פלטי הראשים הבודדים.

קשר למימד פנימי:
תוצאה זו מדגישה שהוספת ראשים נוספים או הפיכת הראשים לאקספרסיביים יותר מגדילה את ה"ממדיות" של המרחב שבו נמצאים פלטי תשומת-הלב. ממדיות מורחבת זו משפרת את יכולת המודל לייצג יחסים מורכבים ותהליכי חשיבה. המשפט מפרמל כיצד מנגנון MHST מחלק ומשלב את ההיבטים הגיאומטריים של מרחב קלט כדי להגביר את האקספרסיביות ויכולת הנמקה של מודלי טרנספורמר.

המימד האפקטיבי של סכום מינקובסקי תלוי בצפיפות גרף attention (כלומר, מספר החיבורים הפעילים בין טוקנים). צפיפות גרף גבוהה יותר, המושגת באמצעות יותר ראשי attention או קישוריות גבוהה יותר, מובילה לממדיות פנימית גדולה יותר של הקלט לשכבות MLP. מימד פנימי בוחן עד כמה טוב טרנספורמר יכול לתפוס יחסים מורכבים בקלט שלו בהתבסס על מספר החיבורים המשמעותיים שהוא מזהה.

מימד פנימי גבוה יותר פירושו שיותר חלקים מהקלט משפיעים על טוקן. זה מוביל לייצוגים עשירים ומפורטים יותר של הקלט, המאפשרים למודל להבין טוב יותר דפוסים ויחסים מורכבים. כאשר למודל יש ממד פנימי גבוה, הוא יכול "לחלק" ביעילות את מרחב הקלט ליותר אזורים, מה שמאפשר לו לתפוס פרטים ודקויות עדינים יותר. זה קריטי למשימות חשיבה, שבהן הבנת יחסים עדינים היא מפתח.

השלכות מעשיות:
הגדלת מספר ראשי attention או קלטים ארוכים יותר עשויים להגדיל את המימד הפנימי. זה משפר את יכולות החשיבה של המודל מבלי לדרוש שינויים בארכיטקטורה שלו או בתהליך האימון. הממד הפנימי משקף עד כמה עמוק טרנספורמר מתעסק עם הקלט שלו. ככל שהחיבורים עשירים יותר, כך המודל יכול לחשוב טוב יותר ולבצע משימות מורכבות.

https://arxiv.org/abs/2407.02678

המאמר היומי של מייק - 23.12.24:
T-FREE: Tokenizer-Free Generative LLMs via Sparse Representations for Memory-Efficient Embeddings

שוב חוזרים לנושא הטוקנייזרים - מתברר שהוא יותר חם ממה שחשבתי. נתקלתי במאמר המעניין שיטה נוספת לטוקניזציה המבוססת על פונקציה האש n-grams. השיטה המוצעת באה להתמודד עם גודל העצום של המילון מלווה כל מודל שפה גדול (עשרות אלפי טוקנים לכל הפחות) וגם טוקנים דומים מאוד מבחינת האותיות האותיות שמצריכות אמבדינגים שונים שזה לא יעיל (לטענת המחברים).

המחברים מנסים שיטת טוקניזציה שה-encoding שלה המורכב משלבים הבאים:

- פירוק של טקסט למה שהם קוראים טוקנים כאשר ב-T-FREE טוקנים אלו הם בעצם מילים

- כל מילה מחולקת לסדרה של grams-3 לא זרים למשל מילה hello מיוצגת על ידי חמישה grams-3 הבאים: {_He, Hel, ell, llo, lo_}. מספר grams-3 בייצוג הזה בדרך כלל מספר n-grams במילה שווה למספר האותיות במילה
- מקודדים כל gram-3 עם m פונקציות האש שכל אחת מהם מקבלת v ערכים אפשריים כאשר v הינו אחד הייפר-הפרמטרים של השיטה.
- כך כל מילה מקודדת על ידי n*m מספרים בין 0 ל-v כאשר n הינו אורך המילה (מספר אותיות). ייצוג המילה הוא ממוצע (ועיגול) של כל nm ערכים האלו.
- כל ערך בין 0 ל-v מקודד על ידי וקטור נלמד כאשר v וקטורים אלו למעשה מהווים את המילון של השיטה

שלב האימון והפענוח (כלומר גנרוט של מילים) נראים קצת יותר מורכבים. קודם כל באימון המטרה היא לחזות את nm האשים של grams-3 של המילה הבאה. כלומר במקום בעיית multi-class בפענוח של הטוקניזציה הרגילה (חיזוי של טוקן ממילון הטוקנים) יש לנו כאן בעיית multi-label כאשר אנו חוזים n*m האשים. שימו לב ש n תלוי באורך המילה כלומר יש לנו מספר "לייבלים" שונה לפי אורך המילה.

הפענוח לא ממש ברור לי האמת. כאשר אנו רוצים לחזות את המילה הבאה אנו קודם כל מחשבים את כל ההאשים עבור כל המילים האפשריות (זה די הרבה כי לכל מילה יש גם את כל ההטיות שלה לכל הפחות ובנוסף מילים בעלות אורכים שונים מקודדים עם מספר n*m שונה של האשים). לאחר מכן בוחרים את המילה המיוצגת על יד האשים בעלי "ההסתברות הגבוהה ביותר". נזכור שהמודל חוזה הסתברות של כל ערך של האש מ 1 עד v (גודל המילון) ולא לגמרי ברור איך נבחרת קבוצת האשים בעלת הסתברות הגבוהה ביותר.

בקיצור מאמר נחמד אבל לא ברור לי העניין עם הפענוח...

המאמר היומי של מייק - 25.12.24:
Vision language models are blind

מאמר נחמד הטוען שמודלי שפה ויזואליים הם די עיוורים כלומר אין להם סיכוי לעבור בדיקה אצל אופטומטריסט מורשה. הנה כמה עובדות על המבחנים הכושלים שלהם:

1. מודלי שפה ויזואליים או VLMs לא יכולים לקבוע באופן אמין האם שני קווים (או שני מעגלים) נחתכים, במיוחד כשהם קרובים זה לזה. הדיוק בזיהוי 0, 1 או 2 נקודות חיתוך בין שתי פונקציות לינאריות למקוטעין בעלות 2 מקטעים נע בין 47% ל-85%. באותה משימת שני המעגלים, המודלים מתפקדים טוב יותר (דיוק של 73-93%) אך עדיין רחוק מה-100% המצופה.

2. מודלי שפה ויזואליים יכולים לזהות בצורה מושלמת מעגל ומילה בנפרד אך כאשר המעגל המילה נמצאת בתוך המעגל המודלים נוטים להתקשות בזיהוי איזו אות מוקפת במעגל.

3. מודלי ראייה-שפה יכולים לספור צורות במדויק, למשל, מעגלים , ריבועיים כאשר הם נפרדים ורחוקים זה מזה. עם זאת, כל המודלים מתקשים לספור מעגלים חותכים (כמו הלוגו האולימפי), ובאופן כללי, צורות בסיסיות שהן חופפות או מקוננות.

4. בסידור ריבועים בצורה של רשת, אנו מגלים ש-VLMs נכשלים באופן מפתיע בספירת מספר השורות או העמודות ברשת, בין אם היא ריקה או מכילה טקסט. זה מפתיע בהתחשב בכך שהמודלים מתפקדים כל כך טוב (דיוק ≥ 90%) על הדאטהסט ב-DocVQA הכולל שאלות רבות עם טבלאות(אוברפיט כנראה).

5. כאשר המודל מתבקש לעקוב אחר מסלולים צבעוניים במפת רכבת תחתית של עד 8 מסלולים וסך הכל 4 תחנות, VLMs לעתים קרובות נכשלים בזיהוי היכן מסלול מסתיים, כלומר, ומפגינים דיוק של 23% עד 50% .

6. המודל GPT-4o עולה בביצועיו על Gemini-1.5 Pro ו-Claude-3 Sonnet ב-7 בנצ'מרקים מורכבים עבור VLMs אך מתפקד באופן משמעותי פחות טוב במשימות הנבחנות במאמר, שבהן Gemini-1.5 Pro ו-Sonnet-3.5 הם הטובים ביותר. כלומר, המאמר מגלה מגבלות מפתיעות של מודלי ראייה-שפה שלא נמדדו בבנצ'מרקים רגילים.

בקיצור אולי VLMs האלו צריכים משקפיים...

https://arxiv.org/abs/2407.06581

המאמר היומי של מייק - 26.12.24:
RL for Consistency Models: Faster Reward Guided Text-to-Image Generation

מזמן לא סקרתי מאמרים על מודלי דיפוזיה אז אחרי שנתקלתי במאמר הנחמד המשלב מודלי דיפוזיה גנרטיביים עם למידה עם חיזוקים (Reinforcement Learning או RL בקצרה), לא היו לי ספקות שזה הולך להיות המאמר היומי. כאמור המאמר פיתח שיטת אימון מודל של דיפוזיה גנרטיבי מסוג Consistency Model או CM.

קודם כל נשאלת השאלה למה צריך לאמן מודלי דיפוזיה גנרטיביים עם שיטות הלקוחות מעולם RL. הרי יש לנו שיטות סטנדרטיות יותר לאימון של מודלי דיפוזיה שהצליחו להביא לנו מודלים בעלי ביצועים מרשימים (בגנרוט תמונות מטקסט). אתם בטח יודעים שאימון מודלי דיפוזיה לגנרוט תמונות זה דבר לא זול ודורש לא מעט זמן ושימוש RL לאימון (או fine-tune) של מודלי דיפוזיה יכול לחסוך לנו זמן במקרים שאנו צריכים לאמן מודל דיפוזיה ייעודי (למשל לדומיין נישתי)

אחת הדוגמאות למשימה כזו היא אימון מודל ליצירת תמונות מפרומפט (תיאור טקסטואלי) כאשר יש בידינו פונקציה המשערכת את התאמת התמונה לפרומפט. אתם כבר יכולים לנחש שפונקציה זו תשרת לנו בתור פונקצית תגמול (reward function).

כבר הזכרתי שהמאמר משלב שיטה חדשה (יחסית) לאימון מודלי דיפוזיה הנקראת CM ושיטה זו (שהומצאה על ידי איליה סלוצקב ושות') מאפשרת גנרוט יותר מהיר של מודלי דיפוזיה גנרטיביים. בגדול מאוד שיטה זו מנסה לאמן מודל שאוכף עקביות בין התמונות המשוחזרות על ידי המודל מתמונות מורעשות עם עוצמות שונות רעש. כלומר לוקחים תמונה, מרעישים אותה עם רעש (בד"כ גאוסי) עם שונויות שונות ומאמנים מודל להחזיר את אותה התמונה הנקייה (עקביות לשמה).

למה השיטה הזו מאפשרת גנרוט יותר מהיר של תמונות? כי בגדול היא מאפשרת לגנרט תמונה נקייה מרעש באיטרציה אחת בלבד (ככה המודל מאומן). במציאות עושים את זה בכמה איטרציות (מספר קטן). מתחילים מרעש, מגנרטים את התמונה ממנו, מוסיפים פחות רעש לתמונה המגונרטת, מגנרטים מהתמונה המורעשת שוב וממשיכים ככה כמה איטרציות (עשרות בודדת). זה מאפשר לזרז את תהליך הגנרוט כי מודלי דיפוזיה סטנדרטיים צריכים מאות איטרציות בד"כ.

אוקיי, אחרי הקדמה ארוכה נעבור לתיאור של מה שעשו במאמר. המחברים הגדירו Markov Decision Process c או MDP המתאר תהליך גנרוט של תמונה (או כל דאטה אחר למעשה). כאמור פונקציה תגמול ניתנת לנו והיא מודדת מידת התאמה של התמונה המגונרטת לפרומפט. המאמר מגדיר:

- המצב s_t בתור שלישיה התמונה מגונרטת באיטרציה t, עוצמת הרעש והפרומפט c
- הפעולה a_t היא התמונה באיטרציה t + 1
- הפוליסי היא זו פונקצית התפלגות מותנית של תמונה מאיטרציה t+1 בהינתן התמונה המגונרטת מאיטרציה t בתוספת רעש
- המצב המתחלתי הוא רעש גאוסי סטנדרטי ופונקציית תגמול נתונה לנו

אחרי שהגדרנו את ה-MDP של תהליך גנרוט התמונה אנו יכולים להשתמש בשיטה DPO או Direct Preference Optimization לאימון פונקצית עקביות (= המודל שאנו מאמנים). למעשה DPO מאמן מודל הממקסם את פונקצית התגמול תוך כדי הגבלת של גודל עדכון פרמטרי המודל בכל איטרציה (הומצא על ג'ו שולמן ה-CTO של OpenAI לשעבר).

המאמר גם טוען שאימון כזה הוא חסכוני מבחינת משאבי החישוב הנדרשים ויעיל מבחינה הדאטה (כלומר יכול לעבוד לדאטהסטים קטנים).

https://arxiv.org/abs/2404.03673

המאמר היומי של מייק - 27.12.24:
Position: Future Directions in the Theory of Graph Machine Learning

דו"ח זה(כן כן, זה דוח למרות שהוא פורסם בארקיב) טוען כי בעוד שרשתות נוירונים גרפיות (GNNs) זכו להצלחה משמעותית במספר משימות, ההבנה התיאורטית שלנו לגביהן נשארת חלקית ומנותקת במידת מה מיישומים מעשיים. החוקרים מזהים שלושה תחומים מרכזיים הדורשים חקירה תיאורטית מעמיקה יותר:

1. יכולת ביטוי(expressiveness) - אילו דפוסים, פונקציות ומבנים יכולות GNNs לייצג בפועל?
2. הכללה(generalization) - עד כמה טוב GNNs מיישמות את הלמידה שלהן על גרפים חדשים שלא ראו?
3. אופטימיזציה - כיצד דינמיקת האימון משפיעה על ביצועי GNN?

נקודות מפתח בנושא כושר ביטוי של GNNs המוזכרות במאמר:

מגבלות נוכחיות:

- רוב העבודה התיאורטית מתמקדת בשאלות בינאריות (האם GNN יכולה להבחין בין שני גרפים?) במקום במדדים כמותיים (עד כמה שונים שני גרפים?)
- הניתוחים מוגבלים לרוב לארכיטקטורות GNN טיפוסיות ואינם מתחשבים בווריאציות של GNN במשימות מהעולם האמיתי
- התוצאות אינן מתחשבות במאפייני צמתים/קשתות רציפים הנפוצים ביישומים אמיתיים

כיוונים מוצעים:

- פיתוח מדדים למדידת דמיון בין גרפים המתואמים עם האופן שבו GNNs מעבדות אותם
- חקירת השפעת הבחירות הארכיטקטוניות (כמו פונקציות אקטיבציה ונורמליזציה) על כושר הביטוי
- יצירת תוצאות אחידות שעובדות על גרפים בגדלים שונים
- התמקדות בסוגי גרפים רלוונטיים מעשית (כמו גרפים מולקולריים)

תובנות לגבי יכולות הכללה של GNNs:

המצב הנוכחי:

- החסמים התיאורטיים הקיימים לרוב מורכבים (לבדיקה) או קשיחים מדי מכדי להיות מעשיים
- הניתוח בדרך כלל מתעלם ממבנה הגרף ותהליך האופטימיזציה
- התוצאות אינן מסבירות מדוע GNNs מורכבות יותר לעתים מכלילות טוב יותר

מחקר נדרש:

- הבנת השפעת מבנה הגרף על הכללה
- ניתוח ביצועים על דאטה out-of-distribution (במיוחד על גרפים גדולים יותר)
- פיתוח טכניקות העשרת דאטה (אוגמנטציה) טובות יותר עבור גרפים
- חקירת השפעת הבחירות הארכיטקטוניות על יכולת הכללה של GNN

אתגרי אופטימיזציה של GNNs:

סוגיות מרכזיות:

- הבנה מוגבלת של אופן שבו מורד הגרדיאנט(gradient descent) עובד עבור GNNs
- לא ברור מדוע בחירות ארכיטקטוניות מסוימות (כמו נורמליזציה) עוזרות או פוגעות בתהליך אופטימיזציה של GNN
- לעתים GNN עם פרמטרים אקראיים עובדים טוב מ-GNN מאומן

כיווני מחקר:

- חקירת תכונות התכנסות עם פונקציות אקטיבציה תואמות יותר לבעיות ספציפיות (כמו למידה מבנה של מולקולות)
- הבנת השפעת מבנה הגרף על אופטימיזציה
- מחקר מתמטי מעמיק המנסה להסביר מדוע GNNs עמוקות יותר קשות לאימון (יש כמה מאמרים המדברים על over-smoothing בהקשר הזה אבל אנו עדיין רחוקים מהבנה מלאה של מה שקורה שם)
- ניתוח תפקיד טכניקות הנורמליזציה

השלכות מעשיות

החוקרים מדגישים שהתקדמויות תיאורטיות צריכות להתחבר לצרכים מעשיים:

- פיתוח נקודות ייחוס סטנדרטיות ופרוטוקולי הערכה של GNNs
- יצירת מימושים יעילים של ארכיטקטורות מבוססות תאוריה
- אינטגרציה עם טכנולוגיות AI מתפתחות כמו מודלי שפה גדולים

חשיבות המאמר:

1. מזהה פערים קריטיים בין תיאוריה ופרקטיקה במחקר GNN
2. מספק מפת דרכים למחקר תיאורטי עתידי שעשוי לשפר יישומים מעשיים
3. מדגיש את הצורך לשקול את כל שלושת ההיבטים (כושר ביטוי, הכללה, אופטימיזציה) יחד
4. "קורא" בהנגשת התקדמויות תיאורטיות למיישמים בפועל

עבור קוראים עם ידע בסיסי ב-GNN, מאמר זה מדגיש מדוע הבנה תיאורטית חשובה וכיצד תיאוריה טובה יותר יכולה להוביל ליישומים מעשיים יעילים יותר. בעוד שחלק מהפרטים הטכניים עשויים להיות מורכבים, המסר המרכזי לגבי הצורך במסגרות תיאורטיות ומעשיות ומקיפות יותר הוא ברור וחשוב.

https://arxiv.org/abs/2402.02287

המאמר היומי של מייק - 30.12.24:
Graph Diffusion Policy Optimization

לפני יומיים סקרתי מאמר על מודלי דיפוזיה המאומנים באמצעות שיטות מעולם למידה עם חיזוקים או RL, אתמול סקרתי מאמר על רשתות נוירונים על גרפים והיום החלטתי לסקור מאמר שמאחד את 3 הדברים האלו (כמעט). המאמר המסוקר היום מציע שיטה לאימון מודל המגנרט גרפים באמצעות מודלי דיפוזיה המאומנים עם שיטות RL (נכון אין כאן GNN בצורתם הטהורה אבל לפחות יש גרפים...

קודם כל אנו צריכים להבין איך ניתן למנף מודלי דיפוזיה לגנרוט גרפים. האמת זה די פשוט ודומה לגנרוט תמונות. אתם זוכרים מודלי דיפוזיה מאומנים לגנרט תמונה מרעש טהור (בד״כ) על ידי הורדה הדרגתית של הקומפוננטה הרועשת שלו עד להפיכתו לפיסת דאטה המפלגות לפי ההתפלגות של דאטהסט אימון. זה ממש בגדול ויש גישות חדשות יותר שעושות את זה טיפה אחרת למשל כמו Consistency Models שדיברנו עליהם באחת הסקירות הקודמות.

האם אנחנו יכולים לעשות משהו דומה עם גרפים? מתברר שכן. אנו יכולים להתחיל מלדגום גרף באקראי (כלומר הצמתים והקשתות שלו) ולאמן מודל לשנות את הערכים בצמתים ובקשתות כך שהגרף יהפוך להיות ״דומה״ לאחד הגרפים מדאטהסט האימון וגם יקבל ערך גבוה לפי איזה פונקציית תגמול(המאמר גם על RL, זוכרים). ד״א, יש כאן הנחה סמויה שצומת יכול לקבל מספר סופי של ערכים (נגיד מ 0 עד a) וכל קשת יכולה להיות מכמה סוגים (כלומר מ- 0 עד b). כלומר ההתפלגויות שאנו דוגמים מהם הם קטגוריאליות וזה שונה ממה שאנו רגילים לראות במודלי דיפוזיה גנרטיביים עבור התמונות.

כמובן מיד עולות כמה שאלות בנוגע לתהליך הזה?

- איך דוגמים גרף באקראי במהלך האינפרנס (זה נושא עתיק ונחקר רבות עלי ידי מתמטיקאים ובפרט על ידי ארדוש, המאמר לא מתעמק בזה יותר מדי). דרך אגב במהלך האימון אנו לוקחים גרף מהדאטהסט ומרעישים אותו עלי ידי ״שינוים אקראיים״ בערכי הצמתים ובסוגי הקשתות
- איך משווים גרפים, כלומר איך מבינים שגרף שקיבלנו במהלך הגנרוט הוא דומה לגרף מהדאטהסט? יש מספר רב גישות להשוות גרפים על ידי השוואה של התת-גרפים שלהם או להשוות את הלפלסיאן שלהם למשל.
- בחירה של פונקצית reward בדומיין הגרפים לא טריוויאלית בכלל. למשל למשימות גנרוט גרפים למולקולות חדשות אחד המדדים לאיכות הגרף המגונרט הוא חדשנותו יחסית לדברים הקיימים, יעילותו בטיפול במחלה מסוימות או פיזיביליות של סינטוזו (synthetic accessibility). ניתן לבחור reward גם בתור פונקצית דמיון לגרפים הקיימים.

אוקיי, אז יש לנו פונקציה להשוואת הגרפים C ופונקצית תגמול לשערוך איכות הגרף r - איך אנו מאמנים מודל דיפוזיה. האמת בצורה די דומה לזו שתיארתי בסקירת של לפני 3 ימים של המאמר: RL for Consistency Models: Faster Reward Guided Text-to-Image Generation.

קודם כל אנו צריכים להגדיר את Markov Decision Process עבור אימון מודל דיפוזיה על גרפים. ומתברר שהוא ממש דומה למאמר שהזכרתי:

- המצב s_t בתור זוג של גרף מגונרטת באיטרציה T-t וגם ערך T-t
הפעולה a_t היא הגרף באיטרציה T−t−1
- הפוליסי (הסתברות של a_t בהינתן s_t) היא זו פונקצית התפלגות מותנית של גרף מאיטרציה T−t−1 בהינתן גרף באיטרציה T-t
- המצב ההתחלתי הוא גרף אקראי באיטרציה T ופונקציית תגמול r שנתונה לנו המחושבת על הגרף הסופי באיטרציה 0

המאמר מציע שתי שיטות לאימון של מודל דיפוזיה לגנרוט גרפים: הראשונה היא REINFORCE הקלאסי שהיא למעשה שיטת policy gradient הממקסת פוליסים בעלי תגמול גבוה. מעשה אנו דוגמים K איטרציה בין 1 ל T וממקסמים מכפלה ממוצעת (על K דגימות) של פונקציית הפוליסו (ונקצית התפלגות מותנית של גרף מאיטרציה T−t−1 בהינתן גרף באיטרציה T-t) והתגמול עבור הגרף המגונרט (באיטרציה 0).

השיטה השנייה המוצעת היא Policy Optimization כאשר במקום למקסם את הפוליסי בצורתו הטהורה אנו ממקסמים הסתברות גנרוט גרף G_0 מהדאטהסט (שאותו מרעישים והמודל "מסיר" ממנו את הרעש) מוכפלת בתגמול עבור הגרף הנוצר. גם כאן יש מיצוע על K איטרציות שמהם נבנה שערוך של G_0.

זהו זה - סקירה קצת כבדה, מקווה שהצלחתם להבין משהו ממנה...

https://arxiv.org/abs/2402.16302

המאמר היומי של מייק - 01.01.25:
Inference-Aware Fine-Tuning for Best-of-N Sampling in Large Language Models

מתחילים את השנה החדשה עם סקירה של מאמר די מעניין שמציע שיטה לשיפור אימון של מודלי שפה. היתרון הגדול של השיטה היא מאפשרת להתאים את האימון לאופן ההיסק (אינפרנס) ודי ברור שאם אכן עושים זאת בהצלחה זה אמור להניב איכות ההיסק. כלומר אם אנו משתמשים בגישה מסוימת במהלך האינפרנס: למשל לבחור את "התשובה הטובה ביותר" מבין N תשובות המודל (המאמר מפתח שיטות רק לגישה זו וקורא לה BoN) או תיקון עצמי (self-correction) אז כדאי לנו להתאים את האינפרנס לכך.

קודם כל המאמר מנסח שתי פונקצית יעד לאימון inference-aware או IA בקצרה, אחת ל SFT וגם ל-RLHF, שקיבלו שמות IA-SFT ו-IA-RL בהתאמה. עבור IA-SFT אנו ממקסמים את נראות של תשובות המומחים לשאלות מהדאטהסט שיש לנו בהינתן פוליסי האינפרנס I (שזה למעשה BoN). למעשה מאפטמים את הפוליסי (שזה מנגנון חיזוי של LLM או בפשטות LLM עצמו) כדי לעשות את BoN בצורה הטובה ביותר על הדאטהסט שיש לנו. עבור IA-RL המטרה היא לאפטם את הפוליסי (שזה LLM כאמור) תחת טכניקה של אינפרנס I (כלומר BoN) כך שהיא ימקסם את פונקצית תגמול R.

לאחר מכן המאמר מגדיר באופן מדויק מה זה BoN (נוסחה 1) כאשר המטרה היא למקסם את איכות התשובות של המודל כאשר אנו בוחרים את התשובה לפי מה שנקרא verifier score (= ציון לאיכות התשובה). דרך אגב מתווספים כאן שני פרמטרים נוספים שהם מספר התשובות שמהן בוחרים את התשובה הכי טובה וגם טמפרטורת T של מודל השפה. באופן אינטואיטיבי ככל ש T גבוהה יותר (יותר רנדומליות ויצירתיות התשובות) מספר התשובות N צריך לעלות.

כאן יש לנו כאן הטרייד-אוף הקלאסי בין exploration ל-exploitation. ככל ש- T גבוה יותר אנו מבצעים יותר exploration (תשובות מגוונות יותר) ואילו T קטן יותר מאפשר לנו "להנות" ממה שלמדנו עד עכשיו (בחירה של N משפיע על הטרייד-אוף באופן הפוך מ-T).

אוקיי, אבל עדיין בבעיית אופטימיזציה של IA-SFT יש לנו את argmax (משתמשים בו לבחירה של התשובה הטובה ביותר) וזה מאוד מקשה על פתרונה למרות שיש לנו שיטות שערוך argmax באמצעות softmax וגם Gumbel softmax עדיין שיטות אלו אינן מדויקות וגם כבדות חישובית (לטענת המאמר). אז המחברים משתמשים בטריק מאוד מפורסם ב-ML - קירוב של פונקציית יעד עם קירוב וריאציוני שהופך אותה (את הלוג שלה) לסכום של הפוליסי (הסתברות של תשובה y בהינתן שאלה x עם המודל) ושל איבר רגולריזציה הנקרא inference aware. איבר זה הוא למעשה הוא win-rate של תשובה y על שאלה x על המודל הנוכחי כאשר הערך של כל זוג (x, y) מחושב עם verifier score r (עם קבוע נרמול).

ב- IA-RL הסיפור קצת מסתבך והמחברים משתמשים בתוצאה מאחד המאמרים של ג'ו שולמן (cto לשעבר של openai) עם סרגיי לוין ופיטר אבל האגדיים כדי לקבל שערוך לגרדיאנט שמקבל צורה דומה לאלגוריתם הישן והידוע REINFORCE כלומר המכפלה של הלוג של הפוליסי עם פונקציית תגמול (״ממורכזת״ עם התוחלת של פונקציית תגמול להקטנת השונות). המאמר גם דן במקרים מעניינים של אופטימיזציה של פונקצית היעד של IA-RL לכמה צורות של verifier score r (למשל בינארי).

מאמר די כבד מתמטית ניסיתי (לפי מיטב יכולתי) להנגיש לכם אותו טיפה...

https://arxiv.org/abs/2412.15287

המאמר היומי של מייק - 02.01.25:
Loss of plasticity in deep continual learning

היום סוקרים קצרות מאמר די קליל מ-nature.

מבוא:

שיטות למידה עמוקה סטנדרטיות מציגות ירידה הדרגתית ביכולתן ללמוד משימות חדשות בצורה מתמשכת(״מוסיפים״ למודל משימה בצורה הדרגתית). בניגוד לשכחה קטסטרופלית(catastrophic forgetting), שבה ידע קודם אובד, אובדן פלסטיות מגביל את יכולת הרשת ללמוד משימות חדשות ביעילות.

ניסויים מקיפים על דאטהסטים כמו ImageNet ו-CIFAR-100, כמו גם תרחישי למידה עם חיזוקים (Reinforcement Learning), חשפו שהנוירונים הופכים רדומים (לא משתנות בכל הדוגמאות) או מתמחות יתר על המידה על משימה ספציפית, מה שמפחית את יכולתן להסתגל לדאטה חדש. לאורך זמן, רשתות החוות למידה מתמשכת מתפקדות לא טוב יותר ממודלים רדודים (לינאריים), מה שמדגיש מגבלה בסיסית של שיטות מבוססות מורד הגרדיאנט(gradient descent) ללמידה מתמשכת (ואנו מאמנים מודלים עם GD היום)....

מורד הגרדיאנט ללמידה מתמשכת:

שיטות למידה מתמשכת מנסות להתמודד עם אובדן פלסטיות על ידי אתחול מחדש של נוירונים רדומים (כאלו שלא ״נדלקים כמעט אף פעם) ואימונם מחדש עם מורד הגרדיאנט. ככה גישה זו מנסה ״ליצור״ על נוירונים שילמדו משימה חדשה בלי להינעל על למשימות מסוימות, וזה שמאפשר לה ללמוד משימות חדשות ללא הידרדרות משמעותית בביצועים.

בניגוד לשיטות קונבנציונליות המסתמכות אך ורק על מורד הגרדיאנט, GD ללמידה מתמשכת מתאפיין בעדכון הדרגתי סטים שונים של משקלי המודל בדומה למה שקורה במערכות למידה ביולוגיות.

שיטות אימון נוספות:

כאמור אובדן פלסטיות קשור לאופטימיזציית יתר (לטענת המאמר) של משקולות והופעת נוירונים רדומים ברשת. נוירונים אלו אלה או מפסיקים לתרום ללמידה (עבור אקטיבציית ReLU) או נכנסות למצב רוויה(מגיעות ל 0 או 1 עבור סיגמואיד). טכניקות כמו רגולריזציית L2 מפחיתות את גדילת משקלי המודל ושומרות על "פלסטיות" (גמישות למשימות חדשות) במידה מסוימת. למשל שיטת Shrink and Perturb, המשלב רגולריזציה עם שינויים אקראיים קטנים במשקולות, מפחית את תופעת הנוירונים הרדומים וכך מגדיל את יכולת למידה של המודל.

אתגרי למידה מתמשכת ב-RL

למידה מתמשכת היא חיונית גם ל-RL אפילו יותר מאשר בבלמידה מפוקחת. לא רק שהסביבה יכולה להשתנות, אלא גם ההתנהגות של הסוכן הלומד יכולה להשתנות, ובכך להשפיע על המידע שהוא מקבל גם אם הסביבה נשארת קבועה. מסיבה זו, הצורך בלמידה מתמשכת הוא לעתים קרובות יותר ברור בלמידה עם חיזוקים, וRL היא סביבה חשובה להדגמת הנטייה של למידה עמוקה לאובדן פלסטיות. והמאמר בוחן שימוש בשיטות שדנו בהם קודם למשימות של RL יחד עם PPO, האלגוריתם המפורסם לאופטימיזציה ב-RL

https://doi.org/10.1038/s41586-024-07711-7

המאמר היומי של מייק - 03.01.25:

A PERCOLATION MODEL OF EMERGENCE: ANALYZING TRANSFORMERS TRAINED ON A FORMAL LANGUAGE

מבוא:

רשתות נוירונים מודרניות, במיוחד מודלי שפה גדולים , מציגות מגוון רחב של יכולות, המאפשרות להן לשמש כמערכות בסיס למגוון יישומים. מאמר זה מציע הגדרה פנומנולוגית של אמרגנטיות בהקשר של רשתות נוירונים, תוך התמקדות באופן שבו מבנים ותהליכים ספציפיים המונחים בבסיס תהליך יצירת דאטה יכולים להוביל לשיפורים פתאומיים בביצועים במשימות ממוקדות יותר.

מושג חשוב:

הפנומנולוגיה היא גישה פילוסופית המתמקדת בחקר מבני התודעה(consciousness) כפי שהם נחווים מנקודת המבט של האדם. היא שואפת לתאר תופעות או הופעת הדברים כפי שהן נתפסות על ידי בני אדם, ללא הנחות מוקדמות או הטיות תיאורטיות. שיטה זו מדגישה את הבנת החוויות כפי שהן נחיות, במטרה לחשוף את המשמעויות הטבועות בהן

יכולות אמרגנטיות(emergent capabilities) ברשתות נוירונים:

החוקרים מגדירים אמרגנטיות ברשתות נוירונים כרכישת מבנים ספציפיים הגורמים לצמיחה פתאומית בביצועים במשימות ספציפיות. הם חוקרים זאת אמפירית באמצעות מערכת ניסויית המבוססת על שפה פורמלית תלוית-הקשר, ומדגימים שטרנספורמרים שאומנו על מחרוזות משפה זו מציגים יכולות אמרגנטיות. ברגע שהמודל לומד את הדקדוק והמבנים הבסיסיים, הביצועים במשימות קשורות משתפרים משמעותית.

הגדרת השפה הפורמלית:

המערכת הניסויית שהוצעה במאמר משתמשת בדקדוק חופשי-הקשר הסתברותי (PCFG) להגדרת שפה פורמלית תלוית-הקשר. הדקדוק כולל:

סימבולים סופיים(terminal symbols): חלקי דיבור הכוללים נושאים, מושאים, פעלים, תארים, פועלים, מילות חיבור ומילות יחס.

סימבולים לא-סופיים: סמלים המגדירים את מבנה המשפטים.
חוקי יצירת טקסט: חוקים המכתיבים כיצד ניתן לשלב סמלים סופיים ולא-סופיים ליצירת משפטים תקפים.

המודל מאומן על משימות כמו יצירה חופשית, פתרון בלבול וייצור מותנה, כאשר מדדי הביצועים נעקבים לאורך תהליך האימון.

משימות ופרוטוקולי הערכת ביצועי מודלים:

1. **יצירה חופשית של טקסט:** המודל מייצר משפטים העומדים בחוקים הדקדוקיים.
2. **תיקון טקסט לא תקין:** המודל מסדר מחדש מחרוזת מבולבלת של מילים ליצירת משפטים תקפים.
3. **יצירה מותנית:** המודל יוצר משפטים על בסיס ישויות או תכונות נתונות.

ההערכה מתבצעת לפי המדדים כוללים בדיקות דקדוקיות, בדיקות טיפוס, דיוק התאמה מדויקת, דיוק פר-טוקן ועוד, המספקים הערכה מקיפה של יכולות המודל.

תוצאות: דינמיקת הלמידה

התוצאות מגלות 3 שלבים מובחנים בדינמיקת הלמידה של המודל:

1. **שלב ראשוני:** המודל לומד מבנים דקדוקיים בסיסיים עם שיפור מינימלי בביצועים.
2. **"שינוי פאזה"**: מתרחשת עלייה פתאומית בביצועים ברגע שהמודל מתחיל "להבין את אילוצי שפה" פשוטים יחסית.
3. **שלב ההכללה:** המודל מדגים ביצועים משופרים במשימות, המעידים על מעבר משינון להכללה.

יכולות אמרגנטיות של מודלים:

החוקרים מבחינים שככל שמודל השפה לומד את הדקדוק ואילוצי הטיפוס, נצפים שיפורי ביצועים משמעותיים במגוון משימות, במיוחד בפתרון בלבול וייצור מותנה. הנוכחות של מבנים ספציפיים מאפשרת למודל לבנות "שילובים מורכבים ותקינים" של ישויות ותכונות, המובילים ליכולות אמרגנטיות בתחום השפה.

נקודת מעבר בלמידה:

המאמר דן באופן שבו הופעת יכולות האמרגנטיות קשורה למספר התכונות התיאוריות שהמודל למד. נקודת המעבר, שבה מתרחשים שיפורי ביצועים משמעותיים, קשורה לסקיילינג של תכונות תיאוריות. קביעה זו מאפשרת לחזות מתי יכולות יופיעו ככל שהמודל ממשיך ללמוד.

מסקנה:

מחקר זה תורם להבנת האמרגנטיות ברשתות נוירונים על ידי יצירת מסגרת המגדירה ומאפיינת תכונות אמרגנטיות על בסיס רכישת מבנים בסיסיים על ידי המודל. הממצאים מצביעים על כך שאילוצים דקדוקיים ואילוצי שפה אחרים משמשים כגורמים חשובים בחיזוי התפתחות יכולות במודלים של שפה.

https://arxiv.org/abs/2408.12578

המאמר היומי של מייק - 06.01.25
A Survey on Efficient Inference for Large Language Models

המאמר מספק סקירה מקיפה של שיטות לייעול היסק (אינפרנס) ב-LLMs. אז יאללה בואו נסקור את הסקירה.

אתגרים מרכזיים:

1. **גודל המודל:** מודלי שפה גדולים (ענקיים הכוונה) דורשים משאבי חישוב וזיכרון משמעותיים.

2. **סיבוכיות ריבועית** (למרות שיש לא מעט שכלולים כמו FlashAttention) **של מנגנון ה-attention**: מורכבות זו (ביחס לאורך אורך הקלט) משפיעה משמעותית על קצב ההיסק(throughput ו- latency) וצריכת הזיכרון.

3. **פענוח אוטורגרסיבי:** יצירת טוקנים אחד אחרי השני לא מנצלת באופן מיטבי את משאבי החישוב (כמו GPU) העומדים לרשותנו ופוגעת בתפוקת המודל (throughput)

טקסונומיה של טכניקות אופטימיזציה:

1**. אופטימיזציה ברמת הדאטה:**

דחיסת קלט: טכניקות כמו חיתוך(pruning) פרומפטים, סיכום(summarization) פרומפטים, דחיסה מבוססת פרומפט רך (למידה של וקטורים רציפים "המייצגים" את הפרופמט) , והיסק מבוסס RAG מפחיתות את גודל פרומפטי הקלט תוך שמירה על מידע סמנטי בן. זה יעיל במיוחד לתרחישים הדורשים קלטים ארוכים יותר.

ארגון פלט: שיטות כמו Skeleton-of-Thought (SoT) וגישות מבוססות גרף תלות מאפשרות מקביליות חלקית של גנרוט טוקנים, תוך ניצול המבנה הפנימי של פלטי LLM.

2**. אופטימיזציה ברמת המודל:**

תכנון מבנה יעיל:

- שיטות כמו Mixture-of-Experts (MoE) מקצים משאבי חישוב באופן דינמי לטוקני קלט, תוך אופטימיזציה של חלקי רשתות MLP הפנימיות בבלוק הטרנספורמר(במימד האמבדינג בד"כ).
- מנגנוני attention מפושטים או מבוסי-kernel (כמו Performer שסקרתי בזמנו) מפחיתים סיבוכיות מריבועית לליניארית (ביחס לאורך הקלט).
- חלופות לטרנספורמרים, כמו State Space Models(SSMs), כה האהובים עליי, וארכיטקטורות RNN (מתברר שיש פה ושם שימוש בהם) מקטינות את סיבוכיות המודל תוך שמירה על ביצועים תחרותיים (לפעמים). בהקשר זה כדאי להזכיר את Jamba של A21 labs ששילבו ארכיטטקטורת טרנספורמרים עם ממבה (סוג של SSM)

דחיסת מודל:

- **קווינטוט:** מפחית רוחב סיביות למשקולות והפעלות. שיטות כימות לאחר אימון ואימון-מודע-כימות שומרות על דיוק למרות הדחיסה.
- **דילול:** מסיר פרמטרים או ראשי תשומת לב מיותרים, באמצעות טכניקות כמו pruning או מנגנוני attention דלילים.

- **זיקוק ידע**(distillation): מאמן מודלים קטנים יותר לחקות את התנהגות המודלים הגדולים, עם אובדן ביצועים מינימלי.

3**. אופטימיזציה ברמת המערכת:**

שיפורים במנועי היסק (למשל, פענוח ספקולטיבי ואסטרטגיות offloading) ומערכות שירות (למשל, חישוב בבאצים, scheduling מתוחכם וניהול זיכרון) משפרים את ניצול החומרה וביצועי המודל (מבחינת ה-throughput).

המאמר מציין שתהליך ההיסק מחולק לשני שלבים:

1. מילוי מקדים(prefilling): אתחול המודל עם פרומפטי קלט העלאה של זוגות KV שישמשו לגנרוט הטקסט.

2. פענוח: יצירת טוקנים רציפה עם תקורת זיכרון וחישוב.

גישות ניתוח יעילות:

מדדי יעילות כמו השהיה (לטוקן ולרצף כולל), שימוש בזיכרון (משקולות מודל, KV cache, צריכת זיכרון מקסימלית), ותפוקה (טוקנים/שנייה, בקשות/שנייה) מנותחים כדי לכמת את ההשפעה של שיטות אופטימיזציה הנבחנת.

כיוונים עתידיים:

1. טכניקות אדפטיביות המתאימות דינמית את גודל המודל והחישוב בהתבסס על מורכבות הקלט.
2. אופטימיזציה משותפת בכל הרמות - דאטה, מודל ומערכת - למקסום היעילות.
3. שיטות מודעות-חומרה לניצול מאיצים מודרניים כמו GPUs ו-TPUs.

https://arxiv.org/abs/2404.14294

המאמר היומי של מייק - 07.01.25
Anchored Preference Optimization and Contrastive Revisions Addressing Underspecification in Alignment

המאמר שנסקור היום מציע שיפור לשיטת יישור (alignment) למודלי שפה, DPO, השייכת למשפחת טכניקות RLHF או Reinforcement Learning with Human Feedback. כמו שאתם זוכרים RLHF הינו אחד השלבים(האחרון בד"כ) לאימון LLM יחד עם אימון מקדים (pretraining) ו-Supervised Fine Tuning או SFT בקצרה.

מטרת RLHF היא להראות למודל מה ההבדל בין תשובות מועדפות (על ידי בני אדם) מתשובות פחות מועדפות. בנימה יותר מתמטית RLHF מאמנת את המודל למקסם את היחס בין הציון של התשובה מועדפת (טובה) יותר לבין תשובה פחות טובה. שיטת RLHF קלאסית Proximal Policy Optimization מוסיפה לאיבר הממקסם פונקציית לוס איבר רגולריזציה המנסה לשמור את הפוליסי הנלמד (כמו LLM מאומן) קרוב ל-LLM ההתחלתי (הקרבה מחושבת עם KL על ההתפלגות של הטוקנים החזויים על ידי שני המודלים).

הציון מחושב על מודל תגמול (reward model) שמאומן (בלשב הקודם ל-RLHF) לשערך את "איכות" התשובה לשאלה נתונה. כלומר מודל תגמול R אמור לתת ציון גבוה לתשובה טובה וציון נמוך לתשובה פחות טובה. המודל מאומן על זוגות של תשובות טובות ולא טובות לשאלות, כאשר בד"כ התיוג של התשובות מתבצע על ידי מתייגים אנושיים (לפעמים רותמים מודל שפה עוצמתי לתיוג הזה).

התברר שניתן לקרב את יעד האופטימיזציה של PPO ללא אימון של מודל תגמול. בשנתיים האחרונות יצאו לא מעט מאמרים שהציעו שיטות ש"יודעות" להסתדר ללא מודל תגמול. אחת מהן היא DPO שזה ראשי תיבות של

Direct Preference Optimization. עם DPO פונקצית תגמול מוגדרת r_dpo בתור לוגריתם של היחס בין הפוליסי (ההתפלגות החזוי של טוקנים הנמדדת על ידי המודל או נראות- likelihoods) עבור המודל המאופטם (שעובר פיין טיון) לבין זה של המודל ההתחלתי. מטרת אימון DPO היא למקסם את הפרש בין התוחלת (עבור הדאטהסט של זוגות שאלות ותשובות) ההפרש של r_dpo ביו התשובות המועדפות לבין פחות מועדפות.

הנקודה העיקרית של המאמר היא האובזרבציה שהאופטימיזציה של פונקצית המטרה של DPO עלולה להשפיע באופנים שונים על יחס הנראויות (likelihoods) של תשובות המועדפות w לאלו של פחות מועדפות l. היא כמובן יכול להגדיל את ההפרש ביניהם (שזה המטרה המוצהרת שלה) אבל יכול להגדיל את p_w יותר מאשר הוא מגדיל p_l, או להקטין את p_l יותר מאשר הוא מקטין את r_w. תרחישים אלה עשויים להוביל ליצירת מודלים שונים מאוד. המאמר מציין שתשובה מועדפת אינה בהכרח טובה יותר ממה שהמודל מייצר לפני היישור. במקרה זה, DPO עלול לפגוע בביצועי המודל.

המאמר מתבונן במקרים השונים של ערכי r_dpo עבור התשובות w ו- l(מועדפת ופחות מועדפת בהתאמה) ובונה שתי פונקציות מטרה ל- DPO שעשויות להוביל לביצועים טובים יותר עבור מקרים אלו. שיטת אימון שמאפטמת פונקציות אלו קיבלה שם Anchored Preference Optimization או APO. הפונקציה המוצעת הראשונה מגדילה את ערך הפוליסי (נראות של תשובה) כאשר הערך הנוכחי של r_dpo עבור w קרוב ל-0 (w הינה בעלת נראות נמוכה יותר עבור המודל ההתחלתי) ומקטינה את הנראות של התשובה הפחות מועדפת עוד יותר אם r_dpo עבור l קרובה ל-0.

הפונקציה המוצעות השניה לעומת זאת מקטינה את הנראות של w כאשר r_dpo קרוב ל -0 עבור w ומגדילה את ההפרש בין הנראויות של w ו- l כאשר ההפרש בין r_dpo עבור w ו- l קרוב ל-0. כל זה במטרה לגרום למודל שפה המאומן באמצעות DPO להתכנס לפתרון טוב יותר.

יש עוד משהו מעניין במאמר הזה. המחברים טוענים שכדי ש- DPO יעבוד בצורה טובה יותר, שתי התשובות(w ו- l) צריכות להיות רלוונטיות לשאלה ואחת מהן צריכה להיות "רק קצת" יותר טובה מהשנייה. כלומר במו בלמידה ניגודות עדיף לאמן את המודל על hard negatives.

המחברים מציעים שיטה לזיהוי (ובניית דאטהסט) של תשובות מועדפות ופחות מועדפות והיא יצירת תשובה מועדפת מתשובה כלשהי(אך רלוונטית) על ידי הפעלת LLM המשפר את התשובה (עם פרומפט מתאים). שיטה אחרת שהמחברים מציעים להשתמש בה היא בהינתן שתי תשובות של המודל המאומן (עם DPO) להפעיל מודל שפה שמטרתו להגיד מהי תשובה טובה יותר (זה נקרא on-policy judge). ניתן גם לבנות דאטהסט באופליין עם מודל שפה שלישי ומודל שופט.

סקירה ארוכה - אני מקווה ששרדתם...

https://arxiv.org/abs/2408.06266

המאמר היומי של מייק - 09.01.25

When Can Transformers Count to n?

המאמר חוקר את המגבלות התיאורטיות והאמפיריות של ארכיטקטורות טרנספורמר כאשר בביצוע משימות ספירה פשוטות. הוא בוחן משימות כמו "ספירת שאילתות" (QC) ו"האלמנט השכיח ביותר" (MFE) כדי לקבוע מתי טרנספורמרים יכולים לפתור בעיות אלה ביעילות. המחקר חושף הן את היכולות והן את המגבלות המובנות של טרנספורמרים בהקשרים כאלה, ומספק תובנות מעניינות לגבי האילוצים הארכיטקטוניים שלהם.

התרומות העיקריות:

משימת QC

משימת QC היא למעשה ספירה של כמה פעמים טוקן מסוים מופיע ברצף. המחברים מדגימים שהטרנספורמרים יכולים לבצע משימה זו ביעילות אם גודל האמבדינג d גדול מפי שניים מגודל המילון m: עבור d > 2m, גישה מבוססת היסטוגרמה (שפותחה במאמר) מאפשרת ספירה על ידי הטמעת ייצוגי טוקנים בצורה אורתוגונלי. זה מאפשר למודל לבנות היסטוגרמה של הופעות טוקן על ידי בלוק טרנספורמר יחיד. עבור d < m, האורתוגונליות של המבדינגס כבר לא אפשרית, מה שהופך ספירה מדויקת לבלתי אפשרית. המאמר מוכיח מגבלה זו בקפידה באמצעות חסמי Welch, המאפשר לנתח את את הטרייד-אופים של מימד האמבדינגס (הקשורים לאורתוגונליות).

שיטת CountAttend

כאשר גודל אמבדינגס d קטן מגודל המילון m, המחברים מציעים את פתרון ה-"CountAttend", כדי לפתות את QC עם מנגנוני ה-attention. הפתרון כולל שני רכיבים עיקריים:

1. **משקלי attention**

- מנגנון attention מייצר משקלים המקודדים את היחס בין טוקן השאילתה לכל הטוקנים ברצף. לצורך ספירה, משקלי attention חייבים להיות הפוכים ביחס לספירת הטוקן ברצף.שקלול זה מבטיח שתרומת כל אסימון לפלט מנורמל לפי התדירות של - זה מבטיח שתרומה של כל טוקן לפלט מנורמלת לפי התדירות שלו.

2. **MLP להיפוך משקלים**

- נדרש MLP כדי לשחזר את הספירה האמיתית c ממשקלי attention. ה-MLP צריך ללמוד פונקציה מהצורה: f(w) = 1/w כאשר w הוא משקל attention עבור טוקן

אתגרים עם פתרון CountAttend:

חישוב משקלי תשומת לב: חישוב משקלים הפוכים ביחס לספירות טוקנים דורש מידול מדויק של יחסי טוקנים לאורך הרצף. זה מוסיף מורכבות למנגנון ה-attention

גודל MLP: עבור סדרות ארוכות יותר, מספר הנוירונים ב-MLP חייב לגדול באופן פרופורציונלי ביחס לאורך הסדרה שזה בעייתי מאוד מבחינה חישובית.

משימת MFE:

מטרת משימת ה-MFE, היא למצוא טוקן בעל התדירות הגבוהה ביותר בסדרה. ניתן ליישם את המשימה אם d=O(m) באמצעות גישה מבוססת היסטוגרמה. עבור d < m, המשימה הופכת לבלתי אפשרית, כפי שמוכח באמצעות טיעוני מורכבות תקשורת. המחברים מציעים פתרון טרנספורמר דו-שכבתי לבעיה זו.

מעבר פאזה בביצועים

המאמר מזהה מעבר פאזה קריטי: טרנספורמרים נכשלים במשימות ספירה כאשר d < m. סף זה מדגיש את הפשרות בין גודל ההטמעה, גודל אוצר המילים, ומורכבות המשימה.

תובנות תיאורטיות:

בניית אמבדינגס אורתוגונליים

המחברים מנצלים את התכונות המתמטיות של א אורתונורמליות ליישום ספירה מבוססת היסטוגרמה. עבור d > m, ניתן לבנות אמבדינגס כך שמכפלה סקלרית בין הטמעות טוקן שונות היא אפס. זה מבטיח ספירת טוקנים מדויקת בתוך בלוק attention יחידה. המאמר משתמש בגבולות Welch להראות שעבור d < m, המכפלה הפנימית בין וקטורי ההטמעה הופכת משמעותית, מה שמכניס שגיאות בהיסטוגרמה. עבור משימת MFE, המחברים משתמשים בכלים מעולם communication complexity כדי להוכיח שהטרנספורמרים דורשים d=Ω(m) כדי לפתור את המשימה.

השלכות מעשיות:

המסקנות מובילות למספר השלכות לתכנון ופריסה של טרנספורמרים ביישומים מעשיים: סקלביליות ארכיטקטונית: טרנספורמרים חייבים להתאים את גודל אמבדינג לגודל המילון. קידוד מיקומי (positional encoding) המחברים מדגישים את הנחיצות של הטמעות מיקום למשימות ספירה. בעוד שפתרון ההיסטוגרמה יעיל עבור d > m, היישום המעשי שלו עשוי להיות מאוד בעייתי מבחינת הזכרון וסיבוכיות.

מסקנה

המאמר מספק ניתוח מקיף של היכולות והמגבלות של טרנספורמרים בפתרון משימות ספירה בסיסיות. באמצעות שילוב של הוכחות תיאורטיות ריזורוזיות עם אימות אמפירי, הוא מדגיש את הפשרות הארכיטקטוניות המובנות במודלי הטרנספורמרים.

מחקר עתיד:

- ארכיטקטורות היברידיות המשלבות טרנספורמרים עם שיטות ניורו-סימבוליות למשימות ספירה

- הרחבות למשימות הכוללות ספירה היררכית או מובנית

- מחקרי mechanistic interpretability להבהרת הייצוגים הפנימיים שנלמדים על ידי טרנספורמרים במהלך משימות ספירה

https://arxiv.org/pdf/2407.15160

המאמר היומי של מייק - 10.01.25
Chain of Thought Empowers Transformers to Solve Inherently Serial Problems

המאמר מציג ניתוח תיאורטי של כיצד Chain of Thought (CoT) מאפשר למודלי טרנספורמר להתמודד עם חישובים סדרתיים(לא מקביליים). המחברים הוכיחו חסמי expressiveness פורמליים ומציגים מחלקת complexity חדשה (איך ניתן לתרגם complexity class? חוץ ממחלקת סיבוכיות) המאפיינת את יכולות החישוב של טרנספורמרים עם CoT.

התרומה התיאורטית העיקרית של המאמר טמונה בחסמי האקספרסיביות שהוא מוכיח. באמצעות ניתוח מתמטי ריגורוזי, המחברים מוכיחים שהטרנספורמרים בעלי עומק קבוע עם דיוק סיביות קבוע מוגבלים לפתרון בעיות ממחלקת סיבוכיות הנקראת AC0 ללא CoT (משפט 3.1). עם זאת, הם מראים שעם T שלבי CoT, טרנספורמרים מסוגלים לפתור כל בעיה הניתנת לחישוב על ידי שרשרת בוליאניות בגודל T (משפט 3.3). לתוצאה זו יש השלכות עמוקות, שכן היא קובעת שמספר פולינומיאלי של צעדי CoT מאפשר לטרנספורמרים לחשב כל פונקציה במחלקת -P/poly.

המסגרת התיאורטית שפותחה במאמר מורכבת משלושה חלקים עיקריים. ראשית, המחברים מציגים ניתוח מקיף של חישובי low-precision לעומת floating-point בטרנספורמרים. שנית, הם מבססים קשרים עמוקים עם תורת סיבוכיות על ידי הגדרת מחלקת מורכבות חדשה [CoT[T(n), d(n), s(n), e(n)) המאפיין את החישובים בטרנספורמר עבור מספר שלבי CoT המסומן בתור T(n), המימד החבוי של הטרנספורמרים d(n), דיוק ייצוג נומרי s(n) והאקספוננטה שלו e(n). שלישית, הם משלבים תורת אוטומטים על ידי שימוש במשפט הפירוק של קרון-רודס לניתוח יכולות הטרנספורמר.

מבחינה ארכיטקטונית, העבודה מספקת ניתוח מפורט של רכיבי טרנספורמר, כולל מנגנוני self-attention, FFNs, קידודי מיקום, והשפעות שכבות נרמול. ניתוח זה מספק אפיון מדויק של יכולות חישוביות ומבסס מיפויים ברורים בין תכונות ארכיטקטוניות וחסמים תיאורטיים.

השפעת המאמר חורגת מניתוח טרנספורמרים. על ידי הצגת מחלקת סיבוכיות חדשה לחישובי הטרנספורמרים, הוא מגשר בין מודלי חישוב וקלאסיים עם אלו המבוססים למידה עמוקה. הכלים המתמטיים שפותחו משלבים מסגרות תיאורטיות מרובות ביעילות ויוצרים קשרים חדשים בין תחומים נפרדים בעבר. במבט קדימה, עבודה זו פותחת כיווני מחקר מבטיחים רבים, במיוחד בהבנת השימוש המיטבי ב- CoT והמגבלות היסודיות של ארכיטקטורות טרנספורמר. המסגרת התיאורטית שנוסדה כאן צפויה לשמש כבסיס לניתוח חידושים עתידיים בארכיטקטורת רשתות נוירונים ואסטרטגיות הנחיה.

https://arxiv.org/abs/2402.12875

המאמר היומי של מייק - 11.01.25
Evaluating the Design Space of Diffusion-Based Generative Models

מאמר זה מספק ניתוח מקיף של מודלים גנרטיביים מבוססי דיפוזיה על ידי הצגת מסגרת מאוחדת המגשרת בין שלבי האימון והדגימה. הוא בונה בסיס מתמטי מוצק להבנת כיצד בחירות תכנון משפיעות על ביצועי המודל ויעילות החישוב. המאמר מתמודד עם יחסי הגומלין המורכבים בין תהליכי האימון והדגימה במודלי דיפוזיה. בניגוד לעבודות קודמות שלעתים קרובות מבודדות שלבים אלה, מחקר זה מספק ניתוח שגיאה מאוחד המשלב את שניהם.

התרומות העיקריות:

1. **דינמיקת אימון וניתוח התכנסות**

המאמר בוחן את התנהגות של פונקציית המטרה של Denoising Score Matching או DSM במהלך תהליך אופטימיזציה שלה (עם מורד הגרדיאנט - Gradient Descent). באמצעות טכניקות מעולם פונקציות סמי-חלקות(ראו נספח להסבר על כך), הוא מבסס התכנסות אקספוננציאלית(במישור האיטרציות של GD) עבור רשתות עמוקות עם אקטיבציית ReLU ומספק תובנות לגבי פונקציות משקל אופטימליות לאימון (איבר המכמת לוס עבור כל עוצמת הרעש ממושקל באופן שונה בפונקציית לוס ב-DSM).

תובנות מרכזיות בדינמיקת האימון:

פונקציית המשקל בצורת פעמון עולה באופן טבעי מהניתוח במאמר. משקל זה מבטיח שהאופטימיזציה מתמקדת יותר ברמות רעש בינוניות, שבהן יחס האות-לרעש מאוזן, מה שמקל על הרשת הנוירונית ללמוד פונקצית Score (גרדיאנט של לוגריתם של פונקציית צפיפות של נקודת דאטה x) בצורה מדויקות. חסמים על הגרדיאנט שהוצגו

במאמר מסתמכים על הנחות מתוכננות בקפידה לגבי סקאלת וממדיות הדאטה, המשקפות תרחישי אימון מציאותיים. חסמים אלה מבטיחים התכנסות של פונקציית עבור מגוון ארכיטקטורות רשת ולוחות זמנים של עוצמת הרעש (noise schedule או NS). על ידי תרגום הממצאים התיאורטיים להמלצות מעשיות, המחקר מדגיש שבחירת מקדמי משקול בפונקציית לוס היא קריטית להבטחת התכנסות מהירה מבלי לפגוע ביכולת הכללה של הציון הנלמד.

2. **תהליך דגימה וחסמים שגיאה**

תהליך הדגימה במודלי דיפוזיה מסתמך במידה רבה על סימולציה מדויקת של משוואה דיפרנציאלית סטוכסטית (SDE) המדמה תהליך הסרת רעש. ביחס לעבודות קודמות המאמר מוכיח חסמי שגיאה הדוקים יותר, לא-אסימפטוטיים תחת NS כלליים. ניתוח זה מכסה שגיאת אתחול, שגיאת דיסקרטיזציה, ושגיאת קירוב הציון.

מוצג במאמר כי סיבוכיות דגימה(כלומר כמה דגימות נדרשות כדי שרשת נוירונים אקספרסיבית מספיק ללמוד שערוך Score מדויק המספיק לגנרוט דגימות באיכות גבוהה) תהליך הדגימה היא כמעט לינארית במימד הדאטה, בהינתן שנעשה שימוש NS אופטימליים. לתוצאה זו יש השלכות משמעותיות על יכולת ההרחבה של מודלי דיפוזיה, במיוחד ביישומים רבי-ממדים כמו יצירת תמונות. המחברים מציינים איך NS שונים (פולינומיאליים לעומת אקספוננציאליים) נעים בין מזעור שגיאות ועלות חישובית, ומציעים הנחיות ברורות לתרחישי אימון שונים. העבודה גם שופכת אור על משמעות אתחול הרעש והשפעתו על איכות הדגימה הסופית, מקשרת בין חסמי שגיאה תיאורטיים לתוצאות מעשיות.

3. **ניתוח שגיאה מלא**

על ידי שילוב ניתוחי האימון והדגימה, המחברים מפתחים מסגרת הוליסטית לכימות שגיאה end2end במודלי דיפוזיה גנרטיביים. שילוב זה חושף כיצד מקורות שגיאה שונים מתקשרים ומספק מבט מאוחד על הגורמים המשפיעים על איכות הדגימה.

נקודות מרכזיות בניתוח השגיאה:

פירוק שגיאת אופטימיזציה: המחקר מבחין בין שגיאות הקשורות לאימון (שגיאות אופטימיזציה וסטטיסטיות) ושגיאות הקשורות לדגימה (דיסקרטיזציה ואתחול). פירוק זה מבהיר את יחסי הגומלין בין אימון המודל לתהליך הגנרוט. השפעת פרמטריזציית יתר (over-parameterization) של המודל: התוצאות מראות כיצד הגדלת רוחב ועומק הרשת יכולה למתן שגיאות אופטימיזציה, מאפשרת ל-GD להשיג התכנסות אקספוננציאלית.

זה מתיישר עם תצפיות אמפיריות בלמידה עמוקה אך מספק בסיס תיאורטי קפדני. נזכיר כי חסמי השגיאה שהתקבלו תלויים בפרמטרים מרכזיים כמו מימד הדאטה, NS, ופונקציות משקל. עבור NS מעשיים (למשל, EDM), החסמים מתיישרים היטב עם מדדי ביצוע אמפיריים. הניתוח גם מדגיש כיצד שגיאות "מתחלקות" בין שלבי האימון והדגימה, ומציע תובנות לגבי איך לאזן מאמץ חישובי בין שלבים אלה לביצועים גנרטיביים אופטימליים.

נספח:

מהי סמי-חלקות?

סמי-חלקות היא תכונה של פונקציית לוס והגרדיאנט שלה, המבטיחה שצעדי GD מפחיתים את הלוס ביעילות גם כאשר הפונקציה אינה חלקה לחלוטין. עבור רשתות ReLU עמוקות, פונקציית הלוס כוללת לינאריות חלקית, מה שהופך אותה ללא-חלקה באופן כללי. תכונת הסמי-חלקות מבטיחה שהגרדיאנט מספק כיוון "טוב" לירידה למרות

חוסר החלקות. קיימים חסמים תחתונים על נורמות הגרדיאנט, המבטיחים התקדמות עקבית לקראת מזעור הלוס. על ידי ניצול הסמי-חלקות, המחברים מבססים קשר מתמטי בין ערך הלוס וגודל הגרדיאנט שלו, המאפשר להם להוכיח דעיכה אקספוננציאלית בשגיאת האופטימיזציה.

https://arxiv.org/abs/2406.12839

המאמר היומי של מייק - 13.01.25
Improve Mathematical Reasoning in Language Models by Automated Process Supervision

מזמן רציתי לכתוב סקירה על MCTS שזה Markov Chain Tree Search ולגמרי במקרה נתקלתי במאמר הזה המציע ליישם את השיטה המגניבה הזו עבור אימון LLMs. הפעם המטרה לאמן מודל שפה לפתור בעיות מתמטיות (לוגיות) מורכבת שפתרונם מכיל שלבים רבים.

קודם כל הסבר קצר מה זה בעצם MCTS. חיפוש עץ מונטה קרלו (MCTS) הוא אלגוריתם לאופטימיזציה של פוליסי עבור תהליכי החלטה מרקוביים (Markov Decision Process) בעלי אופק סופי וגודל סופי, המבוסס על דגימת אפיזודות אקראיות המאורגנות באמצעות עץ החלטה.

י. הוא עובד 4 שלבים:

1. **בחירה**: בוחרים מסלול מהשורש לעלה לפי פוליסי חקירה/ניצול (exploration/exploitation)
2. **הרחבה**: מוסיפים מצב חדש לעץ
3. **סימולציה**: מריצים סימולציה אקראית מהמצב החדש עד סוף המשחק
4. **עדכון לאחור**: מעדכנים את הערכים בכל הצמתים במסלול שנבחר

אנו משתמשים ב-MCTS כדי לשפר את המדיניות (policy) על ידי בחירת פעולות טובות יותר. המודל מספק הערכות למצבים במקום סימולציות אקראיות ו-MCTS משתמש בהערכות אלו כדי לבנות עץ חיפוש יעיל יותר. לדוגמה, AlphaGo משתמש ב-MCTS בשילוב עם רשתות עמוקות כדי לבחור מהלכים. היתרון העיקרי של MCTS הוא בין חקירת מצבים חדשים (exploration) לבין ניצול ידע קיים(exploitation), ומשפר את קבלת ההחלטות לאורך זמן.

המאמר שנסקור היום מציע להשתמש בגישת MCTS כדי לאמן מודל שפה לבנות תשובות בעלות שלבים רבים וכמו שאתם יכולים לנחש הצמתים בגרף הזה יהיו השלבים בפתרון. המאמר מציין פתרונות SOTA לאימון מודלי שפה לפתור בעיות אלו מתחלקים לשני סוגים. הראשון מסמלץ את כל שלבי הפתרון כך שהמודל מאומן (עם טכניקות RLHF לבחירתכם) למקסם את הפרס שהמודל מקבל בסוף (בד״כ בינארי, כלומר האם הפתרון נכון/לא נכון) עם איזשהו איבר רגולריזציה (קירבה למודל המקורי).

השיטה השנייה PRM עושה דבר דומה אבל למסלולים חלקיים (=כמה שלבי פתרון בהתחלה). ניתן לראות שהגישה הראשונה תעבוד פחות טוב עבור בעיות עם הרבה שלבים כי ה-reward מאוד דליל (sparse) וקשה לאופטימיזציה. המקרה השני צריך הרבה דאטה מתויג איכותי וזה מאוד יקר.

המאמר כאמור מציע להשתמש ב-MCTS למטרה זו. כמו שמקובל ב-MDP אנו צריכים להגדיר מה זה המצב, פעולה ותגמול. המצב s מוגדר בתור שלאה q, כל שלבי הפתרון עד עכשיו (לא חייב לכלול את הפתרון) והפעולה a היא בחירת הצומת הבאה שבמקרה הזה הוא שלב הבא של פתרון שאלה q. לאחר שהפעולה a נבחרת היא מתווספת ל-s כלומר המצב החדש הוא (s_old, a). הפעולה a נבחרת על ידי פוליסי p(a|s) כאשר עבור MCTS

הוא מורכב משני מחוברים: הראשונה (exploitation) נוטה לבחור צמתים בעלי תגמול גבוה והאיבר השני (exploration) מעדיף צמתים שלא ביקרנו בהם הרבה.

עכשיו הגיע הזמן לדבר עם התגמול (reward). עבור צומת נותן v התגמול שלו הוא אחוז ה-rollouts הנכונים(המסומן בתור c) שהתחילו משלב v (אחוז המסלולים בגרף שהגיע לפתרון הנכון החל מ v). דרך אגב יש שיטה מאוד אינטואיטיבית לזיהוי של הטעות הראשונה בפתרון לא נכון (שכמה מעבודות קודמות מצאו כמידע יעיל לאימון מודל) שמאפשרת לזהות צמתים "לא נכונים בהחלט" (שמהם לא ניתן להגיע לפתרון הנכון) בפתרון שנקראת "חיפוש בינארי.

השיטה כל פעם מחלקת את מסלול הפתרון לשניים ובודקת היום c עבור הצומת שנמצא בחצי המסלול גדול או קטן מ-0. אם הוא שווה לאפס אז הטעות כנראה בחצי הראשון ואם הוא גדול מ-0 אז הטעות כנראה בחצי השני. אז שוב מחלקים לחצי את החצי שבו אנו חושדים שיש טעות וממשיכים לצמצם את החיפוש עד שמגיע ל"צומת המטעה".

כדי להגדיל את מספר הדוגמאות המחברים מציעים לאחסן rollouts של הפתרון ולבצע חיפוש בינארי של הצומת שבו (ככל הנראה) קרתה טעות ולהתחיל ממנה חיפוש חדש. זה מאפשר לבנות דוגמאות עם אותם השלבים ההתחלתיים והמשך שונה. אזכיר שעם גישת PRM (שעליה המאמר בונה את הפתרון) כל דוגמא היא השלישיה של שאלה, פתרון חלקי, וציון האם זה נכון. כל אלו אנו מקבלים בתהליך המתואר כאן.

לבסוף המאמר משתמש ב-MCTS עם פוליסי Q כאשר המצב של כל צומת בגרף הפתרון מתואר על ידי שלישיה (אחרת) שהיא מספר הפעמים שהפתרון ביקר בצומת הזה, אחוז הפתרונות הנכונים c מהצומת הזו (כלומר שערוך מונטה קרלו שלו) וגם ערך של פוליסי Q שהוא מקבל ערך גבוה עבור ערך של C קרוב ל 1(צומת מוביל לרוב לפתרון הנכון) ויש לו איבר רגולריזציה (כפלי) הקונס אותו על פתרונות ארוכים יותר. בחירה של מסלול rollout נבחר על ידי דגימה שנבנית בהתבסס על הסטטיסטיקה של העץ עם האלגוריתם שנקרא PUCT (נוסחה 3 במאמר). כמובן Q, c וסטטיסטיקה של העץ מתעדכנות במהלך MCTS.

זהו זה - סקירה מאוד ארוכה, מקווה שהצלחתי להסביר אותו, מאמר לא טריוויאלי...

https://arxiv.org/abs/2406.06592

המאמר היומי של מייק - 16.01.25
Diffusion Models for Non-autoregressive Text Generation: A Survey

היום נסקור סקירה מלפני שנה וחצי של תחום (משפחת טכניקות) אז מטבע הדברים זה הולך להיות די קצר. הסקירה היא על שיטות גינרוט טקסט לא אוטורגרסיביות כלומר לא טוקן אחרי טוקן אלא סדרה שלמה. השיטות שנדבר עליהן מגנרטות טקסט בכמה איטרציות אבל זה לא נעשה בצורה אוטורגרסיבית - למשל שיטות אלו יכולת לגנרט טוקן מספר 78 לפני טוקן מספר 24.

אוקיי, בטח כמה מכן חשבו על מודלי דיפוזיה גנרטיביים אחרי שהזכרתי שיטות איטרטיביות ואתם לא טועים כאן. בסקירה קצרה זו אסביר בצורה מתומצתת אין ניתן לגנרט טקסט עם מודלי דיפוזיה. כמו שאתם בטח זוכרים מודלי דיפוזיה מאומנים להסיר רעש מדאטה מורעש וזה נעשה באיטרציות. כלומר המודל מאומן להסיר כמות קטנה של רעש מהדאטה עד להגעה לדאטה נקי וכך לאחר האימון המודל מסוגל לגנרט דאטה מרעש טהור בכמה איטרציות.

אבל איך ניתן להוסיף רעש לטקסט שחי במרחב דיסקרטי (כלומר טוקנים). יש בגדול שתי גישות: הגישה הרציפה והגישה הדיסקרטית. בגישה הרציפה שהיא יותר פשוטה וקרובה ליבנו אנו לא פועלים במרחב הדיסקרטי אלא

במרחב של אמבדינגס. בגישה הרציפה אנו הופכים את הטקסט שלנו לוקטור אמבדינג רציף אבל להבדיל אנקודר רגיל אנו הופכים כל טוקן לייצוגו הווקטורי בנפרד מהאחרים. לאחר מכן מאמנים מודל דיפוזיה לגנרט אמבדינג של טקסטים. הוספת רעש ואימון מודל denoising מתרחשים במרחב האמבדינג כאשר המטרה היא הסופית היא לשחזר את הטוקנים מהאמבדינגס (ד״א יש כמה שיטות לעשות את זה) אחרי ניקוי רעש.

משפחת השיטות השנייה היא לבצע הוספת רעש במרחב הדיסקרטי. מובן שהרעש לא יכול להיות רציף אז מה שניתן לעשות היא לשנות את ערכי הטוקנים (למשל לטוקן [mask]) בהסתברות מסוימת כאשר המטרה היא באיטרציה האחרונה להפוך את כל הטוקנים ל-[mask]. מודל דיפוזיה באיטרציה i מאומן לחזות את הטוקנים מהאיטרציה הקודמת, כאשר באינפרנס הגנרוט מתחיל מכך שכל הטוקנים שווים ל-[mask] והמודל לאט לאט הופך אותם לטקסט.

כמובן שאופן הרעשה של טוקן בכל איטרציה זה הייפרפרמטר השקול ל-noise schedule במודלי דיפוזיה רגילים. ניתן לתאר אופן הרעשה בתור מטריצה. כל טוקן ניתן לייצוג על ידי וקטור ההסתברות (מעל מילון הטוקנים) אז ניתן לייצוג טוקן מאיטרציה i כמכפלה פנימית של ייצוגו באיטרציה i-1 על ידי מטריצה סטוכסטית Q_i (סכום של שורות ועמודות הינו 1). Q_i היא הייפרפרמטר הכי חשוב במודלי דיפוזיה דיסקרטיים.

מתברר שזה תחום מחקר די פעיל למרות עדיין מודלים אלו לא הגיעו לביצועים של מודלי שפה אוטורגרסיביים. אבל אני לא פוסל שזה עוד יקרה כי מודלים אלו מסוגל לעבוד בתפוקה גבוהה יותר ממודלים אוטורגרסיביים (עבור מספר צנוע של איטרציות).

https://arxiv.org/abs/2303.06574

המאמר היומי של מייק - 17.01.25

Towards a Unified View of Preference Learning for Large Language Models: A Survey

מוטיבציה

המאמר מספק סקירה נרחבת של שלב מהותי באימון LLMs: יישור (alignment) של פלט המודל עם העדפות אנושיות. מיותר לציין כי יישור זה חיוני ליישומים רבים LLMs. בעוד ש RLHF וכיוונון מונחה (SFT) היו מרכזיים ליישור, היחסים ביניהם נותרו לא נחקרו מספיק, מה שמוביל לפיצול המאמצים המחקריים בנושאים אלו.

המחברים שואפים לאחד מאמצים מפוצלים אלה על ידי הצגת מסגרת המשלבת גישות RLHF ו-SFT תחת נוסחה מבוססת גרדיאנט אחת בלבד. איחוד זה לא רק מגשר על פערים מתודולוגיים אלא גם מכין את הקרקע להתקדמויות מגובשות יותר בלמידת העדפות (preference learning). המאמר מדגיש יישור כולל מספר מרכיבים - מודל, דאטה,משוב (כגון פונקציית תגמול עבור RLHF) ואלגוריתם - כל אחד הוא חשוב להבטחת (בתקווה) ביצועים חזקים.

תרומות טכניות:

נוסחת גרדיאנט מאוחדת לשני המקרי בלב המאמר נמצאת הנוסחה של גרדיאנט מאוחד לאופטימיזציה של העדפות (נוסחה 1 במאמר)

$$\nabla_\theta = \mathbb{E}_{(q,o)\sim D}\left[\frac{1}{|o|}\sum_{t=1}^{|o|}\delta A(r,q,o,t)\nabla_\theta \log \pi_\theta(o_t|q,o_{<t})\right]$$

כאשר:

δ: מקדם גרדיאנט שתלוי באלגוריתם הספציפי, במשוב ובדאטה.
A: האלגוריתם האופטימיזציה המיושם.
r : אות משוב (feedback) המשפיע על מקדם הגרדיאנט (למשל תגמול)
θ_π: מודל מדיניות המפורמטר על ידי θ.

משוואה זו מכלילה את תהליכי האופטימיזציה המשמשים הן בשיטות מבוססות RL והן בשיטות מבוססות SFT, ומראה שההבדל העיקרי ביניהן טמון באופן שבו המשוב משולב. שיטות מבוססות RL משתמשות בדרך כלל בתגמולים סקלריים, בעוד ש-SFT משתמש בתוויות העדפה או דירוגים.

טקסונומיה של למידת העדפות:

המאמר מסווג למידת העדפות לארבעה שלבים מקושרים:
דאטה:

דגימת דאטה On-Policy: דאטה נוצרים בזמן אמת על ידי המודל המאומן. טכניקות דגימה כמו Top-K, Nucleus Sampling, ו-Monte Carlo Tree Search משמשות ליצירת דאטה מגוון ואיכותי.

איסוף נתונים Off-Policy: הנתונים נאספים מראש, לעתים קרובות ממקורות חיצוניים, כולל סטי נתונים מתויגים על ידי בני אדם (כמו למשל בשיטות HH-RLHF, SHP) או דאטהסטים סינטטיים שנוצרו על ידי LLMs (למשל, UltraChat, ULTRAFEEDBACK).

משוב:

משוב ישיר: כולל תוויות אנושיות וחוקים המנוסחים על ידי בני אדם. דוגמאות כוללות בדיקות נכונות בחשיבה מתמטית או תוצאות יוניטסטים בייצור קוד.

משוב מבוסס מודל:

מודלי תגמול: מעריכים הסתברויות העדפה אנושית באמצעות שיטות כמו מודל Bradley-Terry (נוסחה 2 במאמר):

$$p^*(y_1 \succ y_2|x) = \frac{\exp(r^*(x, y_1))}{\exp(r^*(x, y_1)) + \exp(r^*(x, y_2))}.$$

האופטימיזציה מושגת דרך פונקציית לוס סטנדרטית של לוג הנראות שלילית:

$$L_r = -\log \sigma(r^*(y_c, x) - r^*(y_r, x)).$$

מודל תגמול מבוסס מסווג בינארי (למשימות בהן איכות המקרה ניתנת לקביעה על ידי תוצאותיו):

תיוג ישיר של דגימות לאימון מסווג בינארי כמודל תגמול היאגישה פשוטה ויציבה. למשל, בחשיבה מתמטית, ניתן לתייג
דגימה על בסיס האם התשובה מניבה את התשובה הסופית הנכונה. באופן דומה, במשימות ייצור קוד, ניתן לבצע תיוג על ידי בדיקה האם הקוד שנוצר עובר בדיקות מוגדרות. בניגוד למשימות כמו סיכום טקסט או יצירת דיאלוג, הדורשות השוואות זוגיות של דוגמאות, שיטות הערכה ישירות אלו מפשטות את תהליך תיוג ההעדפות.

בניגוד למודל התגמול המסורתי של Bradley-Terry, ברגע שיש לנו התיוגים עבור הדאטה, ניתן לאמן את מודל התגמול באמצעות פונקציית לוס של סיווג בינארי תגמול עבור כל לייבל מבלי שיהיה צורך לבנות דאטה עבור זוגות.

שיטת LLM-as-a-judge: משתמש ב-LLMs עצמם להערכת פלטים. מנגנוני תגמול עצמי, מטא-תגמול (מודל שפה בונה ציון עבור ה-verdict שהוא בעצמו נותן) ועוד מגוון שיטות בסגנון.

אלגוריתמים:

האלגוריתמים מחולקים לקבוצות על פי מספר הדגימות הנדרשות לחישוב הגרדיאנט:

שיטות Point-Wise: אופטימיזציה באמצעות דגימות בודדות. דוגמאות כוללות (Proximal Policy Optimization (PPO ו-ReMax.

שיטות Pair-Wise Contrast (סוג של למידה ניגודית): מנצלות השוואות בין זוגות של דגימות: הדוגמא הבולטת של שיטה זו היא (Direct Preference Optimization (DPO.

שיטות List-Wise Contrast: משערכות את הגרדיאנט על פני כמה דגימות. גישה זו שימושית במיוחד במשימות הדורשות הערכות הוליסטיות, כמו דירוג או סיכום.

שיטות Training-Free: כוללות טכניקות אופטימיזציה של קלט/פלט, המבטלות את הצורך בעדכוני גרדיאנט במהלך היישור (המאמר לא מרחיב על זה)

אבלואציה:

אסטרטגיות אבלואציה בוחנות עד כמה טוב LLMs מתיישרים עם העדפות אנושיות.

הערכה מבוססת חוקים: משתמשת בקריטריונים מוגדרים מראש כמו נכונות עובדתית או אמות מידה ספציפיות למשימה.

הערכה מבוססת LLM: כוללת LLMs מתקדמים הפועלים כמעריכים, משתמשים בפרומפטים להערכה ודירוג תגובות.

טכניקות דגימת דאטה:

דגימה MCTS :On-Policy משפר את עושר של הדאטה איכותו למשימות הדורשות חשיבה רב-שלבית.

דגימה Off-Policy: דאטהסטים סינתטיים, המיוצרים על ידי LLMs מתקדמים, משמשים יותר ויותר כדי "לתת סקייל" ללמידת העדפות.

מסקנה:

סקירה זו מספקת מבט מתמטי קפדני ומאוחד מושגית על למידת העדפות עבור LLMs. המסגרת שלה מבהירה יחסים בין שיטות RL ו-SFT, מאפשרת לחוקרים להשוות, לשלב ולחדש אסטרטגיות יישור העדפות באופן שיטתי. הדגש על משוב, תכנון אלגוריתמים והערכה מבטיח כיסוי מקיף של התחום, הופך מאמר זה למשאב יקר ערך לקידום מחקר יישור LLM.

https://arxiv.org/abs/2409.02795

המאמר היומי של אוראל ומייק - 18.01.25
MAKING TEXT EMBEDDERS FEW-SHOT LEARNERS

היום להבדיל מהסקירות האחרונות נסקור מאמר מאוד קליל, הלא מערב מתמטיקה כבדה. המאמר מציע שיטה לבניית ייצוג (אמבדינגס) מותאם ללמידה in-context או בקצרה ל-ICL. אזכיר כי ICL היא שיטת בניית פרומפטים כאשר אנו מספקים למודל כמה דוגמאות עבור משימה שאנו מצפים ממנו שיעשה. למשל במשימת גנרוט קוד אנו מספקים למודל (בתוך הפרומפט) כמה דוגמאות שכל אחת מהן היא זוג (שאלה, קוד) במטרה "להבהיר" למודל מה אנחנו מצפים ממנו. ד"א למה ICL לפעמים עובד על המשימות שהמודל לא אומן עליהם אינו ברור ב-100% מהווה נושא מחקר די פעיל.

נציין כי המודל בנידון עדיין צריך לגנרט טקסט כלומר יש לנו מודל דקודר (עם מיסוך קוזאלי שדי מפריע לבניית האמבדינג) ונשאלת השאלה איך אנו בונים אמבדינג איתו כמו שאנו רגילים לעשות עם האנקודר. דרך אגב יצאו כמה מאמרים שהציעו שיטות לבניית אמבדינג עם מודלי דקודר כמו LLM2Vec ו-GritML אבל הם אינם מותאמים למקרה שנדון במאמר. כלומר השאלה איך אנו בונים אמבדינג של פרומפט בסגנון ICL כלומר כזה שמכיל כמה דוגמאות פתורות להדגמה.

אז המחברים מצאו לזה פתרון די פשוט. קודם כל הם הוסיפו טוקן EOS בסוף הפרומפט והתכנון הוא שייצוג הטוקן הזה יכיל את האמבדינג של הפרומפט כולו (כמו שנעשה ב-BERT לפני 7 שנים). באופן לא מפתיע המחברים בחרו לעשות זאת עם למידה ניגודית(contrastive learning או CL). מטרה של CL היא לאמן מודל ייצוג כך שהייצוגים של דוגמאות דומות(חיוביות) יהיו קרובות ואילו אלו של דוגמאות לא דומות(שליליות) יהיו רחוקים במרחק האמבדינג. בתור דוגמאות חיוביות המחברים בחרו כאלו עם תשובה נכונה על השאלה בפרומפט ואילו עבור דוגמאות שליליות מופיעות התשובה הלא נכונה. נציין כי הדוגמאות להדגמה בפרומפט נשארות זהות עבור החיוביים והשליליים.

זהו זה - ככה הם מאמנים מודל אמבדינג על מספר לא גדול של דוגמאות (few-shot) ולפי המאמר התוצאות לא רעות.

המאמר היומי של אוראל ומייק: 19.01.25
The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks

היפותזת כרטיס הלוטו (Lottery Ticket Hypothesis) אומרת שבתוך רשת נוירונים צפופה (dense neural nets) המאותחלת בצורה רנדומלית, יש תת-רשת (או "כרטיס מנצח") שמאמנים אותה בנפרד, היא יכולה להגיע לביצועים כמו של הרשת המקורית.

נמצא שטכניקת חיתוך(pruning) סטנדרטית מגלה באופן טבעי תת-רשתות כאלה, אשר עבורן מתקיים כי האתחול המחודש תחת אותם hyperparameters, משמר את התוצאות של הרשת המקורית בעלות זולה יותר,

כך שהכרטיסים המנצחים הם תת-רשתות אשר "זכו בהגרלת האתחול", ובהן המשקלים ההתחלתיים הופכים את האימון לאפקטיבי במיוחד.

הרעיון הזה מדגיש את החשיבות של המשקלים ההתחלתיים של הרשת. הכרטיסים המנצחים אינם תת-רשתות אקראיות, אלא כאלה שמתאימות במיוחד בגלל האתחול שלהן. תהליך מציאת התת-רשתות הללו אינו פשוט, כיוון שהוא כרוך בזיהוי החלקים הקריטיים(הנוירונים המשמעותיים) ברשת כבר מההתחלה.

מה זה חיתוך רשת?

חיתוך (Pruning) הוא טכניקה המסירה משקלים לא חשובים מרשת הנוירונים. לפי היפותזת כרטיס הלוטו, החיתוך עוזר לייעל את הרשת בכך שהוא מסיר נוירונים וחיבורים מיותרים, וכך יוצר רשת קלה, מהירה ויעילה יותר, ששומרת על הביצועים של הרשת המקורית ולעיתים אף משפרת אותם. החיתוך חושף את "הכרטיסים המנצחים": בתחילה, הרשת מכילה יותר מדי פרמטרים (רשת גדולה וצפופה), ואז במהלך האימון והחיתוך של המשקלים הלא משמעותיים, תת-הרשתות היעילות האלו מתגלות.

סוגי חיתוך
חיתוך לא מובנה (Unstructured Pruning): כאן אפשר להסיר כל משקל או קבוצה של משקלים, ללא מגבלות. זה יוצר רשת נוירונים "דלילה" שבה רק חלק מהמשקלים נשארים. טכניקה זו נקראת גם חיתוך משקלים (Weight Pruning). בחיתוך שכזה, אין בחירה מוגדרת מראש מה ייחתך, הכל לפי הבחירה הפחותה ביותר של התרומה של אותו נוירון שנבחר להיחתך.

חיתוך מובנה (Structured Pruning): כאן מסירים קבוצות שלמות של משקלים, כמו נוירונים שלמים ברשת קדמית (FFN). התוצאה היא רשת נוירונים "צפופה" אך קטנה יותר. הבחירה כאן היא מושכלת, בה המבניות של הרשת חשובה להישמר, יכול להיות שיהיה נוירון שלא יבחר להיחתך על מנת לא לפגוע במבניות שנבחרה, לעומת נוירונים אחרים.

חיתוך בבת אחת מול חיתוך איטרטיבי
חיתוך בבת אחת (One-shot Pruning): מאמנים את הרשת פעם אחת, חותכים אחוז מסוים מהמשקלים (p%), ואז מאתחלים מחדש את המשקלים שנשארו. מדובר בהנחה כי באיטרציה אחת הגענו לפתרון הסופי והמיוחל, ללא צורך בתהליך חוזר ומתמשך.

חיתוך איטרטיבי (Iterative Pruning): מאמנים את הרשת, חותכים חלק מהמשקלים, מאתחלים מחדש, וחוזרים על התהליך כמה פעמים. בכל סיבוב חותכים אחוז קטן מהמשקלים ששרדו מהסיבוב הקודם. תוצאות מראות שחיתוך איטרטיבי מצליח למצוא כרטיסים מנצחים שמגיעים לאותם ביצועים כמו של הרשת המקורית, תוך שימוש ברשת קטנה יותר בהשוואה לחיתוך בבת אחת.

https://arxiv.org/pdf/1803.03635

המאמר היומי של מייק - 21.01.25
Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts

המאמר משך את תשומת ליבי למרות הידע הרדוד שאני מחזיק לגבי תחום הסדרות העתיות (time-series). בגדול הסיבה העיקרית לכך שבשמו מופיע צמד מילים "Foundational Models" שזה חיה די נדירה בתחום הסדרות העתיות להבדיל מתחום מודלי שפה. הסיבה לכך (כנראה) היא מגוון עשיר הרבה יותר של סדרות עתיות השונות יחסית לשפה טבעית.

האמת לא מצאתי ב- Time-MoE, המבוססת כמובן על הטרנספורמרים, מציאות ארכיטקטוניות מאוד מעניינות ועם זאת יש בו כמה דברים שונים מאלו שאנו רגילים לראות ב-LLMs. למשל במקום שכבת טוקניזציה ואמבדינג, מבוססים על מילון טוקנים, שיש לנו ב-LLMs במודל המוצע יש כל טוקן (שזו נקודה בסדרה) עובר טרנספורמציה לא ליניאריות עם אקטיבציה מסוג SwiGLU וכמה טרנספורמציות ליניאריות.

בנוגע לשכבת הטרנספורמרים, המחברים לוקחים ארכיטקטורת MoE די סטנדרטית. השוני היחיד שמשך את עיניי הוא שימוש בשיטת נרמול RMSNorm שלא הכרתי. פרט לכך יש את כל השכבות הרגילות של הטרנספורמרים כולל כמובן שכבות residual.

השכבה האחרונה של Time-MoE היא קצת שונה ממה שאנו רגילים לראות בטרנספורמרים. מכיוון שלהבדיל ממודלי שפה אנו צריכים מודל בעולם של TS אנו צריכים לחזות במספר נקודות זמן שונה (נגיד שניה, דקה או יום קדימה), המחברים משתמשים בכמה ראשים בשכבה האחרונה. כל ראש אחראי על חיזוי באופק מסוים (כמות דגימות קדימה). באימון משלבים את הלוסים מכל הראשים.

גם פונקציות לוס במאמר הן די סטנדרטיות: פונקצית הובר שהיא הגרסה הרובסטית של L2 (הלא נותנת לא להגיע לערכים גבוהים מאוד). בנוסף יש איבר רגולריזציה שמנסה להפעיל את כל המומחים ב-MoE בצורה אחידה. וכמובן אימנו את המודל על דאטהסטים ענקיים ומגוונים.

זהו וזה - סקירה קצרה, ובתקווה גם ברורה....

https://arxiv.org/pdf/2409.16040

המאמר היומי של מייק - 22.01.25

MONOFORMER: ONE TRANSFORMER FOR BOTH DIFFUSION AND AUTOREGRESSION

היום נעשה סקירה קצרה של מאמר די מעניין ששילב שני סוגים של מודלים, מודל שפה ומודל ויז'ן בטרנספורמר אחד. רוב המודלים מולטימודליים מורכבים מכמה מודלים שכל אחד מהם אחראי על הגנרוט של סוג דאטה אחד. למשל מודלי שפה ויזואליים בד"כ מורכבים משני מודלים: מודל שפה ומודל לגנרוט תמונות. המחברים מציעים "לחבר" את שני המודלים האלה למודל טרנספורמר אחד וזה נעשה בצורה די אינטואיטיבית.

קודם כל נציין כי שני המודלים האלו עובדים במרחב הטוקנים כאשר עבור מודלי שפה כל טוקן הוא חלק של מילה או מילה שלמה ואילו עבור מודל ויזואלי כל טוקן הוא פאץ' של תמונה. אז הניסיון לחבר אותם למודל אחד נראה די טבעי אך לא ברור האם ניתן לאמן אותו הטרנספורמר לגנרט שפה ותמונות כאחד.

המודל המוצע מגנרט שפה בדיוק כמו LLM רגיל, בצורה אוטורגרסיבית, כלומר, טוקן אחרי טוקן. אבל איך ניתן לשלב אותו עם מודל לגנרוט תמונות שכמובן מבוסס על מודלי דיפוזיה (בשנת 2025 זה האופציה הדיפולטית הרי). קודם כל צריך לזכור שמודל אוטורגרסיבי (לגנרוט שפה) עובד בצורה סיבתית (קוזלית), כלומר במהלך גנרוט טוקן n כל הטוקנים מאחוריו ממוסכים ולא משתתפים בגנרוט(משתמשים במסכה קוזלית). למודלי אנו צריכים מודל דו כיווני כי בזמן גנרוט פאץ' של תמונה כדאי מאוד להשתמש בכל הפאצ'ים האחרים.

בדיוק כך בנוי המודל המוצע - השפה מגונרטת עם מסכה קוזלית והתמונה מגונרטת עם כל הטוקנים (כולל הטוקנים של טקסט). דרך אגב הגישה הזו תעבוד גם לכיוון השני: כלומר בגנרוט של טקסט מתמונה (למשל למשימת captioning). אבל איך נדע לעבור ממצב "קוזלי" למצב "דו-כיווני". המחברים מציעים להשתמש בטוקן מסוים המסמן שממנו מתחיל גנרוט התמונה - הטוקן הזה אמור להיות מג'ונרט למשל למשימה יצירת תמונה מטקסט.

כמה מילים על הטרנספורמר לגנרוט תמונה. המאמר משתמש במודל דיפוזיה לטנטי כאשר המודל מאומן לבנות ייצוג לטנטי של תמונה מרעש (עבור כל פאץ). לאחר מכן כל הייצוגים (של הפאצ'ים) מועברים דרך הדקודר (מבוסס VAE) שבונה ממנו תמונה.

המודל מאומן עם הלוס שהוא סכום משוקלל של הלוסים הסטנדרטיים עבור המודלים המוזכרים: מודל שפה ומודל דיפוזיה. המאמר מצליח לגנרט תמונות די יפות....

https://arxiv.org/abs/2409.16280

המאמר היומי של מייק - 24.01.25
Back to Basics: Revisiting REINFORCE Style Optimization for Learning from Human Feedback in LLMs

תמצית המאמר:

המאמר בוחן מחדש את השימוש בלמידה מחיזוקים מפידבק אנושי (RLHF) באופטימיזצית LLMs. הוא מאתגר את הדומיננטיות של PPO (Proximal Policy Optimization) כשיטת למידת החיזוקים הסטנדרטית בהקשר זה, תוך הדגשת חוסר היעילות החישובית והמורכבות המיותרת שלו. במקום זאת, החוקרים מציעים לחזור לשיטות פשוטות יותר בסגנון REINFORCE, ספציפית (REINFORCE Vanilla Policy Gradient) והרחבתו הרב-דגימתית, (RLOO REINFORCE Leave-One-Out). שיטות אלו מוכיחות ביצועים טובים יותר מ-PPO מבחינת עלות חישובית, יעילות דגימה אופטימיזצית תגמול במספר מערכי נתונים וארכיטקטורות LLM. הממצאים מדגישים שניתן להשיג התאמה של LLMs להעדפות אנושיות עם אסטרטגיות אופטימיזציה פשוטות יותר המותאמות לייחודיות של RLHF.

הרחבה על נקודות עיקריות:

1. **פישוט תיאורטי:**

החוקרים מראים שרבים מהרכיבים של PPO (למשל, קליפינג, פונקציות ערך (value), ומידול ברמת טוקנים) אינם הכרחיים ל-RLHF, בהינתן האתחול טוב של LLM (לאחר SFT למשל). על ידי מידול סדרות שלמוץ כפעולות בודדות, REINFORCE נמנע מהמורכבות של פונקציות ערך-מצב(V ו-Q) ברמת טוקן, והופך את הבעיה לדומה יותר לבנדיט הקשרי.

2. **יעילות מעשית:**

שיטת RLOO משתמשת בכל הדגימות שנוצרו לבניית בסיס השוואה, משיג יעילות דגימה גבוהה יותר מ-RAFT, שלוקח את רק את הדגימות בעלות ציונים גבוהים (סוג של rejection sampling). זה מוביל לחיסכון משמעותי בחישובים וניצול טוב יותר של הנתונים הזמינים. הגישה מפשטת את תהליכי ה-RLHF על ידי הפחתת התלות בהיפר-פרמטרים רגישים כמו יחסי קליפינג והפרמטרים בשערוך פונקציית יתרון (כמו ב-GAE).

4.**רובסטיות:**
שיטת RLOO מדגימה רובסטיות לתגמולים רועשים ועונשי KL גבוהים יותר, עולה על שיטות כמו RAFT שרגישות יותר לדיוקם.

תובנות תיאורטיות:

1. **איזון שונות-הטיה(bias-variance tradeoff) ושערוך גרדיאנט ללא הטיה**:

שיטת PPO מסתמכת על פונקציות ערך-מצב ושערוך יתרון מוכלל (Generalized Advantage Estimation) להפחתת שונות של שערוך גרדיאנט במחיר של העלאת הטיה. המאמר טוען שב-RLHF עבור LLMs די מאומן (warm start) הופך את הפחתת השונות לפחות קריטית. זה מאפשר לשיטות ללא הטיה כמו REINFORCE לתפקד היטב בלי להכניס הטיה משמעותית. אמפירית, המאמר מדגים ש-REINFORCE משיג אופטימיזציית תגמול טובה יותר מ-PPO, אפילו תחת תנאים של שונות גבוהה תיאורטית.

2. **מידול מסלול מלא(תשובה שלמה) לעומת מידול ברמת טוקנים**:

שיטת PPO ממדלת כל טוקן כפעולה, יוצר תהליך החלטה מרקובי (MDP) בו רצפי טוקנים חלקיים הם מצבים. עם זאת, RLHF מייחס תגמולים רק לתשובות שלמות, מה שהופך מצבי ביניים ללא רלוונטיים. על ידי מידול התשובה כפעולה יחידה, REINFORCE מפשט את הבעיה למבנה Contextual Bandit המתיישר ישירות עם מבנה התגמול. תוצאות אמפיריות מאשרות כי גישה זו עולה על מידול ברמת טוקנים הן ביעילות והן בביצועים.

3. **קליפינג ויציבות עדכוני מדיניות:**

שיטת PPO משתמש במנגנון קליפינג למניעת עדכוני פוליסי גדולים שעלולים לערער את הלמידה. החוקרים מראים שזה מיותר עבור RLHF, מכיוון שמשטח האופטימיזציה יציב הודות ל warm-started LLM. הסרת הקליפינג ב-PPO או הימנעות ממנו לחלוטין עם REINFORCE מובילה לביצועים טובים יותר, מה שמצביע על כך ש-RLHF אינו דורש רמה כזו של ייצוב.

4. **איזון בין הפחתת שונות והעלאה קלה בהטיה:**

אומדן היתרון ב- PPO מאזן בין שונות והטיה, נשלט על ידי ההיפר-פרמטר λ. ערכי λ גבוהים יותר (קרובים ל-1) מפחיתים הטיה אך מגדילים שונות. החוקרים מדגימים שב-RLHF, ערכי λ גבוהים יותר מובילים באופן עקבי לתגמולים טובים יותר של המודל, תומכים בשימוש באומדנים ללא הטיה כמו REINFORCE

מגבלות וכיוונים עתידיים

1. **אופטימיזציית יתר של תגמול:**

המחקר אינו מתמודד עם אופטימיזציית יתר של מודל התגמול(reward hacking), בה המדיניות מנצלת הטיות בפונקציית התגמול על חשבון הכללה. זה נשאר אתגר פתוח עבור RLHF.

2. **הערכה אנושית:**

בעוד שאחוזי ניצחון מדומים באמצעות GPT-4 משמשים כמדד להעדפות אנושיות, הערכות אנושיות ישירות היו מספקות ראיות חזקות יותר לאיכות ההתאמה.

3. "**סקלביליות":**

הסקלביליות של REINFORCE ו-RLOO למודלים(הם בדקו רק מודלים של 7B) ודאטהסטים גדולים יותר מצריכה מחקר נוסף.

מסקנה

המאמר מציג טיעון משכנע לבחינה מחודשת של שיטות בסגנון REINFORCE ב-RLHF, מאתגר את הדומיננטיות של PPO ודומיה. על ידי ניצול המאפיינים הספציפיים של RLHF - כמו warm started LLM ותגמולים ברמת הסדרה - החוקרים מדגימים ששיטות פשוטות יותר כמו REINFORCE ו-RLOO יכולות לעלות על חלופות מורכבות יותר כמו PPO ו-RAFT מבחינת אופטימיזציית תגמול, יעילות דגימה ועמידות.

https://arxiv.org/abs/2402.14740

המאמר היומי של מייק - 27.01.25
FineZip : Pushing the Limits of Large Language Models for Practical Lossless Text Compression

בחרתי את המאמר הזה לסקירה כי יש לי חיבה גדולה לכל מה שקשור לדחיסה - דחיסה של מודלים, דחיסה של דאטה או כל דחיסה שהיא :). המאמר מציע שיטה נחמדה לדחוס דאטה. אתם בטח יודעים שהמודלים שלנו יודעים לדחוס דאטה בצורה לא רעה עם הייצוג הלטנטי (אמבדינג) שהם מפיקים מהדאטה. אם אני לא טועה אנו יכולים לדחוס תמונה ברזולוציה גבוהה פי 100 עם האמבדינג שלו. אבל הדחיסה הזו היא לא lossless. אכן ניתן לשחזר את התמונה מהאמבדינג שלה כך שהעין האנושית לא תבחין שום הבדל בין התמונה המשוחזרת לבין המקורית, אבל הן לא בהכרח ייצאו זהות. במקרה של טקסט זה יכול להיות קצת בעייתי כי אנו רוצים לשחזרו כמו שהוא.

המאמר המסוקר לעמות זאת מציע שיטה לדחיסה טקסט כך שניתן יהיה לשחזרו במדויק. השיטה המוצעת היא די פשוטה ואינטואיטיבית. הרי איך מודל שפה מגנרט טקסט - השכבה האחרונה שלו פולטת התפלגות מעל מרחב הטוקנים והטוקן נדגם מההתפלגות הזו (יש כמה שיטות). אנו יכולים להיעזר בהתפלגות זו כדי לדחוס את הטקסט שלנו. למשל כמו שהוצע במאמר LLMZip אני יכולים לקודד כל טוקן (בהנתן הקשר לפניו) על יד ראנק של

ההסתברות שלו בהתפלגות עבור הטוקן הזה. ראנק זה בעצם המיקום של הטוקן ברשימת הטוקנים הממוינת (בסדר יורד) לפי הסתברות שלו בהתפלגות הטוקן הזה.

אם מודל השפה שאנו משתמשים בו הוא מאוד חזק ראנק זה יהיה קרוב ל- 1(או 2, 3 אבל לא 1000). וידוע כי סדרות כאלו ניתן לדחוס בצורה מאוד יעילה (קצב דחיסה גבוה). אז LLMZip הציע לטייב מודל שפה לטקסט שדוחסים אותו שזה לא פרקטי כי לכל טקסט צריך לשמור מודל משלו וגם האימון כבד. האממר המסוקר מציע להשתמש ב-LoRA (או PEFT אחר) לדחיסה כך שנצטרך לשמור גם את מטריצות התוספות (או adapters) לכל טקסט עם מודל שפה אחד לכולם.

עדיין זה לא מאוד פרקטי אבל מבחינה רעיונית די מעניין...

https://arxiv.org/abs/2409.17141

המאמר היומי של מייק - 29.01.25
A Survey on Diffusion Models for Inverse Problems

מודלי דיפוזיה התפתחו במהירות ככלי חזק המסוגל לייצר דאטה באיכות גבוהה במגוון תחומים. הצלחתם סללה את הדרך להתקדמות פורצת דרך בפתרון בעיות הפוכות(inverse problems), במיוחד בשחזור וחידוש תמונות, שם מודלי דיפוזיה מאומנים משמשים כפריורים (כלומר מסוגל בצורה לא מפורשת להבין האם התמונה המשוחזרת בא מההתפלגות האמיתית).

מאמר זה מציע חקירה מקיפה של שיטות המנצלות מודלי דיפוזיה מאומנים מראש כדי לטפל בבעיות הפוכות ללא צורך באימון נוסף. הם מציגים טקסונומיה מובנית המסווגת גישות אלה על בסיס הבעיות הספציפיות שהן מטפלות בהן והטכניקות שהן מעסיקות.

בגדול כל השיטות האלה ממנפות גישה דיפוזיונית גנרטיביות לשחזור דאטה מורעש.

מסגרת מתמטית של מודלי דיפוזיה גנרטיביים:

המאמר מפרמל בעיות הפוכות תחת הניסוח הכללי:

$$Y = A(X) + \sigma_y Z, \;\; Z \sim N(0, I_m)$$

כאשר A הוא אופרטור או פונקציית שיבוש (יכול לא ליניארי), ו- Z הוא רעש גאוסי. בעיות הפוכות שונות כמו הסרת רעש, השלמת תמונה סופר-רזולוציה,ממוסגרים בתוך ניסוח זה על ידי הגדרת צורות שונות של A.

המאמר דן במודלי דיפוזיה הסתברותיים להסרת רעש (DDPMs) והרחבותיהם המבוססות על משוואות דיפרנציאליות סטוכסטיות (SDEs) כדי לגשת לבעיות הפוכות. התהליך הקדמי מתואר על ידי:

$$\mathrm{d}X_t = f(X_t, t)\mathrm{d}t + g(t)\mathrm{d}W_t$$

כאשר W_t הוא תהליך וינר, X_t הוא התפלגות הדאטה בזמן t. כאן f ו-g הם היפר-פרמטרים של תהליך הדיפוזיה (noise schedule). מסגרת משוואות דיפרנציאליות סטוכסטיות(SDE) הפוכות (כי מתחילים מהרעש ומסירים אותו לאט לאט) של אנדרסון משמשת לדגימה מהתפלגות הנתונים הלא ידועה:

$$\mathrm{d}X_t = \big(f(X_t,t) - g^2(t)\nabla_{X_t}\log p_t(X_t)\big)\mathrm{d}t + g(t)$$

ניסוח זה מאפשר מידול דאטה מורעש על ידי הוספה הדרגתית של רעש ולאחר מכן היפוך תהליך הדיפוזיה לשחזור דאטה. האתגר המתמטי העיקרי הוא שערוך של פונקציית הציון(score function) שהיא הגרדיאנט של התפלגות (p_t(X_t. הסקר מדגיש את תפקידה המרכזי של נוסחת טווידי:

$$\nabla_{x_t}\log p_t(x_t) = \frac{\mathbb{E}[X_0|X_t = x_t] - x_t}{\sigma_t^2}$$

למידת התוחלת המותנית באמצעות רשתות נוירונים מספקת דרך יעילה לקרב את הציון.

טקסונומיה של שיטות בפתרון בעיות הפוכות מבוססות דיפוזיה

מחברי המאמר מספקים טקסונומיה עשירה המסווגת שיטות על בסיס הגישה המתמטית שלהן, סוגי בעיות היעד וטכניקות אופטימיזציה. בגו

שערוך score function באמצעות קירובים ליניאריים לבעיות הפוכות ליניאריים (בקירוב)

קירובים אלה(score function-ל) מנצלים לעתים קרובות פתרונות בצורה סגורה לבעיות הפוכות ליניאריות. הצורה הכללית ניתנת על ידי (y כאן הוא הדאטה המשובש)

$$\nabla_{x_t}\log p(y|x_t) \approx -L_t M_t G_t^{-1}\nabla x_t$$

כאשר: L מייצג את שגיאת המדידה. M הטלת השגיאה בחזרה למרחב הפתרון. G גורם re-scaling השולט בעוצמה התחשבות ב-y (התמונה המשובשת)

שיטות מייצגות:

שיטת (ALD) Score-ALD כאשר ALD הוא Annealed Langevin Dynamics משתמש בקירוב הבא:

$$\nabla_{x_t}\log p(y|x_t) \approx -\frac{A^T(y - Ax_t)}{\sigma_y^2 + \gamma_t^2}$$

שיטת DPS (דגימת פוסטריור דיפוזיה): מקרב את הפוסטריור y (הדאטה המשובש) באמצעות מיפוי (X_t היא הגרסה המורעשת של התמונה המשוחזרת):

$$p(y|X_0 = \mathbb{E}[X_0|X_t]) \sim \mathcal{N}(y; A\mathbb{E}[X_0|X_t], \sigma_y^2 I)$$

המוביל לאומדן הבא עבור ה-score function:

$$\nabla_{x_t} \log p(y|x_t) \propto A^T(y - A\mathbb{E}[X_0|X_t])$$

התאמת מומנטים: מרחיבה את DPS על ידי שילוב קירוב גאוסיאני אנאיזוטרופי (לא איזוטרופי):

$$p(x_0|x_t) \approx \mathcal{N}(\mathbb{E}[X_0|X_t], \sigma_t^2 \nabla_{x_t}\mathbb{E}[X_0|X_t])$$

4.2 **שיטות הסקה וריאציונית**

שיטות אלה מקרבות את התפלגות הפוסטריור האמיתית על ידי הצגת התפלגות תחליפית(וריאציונית) נוחה לטיפול ואופטימיזציה של הפרמטרים שלה באמצעות טכניקות וריאציוניות. המטרה היא למזער את מרחק KL בין הקירוב והפוסטריור האמיתי:

$$\min_q D_{KL}(q(x)\|p(x|y))$$

שיטת RED-Diff מציעה אובדן חדשני המשלב לוס שחזור והתאמת ציון (ככה תרגמתי score matching, שיטה ידועה לגנרוט דאטה) במודלי דיפוזיה:

$$L_{\text{RED}}(\mu) = \frac{1}{2\sigma_y^2}\|y - A\mu\|^2 + \sum_t \lambda_t \|\epsilon_\theta(x_t) - \epsilon\|^2$$

כאשר μ הוא הממוצע של האומדן הוריאציוני, ו-ε_θ הוא פונקציית denoising (שערוך רעש) שנלמדה על ידי מודל הדיפוזיה.

Blind RED-Diff: מרחיב את RED-Diff על ידי אופטימיזציה משותפת של הייצוג הלטנטי של התמונה ופרמטרי המודל φ. זה מוביל לבעיה וריאציונית הבאה:

$$\min_q D_{KL}(q(x,\phi)\|p(x,\phi|y))$$

כאן אנו מאפטמים את המודל הלטנטי לתמונה יחד עם מודל דיפוזיה המשחזר אותו.

4.3 שיטות מסוג CSGM (מודלים גנרטיביים מבוססי ציון מותנה - conditional score).

גישות אלה מבצעות אופטימיזציה ישירות על פני מרחב לטנטי באמצעות backprop. הרעיון הבסיסי הוא להתאים באופן איטרטיבי וקטורי רעש התחלתיים כדי לספק אילוצי מדידה (של התמונה המורעשת כלומר).

טכניקות מרכזיות:

- בקפרופ (backprop) דרך שימוש דוגם דיפוזיה דטרמיניסטי.
- אופטימיזציית מרחב לטנטי לאכיפת נאמנות למדידות הנצפות (המח.

4.4 שיטות מדויקות אסימפטוטית(asymptotically exact).

שיטות אלה מסתמכות על דגימה מהתפלגות הפוסטריור האמיתית באמצעות טכניקות מתקדמות של שרשרת מרקוב מונטה קרלו (MCMC).

טכניקות מרכזיות:

- התפשטות חלקיקים(particle propagation): שיטות מונטה קרלו רציפות (SMC) מפיצות חלקיקים מרובים דרך התפלגויות כדי לקרב את הפוסטריור.
- דגימה מפותלת (twisted sampling): שיטות כמו דוגם הדיפוזיה twisted משתמשות בעדכונים מודעי גיאומטריה (של תמונות או דאטה אחר) כדי לשפר את קצבי ההתכנסות.

4.5 **טכניקות אופטימיזציה**

השיטות משתנות עוד יותר לפי אסטרטגיות האופטימיזציה המועסקות:

- טכניקות מבוססות גרדיאנט: משתמשות בנגזרות לאכיפת עקביות מדידה.
- טכניקות מבוססות הטלה: מטילות דגימות על תת-מרחבים אפשריים.
- טכניקות דגימה סטוכסטיות: משתמשות בגישות הסתברותיות כמו דינמיקת לנג'בין לעדכוני חלקיקים (כמו בSMC).

המאמר היומי של מייק

סקירה זו זה מאגדת באלגנטיות כלים מתמטיים מתקדמים, ומספק בסיס מוצק לחוקרים השואפים לפתור בעיות הפוכות באמצעות תהליכי דיפוזיה. השילוב של חשבון סטוכסטי, הסקה בייסיאנית וטכניקות אופטימיזציה הופך אותו לנקודת התייחסות קריטית לדחיפת גבולות פתרון הבעיות ההפוכות.

https://arxiv.org/pdf/2410.00083

המאמר היומי של מייק - 31.01.25
Law of the Weakest Link: Cross Capabilities of Large Language Models

מבוא והגדרת הבעיה:

המחברים מדגישים פער קריטי במחקר ה-LLM הקיים - הנטייה להתמקד בהערכת יכולות מבודדות תוך התעלמות ממשימות מהעולם האמיתי הדורשות מיומנויות מרובות(aka AGI :)), המכונות יכולות צולבות (cross-cap). המאמר ממסגר בעיה זו באמצעות טקסונומיה מקיפה של 7 יכולות בודדות ושבע יכולות צולבות, כגון קידוד וחשיבה ושימוש בכלים וקידוד. כדי להתמודד עם המורכבות הטבועה בהערכת הצמתים הללו, המחברים מציעים את CrossEval, מדד המורכב מ-1,400 הנחיות מתויגות על ידי בני אדם המיועדות לבדוק את ביצועי ה-LLM במשימות רב-ממדיות.

דוגמאות ליכולות צולבות:

קידוד וחשיבה: פרומפט בקטגוריה זו עשוי לבקש מהמודל לנתח קטע קוד ולקבוע אם הוא מיישם נכון פונקציה מתמטית מורכבת. משימה זו דורשת לא רק ידע בקידוד אלא גם חשיבה לוגית כדי לאמת את נכונות הפונקציה.

שימוש בכלים וחשיבה: בדוגמה אחרת, הנחיה עשויה לדרוש מהמודל להשתמש בכלי אחזור מידע מבוססי אינטרנט כדי לענות על שאלה לגבי מגמות מזג אוויר היסטוריות, ולאחר מכן לספק הסבר אנליטי שלב-אחר-שלב של הדפוסים הנצפים. משימה זו דורשת הן יכולות חשיבה והן שימוש בכלים חיצוניים.

מתודולוגיה:

הגדרות יכולת מקיפות: הם בונים טקסונומיה מפורטת של יכולות בודדות וצולבות, המסווגת משימות לקטגוריות רחבות ותתי-קטגוריות מדויקות.
מדד CrossEval: מסגרת הערכה חדשנית זו מורכבת מ-1,400 הנחיות, 4,200 תגובות מודל, ו-8,400 דירוגים אנושיים. מערך ההנחיות כולל משימות ברמות קושי שונות, החל משאלות עובדתיות פשוטות ועד למשימות מורכבות הדורשות יכולות צולבות.

הערכה מבוססת LLM: המחקר מציג מסגרת הערכה מרובת-התייחסויות שבה מעריכים מומחים מעריכים את איכות התגובות המרובות של המודל בסולם ליקרט. המחברים גם מפתחים אסטרטגיית הערכה מבוססת הפחתת נקודות לדיוק משופר.

ניתוח דינמיקת יכולות צולבות: המחברים מוצאים שביצועי יכולות צולבות לעתים קרובות מצייתים ל"חוק החוליה החלשה ביותר" — שבו הביצועים מוגבלים על ידי היכולת האינדיבידואלית החלשה ביותר.

ממצאים ניסיוניים:

הממצאים חושפים מספר תובנות מפתח המדגישות את המגבלות והחוזקות של ה-LLM הנוכחיים כאשר הם מתמודדים עם פונקציות יכולת צולבות.

חוק החוליה החלשה ביותר:

התצפית הבולטת ביותר היא שביצועי היכולות הצולבות מוגבלים על ידי היכולת האינדיבידואלית החלשה ביותר, בהתאם ל"חוק החוליה החלשה ביותר". מתוך 58 תרחישי יכולת צולבת שנבדקו ב-17 מודלי LLM, 38 הראו ביצועים נמוכים יותר מכל אחת מהיכולות האינדיבידואליות המעורבות, בעוד ש-20 ציונים נמצאו בין היכולות החזקות והחלשות אך היו קרובים הרבה יותר לחלשה יותר. למשל, במשימות המשלבות שימוש בכלים וחשיבה, אם המודל הציג כישורי חשיבה חלשים, זה פגע משמעותית בביצועים גם כאשר יכולת המודל להשתמש בכלים הייתה מיומנת. אפקט זה נצפה ללא קשר למורכבות או לאופי המשימה.

אפקט "חוק החוליה החלשה ביותר" נשמר ללא קשר לאיזה מעריך מבוסס LLM שימש. בין אם GPT-4o או Claude 3.5 Sonnet שימשו כשופטים, התוצאות באופן עקבי התקבצו ליד היכולת האינדיבידואלית החלשה יותר. עקביות זו מחזקת את חוסנם של ממצאי המדד ומרמזת שהמגבלות הנוכחיות של LLM הן מבניות עמוקות ולא ספציפיות למתודולוגיות הערכה.

חסרונות בשימוש בכלים:

שימוש בכלים התגלה כיכולת החלשה ביותר בכל ה-LLM שנבדקו. משימות הדורשות גלישה באינטרנט, אחזור נתונים דינמי, או הרצת קוד חיצוני הוכחו כמאתגרות במיוחד. הציונים הגבוהים ביותר למשימות הכוללות שימוש

בכלים מעולם לא עלו על 50 בסולם של 1-100 לאורך המדד. באופן בולט, אפילו מודלים עם פונקציונליות מפרש קוד, כמו Gemini Pro Exp, התקשו לשמור על ביצועים שווים למשימות חשיבה פשוטות יותר.

חולשה זו קריטית מכיוון ששימוש בכלים הוא יסודי ליישומים רבים בעולם האמיתי, כגון סיוע במחקר, ניתוח נתונים, וסוכני AI. המחברים מדגישים שמודלים המסתמכים אך ורק על מקורות נתונים סטטיים ביצעו באופן גרוע בהשוואה למשימות שבהן מידע מפורש יותר היה זמין ישירות בתוך ההנחיה.

פער ביצועים ביכולות צולבות:

בממוצע, מודלים השיגו 65.72 למשימות יכולת בודדות אך רק 58.67 למשימות יכולת צולבות, פער של 7.05 נקודות. זה מדגיש את הקושי שמודלים נתקלים בו בעת שילוב מיומנויות מרובות. משימות "תרגום מספרדית וחשיבה" ו"הקשר ארוך (long context) וקידוד" הדגימו פערים גדולים במיוחד, המרמזים שנדרש אופטימיזציה נוספת בתרחישי עיבוד רב-לשוני והקשר ארוך.

יעילות CrossEval בהבחנה:

CrossEval הוכח כיעיל בהבחנה בין הבדלים עדינים אפילו בין LLM מתקדמים ביותר. למשל, מודל Claude 3.5 Sonnet עקב בעקביות על קודמיו (המודלים הקודמים של אנטרופיק) במשימות הכוללות זיהוי תמונות וחשיבה וספרדית וזיהוי תמונות. התקדמות זו משקפת את ההתפתחות של מודלי Claude מתוחכמים יותר ומדגישה את הערך של CrossEval במדידת השיפורים העדינים ביכולות LLM.

שיפור מדדי קורלציה:

המדד הדגים שיפור במדדי קורלציה להערכות מבוססות LLM במקרה שמספקים ל-LLM המבצע אבלואציה דוגמאות מתויגות. קורלציית פירסון השתפרה מ-0.578 ללא דוגמאות מתויגות ל-0.697 עם שתי דוגמאות, המצביע על כך שהכללת התייחסויות מתויגות היטב שיפרה משמעותית את אמינות ההערכה.

סיכום:

הניסויים מגלים שבעוד ש-LLM משתפרים במהירות, הם נשארים מוגבלים מאוד על ידי הרכיבים החלשים ביותר שלהם. טיפול במגבלות אלו חיוני להשגת מערכות AI חסונות יותר, רב-תפקודיות המסוגלות לפתור בעיות מורכבות מהעולם האמיתי.

https://arxiv.org/abs/2409.19951

המאמר היומי של מייק - 01.02.25

Classical Statistical (In-Sample) Intuitions Don't GeneralizeWell: A Note on Bias-Variance Tradeoffs, Overfitting and Moving from Fixed to Random Designs

מבוא:

שיטות ML מודרניות מציגות התנהגויות שסותרות באופן בולט אינטואיציות סטטיסטיות מסורתיות, במיוחד בנוגע לאימון-יתר (over-training), לאיזון בין הטיה לשונות, וליכולת הכללה. הסטטיסטיקה הקלאסית טוענת לעתים קרובות שככל שמורכבות המודל עולה, ההטיה יורדת, אך השונות עולה - איזון ידוע בין הטיה לשונות. עם זאת, תופעות כמו Double Descent או DD בקצרה ו- benign overfitting מאתגרות השקפה זו. המאמר המסוקרה טוען שתופעות אלה אינן נובעות באופן בלעדי ממודלים מורכבים, פרמטריזציית-יתר, או דאטה רבי-ממד, אלא

דווקא ממעבר יסודי בין שני סוגי הבעיה הסטטיסטית: fixed and random design. המאמר מספק חקירה מתמטית של האופן שבו מעבר זה משנה באופן משמעותי עקרונות סטטיסטיים.

הגדרת הבעיה: משטרי D_f vs fixed design D_r - random design

ההבחנה בין D_f ל- D_r היא התובנה המהותית של המאמר:

משטר D_f: הנקודות בטסט סט נותרות זהות לאלו שבאימון, כאשר רק התוויות שלהן נדגמות מחדש. ניתוח סטטיסטי קלאסי מניח את זה לעתים קרובות ועבורו אנו מנסים למזער את שגיאת השערוך *in-sample*.

משטר D_r: גם הנקודות וגם התוויות במהלך הבדיקה נדגמים באופן בלתי תלוי מהתפלגות הדאה. משטר זה מתיישר עם האופן שבו מודלי ML משוערכים כיום, תוך התמקדות בשגיאת הכללה או שגיאת חיזוי מחוץ למדגם (out-of-distribution).

המעבר D_f ל-D_r גורם לשינויים עמוקים בהתנהגות של הטיה, שונות, ושגיאת החיזוי הכוללת. שינוי עדין אך משפיע זה הוא הסיבה המרכזית לכך שתופעות ML מודרניות נראות כמפרות את האינטואיציה הסטטיסטית הקלאסית.

מתמטית, השגיאות בשני המשטרים מוגדרות כך. שגיאת D_f (שהיא in-sample) כאשר הן תוצאות שנדגמו מחדש בקלטים קבועים.

$$\mathrm{ERR}_{\mathrm{fixed}} = \mathbb{E}_{\tilde{y}}\left[\frac{1}{n}\sum_{i=1}^{n}(\tilde{y}_i - \hat{f}(x_i))^2\right]$$

כאשר y_i הפלטים שנדגמו מחדש עבור הפלטים מהטריין סט. שגיאת D_r (מחוץ למדגם או out-of-distribution) מוגדרת באופן הבא:

$$\mathrm{ERR}_{\mathrm{random}} = \mathbb{E}_{x_0,y_0}\left[(y_0 - \hat{f}(x_0))^2\right]$$

כאשר גם x_0 וגם y_0 הם דגימות חדשות מהתפלגות הדאטה. שינוי זה מוביל להשלכות מרחיקות לכת עבור איזון ההטיה-שונות ותכונות ההכללה של מודלים. הטיה ושונות ב-D_f מקבל צורה שמוכרת לנו היטב:

$$\mathrm{MSE}(x) = \mathrm{Bias}^2(x) + \mathrm{Var}(x) + \sigma^2$$

$$\mathrm{Bias}(x) = f^*(x) - \mathbb{E}[\hat{f}(x)], \quad \mathrm{Var}(x) = \mathbb{E}[(\hat{f}(x) - \mathbb{E}[\hat{f}(x)])^2]$$

כאשר σ^2 הינו הרעש שלא ניתן לצמצום, f*(x) היא הפונקציה ground-truth הנלמדת ואילו f^(x) הוא המשערך. עבור אומדנים פשוטים כמו k-NN. השונות יורדת מונוטונית עם עליית k כאשר יותר שכנים ממוצעים וההטיה עולה מונוטונית מכיוון שהממוצע כולל שכנים פחות דומים. איזון זה יוצר את העקומה בצורת U המוכרת מספרי הלימוד עבור שגיאת החיזוי כפונקציה של מורכבות המודל.

אולם במשטר D_r מוליד התנהגות חדשה- האינטואיציה של הטרייד-אוף בין הטיה לשונות כבר לא עובדת בצורה כה פשוטה. ההטיה אינה יורדת מונוטונית עם המורכבות: השכן הקרוב ביותר עשוי שלא להתאים באופן מושלם לנקודת הבדיקה, מה שמוביל להטיית התאמת שכנים שאינה אפס. ההטיה יכולה להציג דפוס בצורת U, כאשר מודלים בעלי מורכבות בינונית ממזערים את ההטיה. התנהגות זו ניתן לבטא על ידי פירוק ההטיה ל:

$$\mathrm{Bias}_k(x_0) = \left(f^*(x_0) - f^*\left(\sum_{i=1}^{n} w_{k,i}(x_0)x_i\right)\right) + \left(f^*\left(\sum_{i=1}^{n} w_{k,i}(x_0)x_i\right) - \sum_{i=1}^{n} w_{k,i}(x_0)f^*(x_i)\right)$$

שני הרכיבים הם:
הטיית התאמת שכנים: נוצרת כאשר הממוצע המשוקלל של נקודות האימון אינו משחזר באופן מושלם את נקודת הבדיקה.
הטיית מיצוע: נובעת אי-לינאריות של פונקציה האמיתית (כלומר המיפוי מנקודה ללייבל).

פירוק זה חושף שגם במצבים פשוטים ונמוכי-ממד, תכנון אקראי מכניס מורכבויות שמשבשות אינטואיציות קלאסיות.

תופעת Double Descent:

תופעת DD מתייחסת להתנהגות הלא-מונוטונית של שגיאת החיזוי כפונקציה של מורכבות המודל. היא מורכבת מעקומה בצורת U במשטר under-parametrization (מספר פרמטרי מודל קטן ממספר הדוגמאות) וירידה שנייה במשטר over-parameterization (מספר פרמטרי מודל גדול ממספר הדוגמאות). המחברת מדגישה כי DD אינו יכול להתרחש במצבי D_f מכיוון שאינטרפולציה תמיד מובילה לשגיאת in-sample קבועה ERR_fixed = σ^2. זאת מכיוון שמודלים במשטר זה חוזים באופן מושלם בנקודות האימון, מה שמוביל להטיה ושונות אפס בתכנון קבוע. עם זאת, במשטר D_r, תופעת DD מופיעה באופן טבעי בגלל שינויים במורכבות המודל האפקטיבית (שלא נמדדת במספר הפרמטרים) ותכונות ההכללה בעת המעבר מעבר לאינטרפולציה.

Benign Overfitting(BO) vs. Benign Interpolation(BI)

המחבר מבקר את המונח BO, ומציע במקומו מונח BO. הגדרות קלאסיות של אוברפיט מרמזות על ביצועי הכללה ירודים, מה שסותר את הרעיון שביצועים מושלמים בטריין סט יכולה לעתים להניב ביצועים טובים גם על הטסט. במשטר R_f, אינטרפולציה אינה יכולה להיות benign בגלל הדומיננטיות של שונות הרעש ERR_fixed = σ^2.

במשטר R_d, לעומת זאת, מודלים כמו רשתות נוירונים ויערות אקראיים(random forests) יכולים להציג התנהגות חדה-חלקה, בה הם מבצעים אינטרפולציה חדה בנקודות האימון אך מכלילים בצורה חלקה לקלטים שלא נראו. התנהגות זו ניתנת לכימות באמצעות מדדי מורכבות אפקטיבית. זאת אומרת מודלים שמפחיתים מורכבות אפקטיבית על טסט סט נוטים להציג אינטרפולציה שפירה.

השלכות:

חשיבה מחדש על חינוך סטטיסטי: קורסי מבוא צריכים להבהיר את ההבחנה בין R_f ל-R_d.

הסקה סיבתית ו-ML: בתחומים בהם נקודות מסט אימון עשויים לחזור (למשל, הסקה סיבתית), הנחות R_d עשויות עדיין להיות רלוונטיות.

בחירת מודל ML: הבנה מתי אינטרפולציה היא benign דורשת מדידת מורכבות בזמן בדיקה, לא רק ביצועי אימון.

סיכום

עבודה זו מציעה פרספקטיבה מאוד מעניינת על מדוע אינטואיציות סטטיסטיות קלאסיות לא תמיד עובדת טוב בבעיות ב-ML מודרני. על ידי הדגשת השוואה בין R_f ל- R_d, המאמר מספק מסגרת מאחדת להבנת DD, Benign Interpolation, והתפקיד המתפתח של טרייד-אוף ההטיה-שונות.

https://arxiv.org/pdf/2409.18842

המאמר היומי של מייק - 03.02.25
The Perfect Blend: Redefining RLHF with Mixture of Judges

אחרי יציאת המודל האחרון של DeepSeek העניין ל-RLHF או שיטת טיוב (fine-tuning) של מודלי שפה באמצעות שיטת Reinforcement Learning with Human Feedback. חוקרי DeepSeek הראה שניתן לאמן מודל שפה חזק לעשות הנמקה(reasoning) בעיקר עם RLHF (יש קצת SFT אבל עדיין הרוב). המאמר שנסקור היום יצא כמעט 4 חודשים לפני R1 של DeepSeek והוא מציע שיטה שמשפרת ביצועים של RLHF.

אחת הבעיות הגדולות של אימון RLHF הוא reward hacking שמתרחש כאשר המודל לומד למקסם את פונקציית התגמול (reward) אך כתוצאה מכך מתכנס למודל חלש או לא בטוח (למרות איבר הרגולריזציה שמנסה לשמור את המודל הסופי קרוב למודל שממנו מתחילים לעשות RLHF). המחברים מציעים להתמודד עם הבעיה הזו בשלוש דרכים. הראשונה היא סט של אילוצים על התשובה לפרומפט (שלמשל בודק האם הוא פוגעני) הנבדק על ידי "השופט" (judge) שתפקידו ממלא מודל שפה אחר. השיפור השני הוא שינוי של פונקציית תגמול המתבטא בחיסור ממנו תגמול בייסליין מסוים שתיכף אסביר מהו. השינוי השלישי בבניית הוא דאטהסט עליו מאומן RLHF.

החיסור הזה מזכיר לי שני דברים. קודם כל התגמול החדש (אחרי החיסור) נראה דומה לפונקציית יתרון (advantage) (רק דומה אבל היא לא) המוכרת לנו פונקציית לוס (יעד) משיטת PPO רק שהפעם היא לא מחושבת דרך פונקציית Value באמצעות שיטות GAE אלא בדרך אחרת. תגמול חדש זה מזכיר לנו מה שראינו בפונקציית יעד של המאמר של DeepSeek, שם הבייסליין חושב באמצעות תגמול ממוצע(מעל באץ') של המודל המטויב (המתקונן עם השונות). במאמר ההוא איבר זה שימש כאומדן של אותה פונקציית היתרון.

כאמור החידוש השני של המאמר (הראשון האילוצים שאנו מטילים על פלטי המודל) הוא הבייסליין המחוסר מהתגמול. המחברים מציעים לקחת את הבייסליין בתור התגמול עבור דוגמאות (תשובות) הזהב(= מועדפות) מדאטהסט של SFT (שאלות ותשובות) או מהשאלות עם התשובות המועדפות מדאטהסט של RHLF. כך התגמול שלנו הוא עד כמה התשובות של המודל המאומן נראות יחסית לתשובות המועדפות מבחינת תגמולן.

השינוי השלישי הוא בפונקציית יעד. בנוסף (המחברים מציעים 2 וריאנטים) למקסום של הנראות של התשובות המועדפות ומזעור הנראות לתשובות הפחות מועדפות (מבחינת התגמול), המאמר מציע רק למקסם את הנראות

של התשובות המועדפות בלבד (באופן מפתיע זה עובד). המחברים גם "מלבישים" את הרעיונות הללו ששיטות קלאסית של RLHF כמו DPO ו-RAFT.

https://arxiv.org/abs/2409.20370

המאמר היומי של מייק - 03.02.25

Deep Generative Models through the Lens of the Manifold Hypothesis: A Survey and New Connections

תמצית המאמר:

רציתם לדעת למה מודלי דיפוזיה ניצחו את הגאנים, VAE וכל השאר מזווית מתמטית? רוצים להבין בעזרת מתמטיקה למה מודלי דיפוזיה לטנטיים עובדים מעולה? תצללו לסקירה הזו...

מאמר זה מציע חקירה מקיפה של מודלים גנרטיביים עמוקים (DGMs) תחת המסגרת של השערת היריעה, הטוענת שדאטה בעל ממד גבוה נמצאים לעתים קרובות על תת-יריעה בעלת ממד נמוך יותר המוטמעת בתוך המרחב המקורי (במאמר נקרא אמביינטי). המחברים מספקים הסבר מדוע מודלים כמו מודלי דיפוזיה ו- GANs מסוימות מציגים ביצועים טובים יותר מאחרים, כולל שיטות מבוססות נראות כמו אוטואנקודרים וריאציוניים (VAEs) וזרימות נורמליזציה (NFs). על ידי אימוץ נקודת מבט מבוססת יריעה, המחברים מספקים תובנות לגבי המגבלות המובנות של גישות קיימות תוך יצירת קשרים תיאורטיים חדשים בין DGMs והסעה אופטימלית..

המחקר בולט בכך שהוא מוכיח באופן פורמלי את חוסר היציבות הנומרית המובנית שמודלים מבוססי נראות בממד גבוה חווים כאשר הם מנסים לייצג דאטה על יריעה, ומציע פרשנות חדשה של DGMs דו-שלביים כמקרבים של מרחק וסרשטיין בין התפלגות המודל להתפלגות הדאטה האמיתי.

נקודות מרכזיות

1. סקירה של מודלים DGM מודעי-יריעה ולא-מודעי-יריעה (manifold-aware and manifold unaware)

מודלים לא-מודעי-יריעה: מודלים אלה אינם מתחשבים באופן מפורש במבנה היריעה של דאטה. דוגמאות כוללות VAEs, NFs ומודלים מבוססי אנרגיה. מודלים כאלה נוטים להתאמת יתר ליריעה, כאשר הצפיפויות שואפות לאינסוף לאורך היריעה אך נכשלים בשעורכה של ההתפלגות בתוכה.

מודלים מודעי-יריעה: מודלים אלה מוסיפים רעש כדי לפזר את מסת ההסתברות מעבר ליריעה או מאפטמים פונקציות יעד שאינם מגבילות את ההתפלגות על היריעה שתופסות באופן לא מפורש את מבנה היריעה. דוגמאות כוללות מודלי דיפוזיה, התאמת זרימה מותנית (conditional flow models), ו-Wasserstein GANs.

2. **חוסר יציבות נומרית של שיטות מבוססות נראות**

אחת התרומות התיאורטיות המרכזיות היא ההוכחה שמודלים מבוססי נראות סובלים מחוסר יציבות מספרית בלתי נמנע כאשר הם מנסים למדל הדאטה הנתמך על יריעה. המחברים מדגימים שכאשר צפיפויות המודל מנסות להתרכז על היריעה, פונקציית הנראות הופכת לבלתי מוגבלת, מה שמוביל לפתרונות מנוונים(זה קורה הרבה ב-VAE וב-GANים רגילים).

מתמטית, אם P_X התפלגות הדאטה ב- R^d בעלת תומך של יריעה M בעלת ממד פנימי d*< d, עבור כל סדרה של מודלים מבוססי נראות { P_{X, θ_t} המקרבים את התפלגות דאטה מתקיים:

$$\lim_{t\to\infty} p_{X,\theta_t}(x) = \infty \text{ for all } x \in M.$$

תוצאה זו מרמזת שצפיפויות במרחב הדאטה מתבדרות באופן מובנה כאשר הן מנסות למדל התפלגויות הנתמכות על יריעה, מה שהופך את היעדים מבוססי הנראות לבעייתיים עבור דאטה כזה כאלה.

3. **מגבלות מרחק KL:**

המחברים מדגישים שמרחק KL, יעד נפוץ לאימון DGMs, הופך ללא יעיל בלמידת היריעה. הבעיה העיקרית מתעוררת כי KL מניח ששתי ההתפלגויות חולקות את אותה תומך (support). אולם כאשר משווים צפיפות של מודל במרחב הדאטה {p_{X,θ עם התפלגות דאטה P_ הנתמכת על יריעה, ה-KL הופך לאינסופי:

$$KL(p_X||p_{X,\theta}) = \infty.$$

תופעה זו מתרחשת כי P_x מקצה הסתברות שאינה אפס רק לנקודות על היריעה, בעוד {p_{X,θ מפזר מסת הסתברות על פני כל המרחב האמביינטי. כתוצאה מכך, מקסום הנראות, השקול למזעור את ה-KL, נכשל במתן אות למידה משמעותי.

4. **מרחק וסרשטיין כיעד חלופי**

כדי להתמודד עם מגבלות ה-KL, המחברים מקדמים את השימוש במרחקי וסרשטיין(זה עובד לא רע בגאנים כאמור), שנשארים מוגדרים היטב גם כאשר להתפלגויות יש תמיכות לא תואמות. מרחק וסרשטיין-1 בין התפלגויות p ו-q מוגדר כ:

$$W_1(p,q) = \inf_{\gamma\in\Pi(p,q)} \mathbb{E}_{(X,Y)\sim\gamma}[||X-Y||]$$

כאשר Π(p,q) מסמן את קבוצת ההתפלגויות המשותפות עם בעלות התפלגות שולית p ו-q. בניגוד ל- KL, מרחק וסרשטיין ממטר התכנסות חלשה, מה שהופך אותו ליעד חסין לאימון DGMs בתרחיש היריעה.

5. **פרשנות של מודלים לטנטיים**

המחברים מספקים פרשנות חדשה של DGMs לטנטיים שקודם לומדים ייצוג בממד נמוך של יריעת דאטה ואז ממדלים את ההתפלגות בתוך ייצוג זה. הם מראים שמודלים אלה ממזערים באופן יעיל חסם עליון של מרחק וסרשטיין בין התפלגות המודל להתפלגות דאטה האמיתית:

$$W_1(p_X, p_{X,\theta}) \leq \text{Reconstruction Error} + \text{Distributional Divergence}$$

כאשר שגיאת השחזור מודדת עד כמה טוב היריעה שנלמדה מקרבת את יריעת הדאטה האמיתית, והמרחק בין ההתפלגות מכמת את ההבדל בין התפלגויות בתוך היריעה שנלמדה. תוצאה זו מספקת הצדקה תיאורטית להצלחה האמפירית של מודלי דיפוזיה לטנטיים וגישות דו-שלביות אחרות.

תובנות מתמטיות

משפט חוסר היציבות המספרית

המחברים מוכיחים באופן פורמלי שעבור כל התפלגות הדאטה P הנתמכת על יריעה וכל סדרה של צפיפויות מודל במימד הדאטה Q, פונקציית יעד בצורה של נראות לא מתכנסת מקסימום. תוצאה זו נגזרת מניתוח התנהגות הצפיפויות על יריעות בממד נמוך תוך שימוש בתכונות של גיאומטריה דיפרנציאלית ותורת המידה (די כבד האמת).

מזעור מרחק וסרשטיין:

על ידי הצגת מודלים דו-שלביים כמקרבים של מרחק וסרשטיין, המחברים מבססים קשר בין למידת יריעה וטרנספורט אופטימלי. תובנה זו לא רק מסבירה את הביצועים העדיפים של מודלי דיפוזיה לטנטיים אלא גם מספקת מסגרת עקרונית לתכנון DGMs חדשים.

הסבר קריסת מודים:

המחברים מראים שקריסת מודים ב-VAEs ו-GANs ניתנת להבנה כתוצאה של התאמת יתר ליריעה, כאשר צפיפויות המודל מתבדרות לאורך תתי-קבוצות של היריעה מבלי "לתפוס" את התפלגות הדאטה האמיתית.

מודלי דיפוזיה:

ההצלחה של מודלי דיפוזיה מיוחסת ליכולתם להתחשב באופן מרומז במבנה היריעה על ידי פיזור מסת הסתברות מעבר ליריעה. המחברים מספקים ניתוח מפורט של מודלי דיפוזיה מבוססי-ציון וגרסאות חבויות שלהם.

סיכום

מאמר זה מספק חקירה קפדנית ומעמיקה של DGMs דרך עדשת השערת היריעה. על ידי זיהוי המגבלות של שיטות מבוססות-נראות והדגשת היתרונות של מרחקי וסרשטיין ומודלים לטנטיים, המחברים סוללים את הדרך לפיתוח מודלים גנרטיביים יעילים יותר.

המאמר היומי של מייק - 06.02.25

SMALL LANGUAGE MODELS: SURVEY, MEASUREMENTS, AND INSIGHTS

תמצית:

המאמר זה חוקר את החשיבות הגוברת של מודלי שפה קטנים (SLMs) ומשווה את התפתחותם ל-LLMs. בעוד ש-LLMs דורשים משאבי מחשוב משמעותיים ובדרך כלל מופעלים בשרתים, SLMs מתוכננים לפעול במכשירים מוגבלי משאבים כמו מחשבים ניידים, טאבלטים, סמארטפונים ומכשירי IoT. המחקר מציע סקירה מקיפה של 59 מודלי SLM, מעריך אותם על בסיס התקדמויות ארכיטקטוניות, אלגוריתמי אימון ויעילות הסקה. באמצעות קידום אימוץ SLM, עבודה זו שואפת להפוך את הבינה המלאכותית לנגישה, זולה ויעילה יותר ליישום מעשי.

אני אתרגם את הטקסט לעברית:

ניתוח טכני

1. **חידושים ארכיטקטוניים** מאמר המחקר בוחן לעומק את האלמנטים הארכיטקטוניים המבדילים בין SLM ל-LLM, תוך הדגשת השינויים המשפרים את היעילות במכשירים עם משאבים מוגבלים.

- **מנגנוני self-attention:** באופן מסורתי, attention מרובת-ראשים (MHA) הייתה המנגנון הדומיננטי ב-SLM. עם זאת, המאמר מזהה מעבר הדרגתי לטכניקת תשומת לב מבוססת-קבוצות (GQA). גרסה זו מפחיתה את המורכבות החישובית על ידי שיתוף ייצוגי שאילתות בין ראשים תוך שמירה על גיוון בייצוגי מפתח-ערך. הדוח מספק ראיות לכך שמודלי GQA, כמו Qwen2.5, עולים משמעותית על אלה עם MHA מבחינת ה-latencies ויעילות זיכרון, במיוחד בשלב ההיסק (אינפרנס).
- **רשתות feed-forward:** האבולוציה הארכיטקטונית מראה העדפה ל-Gated FFNs (שזה שילוב של שני FFNs -אחת עם אקטיבציה א לינארית בסוף) על פני FFN סטנדרטיות. Gated FFN מפעיל באופן סלקטיבי חלקים מהרשת, מה שמוביל ליעילות פרמטרית טובה יותר. ממצא מעניין הוא הגיוון ביחסי הביניים ב-Gated FFNs, (כלומר המימד של השכבה הלינאריות הראשונה שם) הנעים בין פי 2 ל-8 מהממד החבוי, כאשר יחסים גדולים יותר משפרים בדרך כלל את הדיוק במשימות היסק מורכבות.
- **פונקציות אקטיבציה:** נצפה מעבר משמעותי מ-GELU (שזה Gaussian Error Linear Unit) וגרסאותיה ל-SiLU (שזה Sigmoid Linear Unit). המאמר מציין כי SiLU יעילה מבחינה חישובית ומתאימה יותר למודלים מקוונטטים, שולטת במודלים שהושקו ב-2024.
- **נרמול שכבות:** מודגש המעבר מ-LayerNorm ל-RMSNorm. RMSNorm מפחית את העומס החישובי על ידי ביטול הצורך בחישוב הממוצע במהלך הנורמליזציה, יתרון משמעותי במכשירי קצה.
- **גודל מילון:** גדלי מילון של SLM גדלו משמעותית, ולעתים קרובות עולים על 50K טוקנים. המחברים מקשרים עלייה זו עם יכולות משופרות בהבנת שפה.

ניתוח דאטהסטים לאימון:

- **מגמות בשימוש בדאטהסטים:** המחקר מתעד מעבר מדאטהסטים כלליים נפוצים כמו *The Pile* ו-*RefinedWeb* לדאטהסטים מאוגדים כמו *FineWeb-Edu* ו-*DCLM*. דאטהסטים החדשים הללו משלבים טכניקות סינון מבוססות-מודל המשפרות משמעותית את איכות דאטה.
- **איכות דאטה לעומת כמות:** למרות ההסתמכות המוקדמת על נפח דאטה גדול, הדוח מוצא שדאטהסטים באיכות גבוהה מניבים ביצועי מודל טובים יותר, גם עם פחות טוקנים. לדוגמה, מודלים שאומנו על *FineWeb-Edu* משיגים דיוק תחרותי בהשוואה למודלים מסחריים (סגורי-קוד) מתקדמים.
- **כמה טוקנים צריך לאימון**: המחברים מציינים מגמה מפתיעה: SLM רבים מאומנים על מספר טוקנים הגדול בהרבה ממה שחוק צ'ינצ'ילה מציע. לדוגמה Qwen2.0 - 500M מאומן על 12 טריליון טוקנים, בעוד ש-Qwen2.0 - 1.5B מאומן על 7 טריליון בלבד. אסטרטגיית "אימון-היתר" המכוונת הזו מוצגת כאופטימיזציה לסביבות מוגבלות-משאבים, המאפשרת למודלים להכליל טוב יותר כאשר הם מופעלים במכשירים עם כוח חישוב מוגבל.

3. **חידושים באלגוריתם האימון**

- **שיטת Maximal Update Parameterization - μP**: בשימוש במודלים כמו Cerebras-GPT, שיטת μP מבטיחה אימון יציב על ידי בקרה על אתחול, קצבי למידה בכל שכבה, ועוצמות ראקטיבציה. טכניקה זו מאפשרת להעביר היפר-פרמטרים שאופטמו עבור מודלים גדולים ישירות למודלים קטנים יותר, מה שמייעל את תהליך האימון.

- **זיקוק ידע(דיסטילציה):** LaMini-GPT ו-Gemma-2 מנצלים טכניקה זו להעברת ידע ממודלי מורה גדולים למודלי תלמיד קטנים יותר, מה שמוביל לביצועים משופרים ללא צורך באימון נרחב.
- **אסטרטגיית אימון מקדים דו-שלבית:** אומצה על ידי MiniCPM, אסטרטגיה זו כוללת שלב ראשוני עם דאטה באיכות נמוכה ולאחר מכן פיין-טיון עדין עם דאטה באיכות גבוהה וספציפיים למשימה. השיטה מוכיחה את עצמה כיעילה באיזון בין יעילות חישובית לביצועי המודל.

4. הערכת ביצועים

היסק מבוסס שכל ישר: מודלים כמו Phi-3-mini משיגים ביצועים מתקדמים, (המתחרים ב-LLaMA 3.1 7B). תוצאות הבנצ'מרק מגלות שהשיפורים באיכות מערך הדאטה ואסטרטגיות האימון אפשרו ל-SLM לצמצם את הפער עם מודלים גדולים יותר.

פתרון בעיות ריזונינג: Phi-3-mini ו-SLM אחרים בעלי ביצועים גבוהים מציגים שיפור של 13.5% בביצועים משנת 2022 עד 2024, עולים על קצב השיפור של מודלי LLaMA. זה מדגים את הבשלות הגוברת של SLM בטיפול במשימות היסק מורכבות.

מתמטיקה: ביצועי ה-SLM נשארים תת-אופטימליים במתמטיקה, כאשר המודלים מתקשים לטפל במשימות הדורשות חשיבה לוגית. המחברים מייחסים פער זה למחסור בדאטהסטים באיכות גבוהה המתמקדים בלוגיקה.

למידת in-context: הניסויים מגלים ש-SLM מפיקים תועלת משמעותית מלמידה בהקשר, במיוחד עבור משימות כמו אתגר ARC, שם נצפים שיפורים בדיוק של עד 4.8%. עם זאת, חלק מהמודלים, כמו LaMini-LM, מציגים הידרדרות בביצועים עקב אוברפיט.

5. ניתוח יעילות בזמן ריצה

לייטנסי וזיכרון נדרש: המחקר מוצא שהלייטנסי של היסק מושפעת הן מגודל המודל והן מהארכיטקטורה. לדוגמה, Qwen1.5 רץ 31.9% מהר יותר מ-Qwen2.0 על המעבד Jetson Orin, למרות שיש לו 25.4% יותר פרמטרים. זה מיוחס להבדלים במנגנוני attention ואסטרטגיות שיתוף פרמטרים (parameter sharing). צריכת זיכרון היא לינארית בד"כ ביחס לגודל המודל אך מושפע גם מגורמים כמו גודל אוצר המילים ומנגנוני תשומת לב. מודלים כמו Bloom-1B1, שיש להם אוצרות מילים גדולים יותר, מציגים שימוש גבוה באופן לא פרופורציונלי בזיכרון.

קווינטוט: טכניקות קווינטוט, במיוחד 4 ביט, מוכיחות את עצמן כיעילות בהפחתת השהיה ושימוש בזיכרון. שיטת Q4 KM מפחיתה לייטנסי בממוצע ב-50% במהלך היסק, עולה על שיטות כימות 3-ביט ו-6-ביט, הסובלות מחוסר יעילות חומרתית.

סיכום:
הניתוח הטכני המוצג במאמר זה מספק הבנה מקיפה של השיקולים הארכיטקטוניים, האימון, וזמן הריצה החיוניים לפיתוח ופריסה של SLM. על ידי התמודדות עם אתגרי היעילות ומגבלות המשאבים, המאמר מציע תובנות חשובות לקידום מחקר ה-SLM ויישומים מעשיים.

https://arxiv.org/abs/2409.15790

המאמר היומי של מייק - 08.02.25
Rejection Sampling IMLE: Designing Priors for Better Few-Shot Image Synthesis

היום עושים הפסקה קלה עם LLMs וסוקרים מאמר המציע שיטה מעניינת לאימון מודלי גנרטיביים במקרה שיש לכם מעט דאטה לאימון. כידוע מודלים גנרטיביים מודרניים כמו מודלי דיפוזיה, גאנים, VAEs מצריכים כמות עצומה של דאטה אבל לפעמים אין לנו את הלוקסוס הזה ואנו צריכים לאמן על כמות קטנה של דאטה. האם זה אפשרי בכלל?

התשובה על כך חיובית (לפחות לפי המאמר). המחברים מציעים שיטה הנקראת RS-IMLE לאימון מודל גנרטיבי עם מעט דאטה שמשכלל שיטת IMLE שזה Implicit Maximum Likelihood Estimation. בגדול מאוד IMLE די דומה לשיטה גנרטיבית סטנדרטית - היא דוגמת משתנה בעל התפלגות קלה לדגימה (גאוסית) z ומאמנת מודל גנרטיבי(רשת נוירונים) כדי לגנרט פיסת דאטה. ההבדל הוא בפונקציית לוס: עם IMLE לכל דגימה x מהדאטהסט אנו ממזערים את רק המרחק בינה לבין נקודה z_i אחת בלבד: כזו ש-T(z_i) שלה הינו קרוב ביותר אליה. כאן T(z_i) היא פיסת דאטה שגונרטה מ-z_i ו- T זה המודל שאנו מאמנים.

כלומר בשלב הראשון של IMLE אנו דוגמים m נקודות ומעבירים אותם דרך מודל T(נקרא לו מיפוי בהמשך) ובונים m פיסות דאטה מגונרטות. לאחר מכן לכל דגימה x_j מדאטהסט האימון אנו בוחרים את z_i הקרובה ביותר ל-x_j. בסוף רק נקודות כאלו משתתפות במזעור של פונקצית לוס. כמובן שמספר הנקודות m המגונרטות בשלב הראשון צריך להיות גבוה משמעותית מאשר גודל הדאטהסט לאימון n. המטרה של שיטת אימון זו היא לאפטם את המודל רק עבור הנקודות במרחב הלטנטי (z) שהן הממופות קרוב לנקודת מהדאטהסט.

הבעיה עם הגישה הזו שההתפלגות של הנקודות "הנבחרות" במהלך האימון כבר לא גאוסית שעלול ליצור לנו בעיות באינפרנס כי אנו כן רוצים לדגום את z מהתפלגות גאוסית. המרחק בין מיפוי T של דגימה גאוסית מנקודה מהדאטהסט שונה בהתפלגות מזה של הדגימה z הממופה הכי קרוב לקודה זו (האמת זה די ברור). דרך אגב המאמר מוכיח את הטענה הזו ומציע שיטה להתגבר על זה.

השיטה שהמאמר מציע נראית ממש פשוטה אך מבוססת על ניתוח מתמטי די מעמיק של התפלגויות המרחקים. בשלב הראשון של האימון (אחרי הדגימה מהתפלגות גאוסית) בוחרים את z_i כאשר נופלים במרחק יותר גדול מבוע אפסילון מכל נקודות בדאטהסט האימון אחרי המיפוי (כלומר יש לנו rejection sampling). לאחר מכן, בדומה ל-IMLE, לכל נקודה x בדאטהסט בוחרים את z שהמיפוי שלו עם T נופל הכי קרוב אליה ומאמנים את T למזער את המרחק הממוצע בין z-s הנבחרים לנקודות העוגן שלהם. הייפרפרמטרים החשובים כאן זה אפסילון ומספר נקודות z שנדגמות.

אינטואיטיבית זה עובד כי מלכתחילה אנו בוחרים נקודות רחוקות יותר (לאחר המיפוי) מהנקודות בדאטהסט שמאפשר לשמור התפלגות של הנקודות הנבחרות בשלב לאחר מכן קרובה לגאוסית.

https://arxiv.org/abs/2409.17439

המאמר היומי של מייק - 09.02.25
Why Is Anything Conscious?

מבוא:

המאמר המעניין מאת מייקל טימותי בנט, שון וולש ואנה צ'יאוניקה מתמודד עם "הבעיה הקשה של התודעה", שנוסחה על ידי דייויד צ'אלמרס(David John Chalmers). אתגר פילוסופי זה מעלה את השאלה מדוע עיבוד

מידע במערכות מסוימות, במיוחד ביולוגיות, מוביל לחוויות סובייקטיביות או *קוואליה*. המחברים מציעים שינוי פרדיגמה, המעגן את התודעה בדינמיקה של מערכות self-organizing שעוצבו על ידי הברירה הטבעית.

הם טוענים כי תודעה תופעתית (phenomenal) - החוויה הסובייקטיבית של "איך זה מרגיש" - אינה רק יסודית אלא הכרחית להתנהגות אדפטיבית. מעניין כי באמצעות פריימוורק חישובי פורמלי, המחברים טוענים נגד האפשרות של "זומבים", מערכות המתפקדות כמו בני אדם אך חסרות חוויה סובייקטיבית, ומצהירים באופן פרובוקטיבי כי "הטבע אינו אוהב זומבים". חוויה סובייקטיבית היא ההבנה המלאה והחווייתית של ההשפעה הרגשית והקוגניטיבית כאחד הנובעת מאופן שבו הבני אדם מבינים ומפרשים אירועים שנצפו או נחוו על ידי הם.

תרומות מרכזיות:

מסגרת מתמטית לאנקטיביזם פנ-חישובי

המחברים מציגים מערכת פורמלית המעוגנת ב*פנ-חישוביות* ו*אנקטיביזם*(Pancomputational Enactivism). פנ-חישוביות מניחה שכל המערכות הדינמיות מחשבות משהו, בעוד שאנקטיביזם מדגיש את ההכרה כנובעת מאינטראקציות בין מערכת לסביבתה. האלמנטים המרכזיים במודל שלהם כוללים:

- סביבה: מוגדרת כקבוצת מצבים, עם מעברים המתוארים על ידי תכנות דקלרטיבי.

- שכבת הפשטה: מבנה המגדיר כיצד מערכות מפרשות היבטים סביבתיים.

- משימות ומדיניות: מבני התנהגות הממפים קלט לפלט, המאפשרים התנהגות אדפטיבית.

- זהויות סיבתיות:*ייצוגים של התערבויות והשפעותיהן, חיוניים למודעות עצמית.

הפריימוורק מתאר כיצד מערכות מודעות שומרות על קוהרנטיות והסתגלות על ידי בניית זהויות סיבתיות מורכבות יותר ויותר, המהוות בסיס למודעות עצמית.

היררכיה של תודעה

תובנה מרכזית היא ההתפתחות ההיררכית של התודעה, המונעת על ידי ברירה טבעית ולחצי סקאלה. המחברים מתארים 6 שלבים מתקדמים:

1. מערכות לא מודעות: ישויות חסרות חוויה או הכרה, כמו סלעים.

2. מערכות מקודדות באופן קשיח: מערכות עם תגובות קבועות, מתוכנתות מראש (למשל, חד-תאיים).

3. מערכות לומדות: מערכות מסתגלות ללא מודעות עצמית (למשל, תולעים נמטודות).

4. מערכות עצמי מסדר ראשון: מסוגלות להבחין בין פעולות שנוצרו עצמאית לבין אירועים חיצוניים (למשל, זבובי בית).

5. מערכות עצמיות מסדר שני: מסוגלות למטא-ייצוג ותקשורת מכוונת (למשל, עורבים).

6. מערכות עצמי מסדר שלישי: ישויות רפלקטיביות במלואן המסוגלות לחשוב על המודעות שלהן עצמן (למשל, בני אדם).

היררכיה זו מדגישה כיצד היבטים איכותיים של תודעה מתפתחים באופן טבעי ככל שמערכות נעשות מסוגלות יותר למדל את עצמן ואת סביבתן.

עיבוד איכותי וכמותי:

המחברים טוענים כי *איכות קודמת לכמות* בעיבוד מידע. לפני שאורגניזם יכול לתייג או למדוד מידע, עליו לחוות הבדלים איכותיים. תודעה פנומנלית מתפתחת מכיוון שמערכות חיות חייבות לסווג ולתעדף מידע הרלוונטי להישרדות. סיווגים איכותיים אלה מהווים את הבסיס לחוויה סובייקטיבית. טענה זו מאתגרת תיאוריות חישוביות מסורתיות, המתייחסות לעתים קרובות לתודעה כתהליך ייצוגי טהור. על ידי הדגשת הקדימות של החוויה האיכותית, המחברים מספקים פרספקטיבה רעננה על מקורות התודעה.

גישת עקרונות ראשוניים:

הפורמליזם במאמר נגזר משתי אקסיומות בסיסיות:

1. במקום שיש דברים, אנו קוראים לדברים אלה הסביבה.

2. במקום שדברים שונים, יש לנו מצבים שונים של הסביבה.

אקסיומות אלה מובילות לצורה חסרת ייצוג של פנ-חישוביות, בה מצבים ומעברים מגדירים סביבות מבלי להניח מבנים פנימיים ספציפיים. המחברים ממסגרים ארגון עצמי כיכולת להגביל פלטים על בסיס קלטים, ובכך להשיג התנהגות אדפטיבית.

דחיית זומבים

אחת הטענות המעניינות ביותר במאמר היא ש"הטבע אינו אוהב זומבים". המחברים טוענים שתודעה פנומנלית חיונית למודעות גישה ולהתנהגות אדפטיבית. תוכן ייצוגי - מה שאורגניזמים חושבים עליו - נגזר תמיד מחוויה איכותית. לכן, מערכת המתנהגת כמו ישות מודעת חייבת בהכרח לחוות חוויה סובייקטיבית. טענה זו מאתגרת ישירות ניסויי מחשבה המציעים את קיומן של ישויות לא מודעות אך זהות בהתנהגותן.

קשרים אמפיריים

המאמר מבוסס על ממצאים אמפיריים לגבי *רה-אפרנציה*, כלומר היכולת להבחין בין גירויים שנוצרו עצמאית לבין גירויים חיצוניים. רה-אפרנציה, הנצפית ביונקים וחרקים, קשורה ליצירת עצמי מסדר ראשון. המחברים גוזרים מבנה זה מעקרונות מתמטיים ומיישרים את מסקנותיהם עם עבודתם של מרקר, ברון וקליין.

סיכום:

המאמר מציע גישה מסקרנת לבעיה הקשה של התודעה על ידי עיגונה בברירה טבעית, ארגון עצמי ופורמליזם חישובי. המסגרת ההיררכית של המחברים מספקת הסבר משכנע לאופן שבו תודעה מתפתחת ומדוע חוויה סובייקטיבית היא יסודית להתנהגות אדפטיבית. טענתם הפרובוקטיבית שזומבים הם בלתי אפשריים מאתגרת הנחות ותיקות, ומסמנת מאמר זה כתרומה משמעותית לחקר התודעה.

https://arxiv.org/abs/2409.14545

המאמר היומי של מייק - 10.02.25
On the expressiveness and spectral bias of KANs

מבוא:

המאמר שאסקור היום מציג חקירה מעמיקה של רשתות קולמוגורוב-ארנולד (KANs), ארכיטקטורה חדשנית המבוססת על משפט הייצוג של קולמוגורוב-ארנולד. המחברים משווים באופן מדוקדק בין KANs לבין רשתות MLPs מסורתיות, הן מבחינה תיאורטית והן אמפירית, תוך התמקדות בהיבטים כמו אקספרסיבנס, יעילות ודינמיקת אימון. המאמר מבסס תכונות תיאורטיות מרכזיות ומאמת אותן באמצעות ניסויים, ובכך מהווה תרומה משמעותית לתכנון רשתות נוירונים למשימות חישוב שונות.

אקספרסיבנס:

הישג מרכזי של עבודה זו הוא ההוכחה הפורמלית ש- KANs הן בעלות אקספרסיבנס לפחות כמו MLPs. המחברים מראים שכל MLP מבוסס ReLU ניתן "למפות" לארכיטקטורת KAN מקבילה, תוך שמירה על יעילות וללא הגדלה משמעותית בגודל הרשת. מנגד, בעוד ש-KANs ניתנות לייצוג גם על ידי MLPs, טרנספורמציה זו כרוכה בעלות משמעותית: מספר הפרמטרים גדל עם גודל גריד (מספר נקודות עוגן בספליין) של ה-KAN. ממצא זה מרמז ש-KANs עשויות להציע ייצוגים יעילים יותר עבור סוגים מסוימים של פונקציות, במיוחד כאשר נעשה שימוש במבני גריד עדינים.

המחקר מנצל תוצאות קיימות עבור MLPs כדי לקבוע קצבי קירוב לפונקציות עבור KANs במרחבים פונקצייאונליים שונים כמו מרחב סובולב. הוא מדגים ש-KANs משיגות קצבי קירוב דומים או טובים יותר מאשר MLPs בשערוך פונקציות מורכבות, מה שמחזק את חוסנן התיאורטי.

ניתוח הטיית ספקטרלית (spectral bias):

אחד ההבדלים המרכזיים בין KANs ל-MLPs המודגשים במאמר זה הוא ההבדל בהטיה הספקטרלית שלהם - תופעה שבה רשתות נוירונים נוטות ללמוד תחילה בתדרים נמוכים של פונקציות. המחברים מציגים ניתוח תיאורטי ואמפירי מפורט, המראה ש- KANs סובלות פחות משמעותית מהטיה זו.

הבדל זה מיוחס לפונקציות האקטיבציה מבוססות ה-B-spline ולארכיטקטורה הקומפוזיציונלית של KANs, המאפשרות להן ללמוד תדרים גבוה ביעילות רבה יותר. תובנות תיאורטיות מציעות שדינמיקת האימון של KANs רדודות אחידה יותר ביחס לתדרים השונים בהשוואה ל-MLPs, שבהן נצפית התכנסות מהירה יותר של תדרים נמוכים. ההטיה הספקטרלית המופחתת הופכת את KANs למתאימות יותר למשימות הדורשות שערוך פונקציות בעלות בתדרים גבוהים משמעותיים, כגון פתרון משוואות דיפרנציאליות ומידול תופעות פיזיקליות מורכבות.

ממצאים אמפיריים:

1. **מבחני רגרסיית תדרים**: KANs מצליחות להתאים רכיבי גל בתדר גבוה בו-זמנית, בעוד ש-MLPs מציגות קשיים מתמשכים עם תדרים גבוהים יותר גם לאחר אימון ממושך.

2. **ניסויי שדה גאוסי אקראי:** KANs עולות בביצועיהן על MLPs בקירוב פונקציות שנדגמו משדות גאוסיים גסים, מה שמעיד על יכולת הסתגלות עדיפה למבני פונקציות מורכבים.

3. **פתרונות PDE:** בפתרון משוואות פואסון בתדר גבוה, KANs משיגות שגיאות נמוכות יותר באופן עקבי בהשוואה ל-MLPs, תוך שמירה על ביצועים יציבים גם כאשר תדר הפתרון עולה.

טכניקת הרחבת גריד(של הספליין):

חידוש טכני בולט הנדון במאמר הוא טכניקת הרחבת גריד הייחודית ל- KANs. שיטה זו מאפשרת עידון הדרגתי של גריד של ה-spline במהלך האימון, המאפשר תהליך למידה יעיל יותר. גישת הרחבת הגריד מפחיתה את הסיכונים ל-overfitting ומשפרת את יכולת ההכללה של הרשת, במיוחד כאשר מתמודדים עם פונקציות מורכבות או מערכי נתונים בעלי דגימה חסרה.

סיכום:

עבודה זו מבסמת את KANs כחלופה חזקה ויעילה לרשתות MLPs, במיוחד למשימות בחישוב מדעי. על ידי התמודדות עם הטיה ספקטרלית, שיפור יכולות קירוב, וניצול שיטות אימון אדפטיביות, המחברים מספקים ראיות משכנעות לפוטנציאל של KANs לעלות בביצועיהן על רשתות נוירונים מסורתיות ביישומים הדורשים למידת פונקציות בעלות תדרים גבוהים ומציגות יכולות קירוב משופרות. המסגרת התיאורטית בשילוב עם ניסויים מקיפים הופכת מאמר זה לתרומה חשובה למחקר רשתות נוירונים.

המאמר היומי של מייק - 12.02.25

STUFFED MAMBA: State Collapse and State Capacity of RNN-Based Long-Context Modeling

המאמר מספק חקירה מעמיקה של מצבי כשל במודלים מבוססי RNN במידול שפה עם הקשר ארוך ומציע פתרונות לשיפור יכולות ההכללה שלהם לאורכים גדולים. המחברים מזהים ומנתחים תופעה בעייתית מאוד שקיבלה שם קריסת מצב (State Collapse - SC) - כשל של מודל בעקיבה אחרי דינמיקת של הדאט המונע מרשתות RNN להכליל מעבר לאורכי האימון שלהן. הם מציגים סט של טכניקות מיטיגציה ללא אימון ואסטרטגיות אימון המשכי המאפשרות למודל Mamba2 לעבוד עם מעל מיליון טוקנים מבלי לסבול מקריסת מצב.

הגדרת הבעיה:

מודלי RNN לעומת טרנספורמרים במידול הקשר ארוך

- טרנספורמרים משיגים ביצועים עדיפים במשימות המצריכות הקשר ארוך אך סובלים מסיבוכיות חישובית ריבועית ביחס לאורך הסדרה בשל מנגנון attention.

- מודלי RNN מציגות סיבוכיות לינארית ביחס לאורך הסדרה, מה שהופך אותן ליעילות חישובית בטיפול בסדרות ארוכות.

- מודלים בעלות סיבוכיות לינארית כמו Mamba, RWKV מאומנים בד״כ על סדרות קצרים יחסית (~10K טוקנים) ונכשלים בהכללה מעבר לאורכי האימון(זו הטענה במאמר)

ניתוח כשלים ברשתות RNN (וגם Mamba, RWKV) עם הקשר ארוך

כישלון בהכללה עבור סדרות ארוכות יותר: רשתות אלו מציגות הידרדרות חדה בביצועים כאשר נחשפות לאורכי סדרות מעבר לדאטה שאומנו עליו. כישלון זה אינו נובע פשוט מגרדיאנטים דועכים אלא מיוחס לקריסת מצב (SC).

קיבולת זיכרון קבועה: מכיוון RNNs שומרות על מצב זיכרון בגודל קבוע, יכולתן לשמור מידע היא מוגבלת מטבעה. קיימת מגבלה עליונה על קיבולת הזיכרון ההקשרי - טוקנים מעבר למגבלה זו נשכחים בהכרח.

2. ניתוח פורמלי של קריסת מצב (SC)

הגדרה וממצאים: קריסת מצב (SC) מתרחשת כאשר התפלגות המצב החבוי קורסת(מתנוונת), מה שמוביל לכישלון המודל בעיבוד רצפים ארוכים יותר מקבוצת האימון. המחברים מבצעים ניסויים מבוקרים על Mamba2 ומבחינים שערוצי מצב חבוי מסוימים מציגים התפוצצות של שונות, הגורמת ל:

- ערוצים(channels) חריגים דומיננטיים המדכאים ערכי מצב אחרים.

- חוסר יכולת לשכוח טוקנים מוקדמים, המוביל זיכרון.

- SC מתבטא בעלייה חדה בפרפלקסיות(אי וודאות) מעבר לאורך האימון.

ייחוס תיאורטי: פרמטריזציית יתר בדינמיקת המצב

המחברים מנסחים את משוואת עדכון המצב:

$$h_t = \sum_{i=1}^{t} \alpha_{i:t} \overline{B}_i x_i, \quad \alpha_{i:t} = \left(\prod_{j=i}^{t} \alpha_j \right) \in (0,1)$$

כאשר h_t הוא וקטור המצב החבוי, $\alpha_{i:t}$ המקדמים מייצג את קצב דעיכת הזיכרון, $B_i x_i$ מייצג מידע חדש שהוכנס בצעד זמן. כאשר מאמנים על סדרות באורך T_{train} פרמטרי המודל הנלמדים מעדיפים ״לשמור את כל המידע בתוך T_{train},״ ועקב כך נכשלים בעת עיבוד סדרות ארוכות יותר. זה מוביל לצבירת יתר של מידע, שמובילה לרוויה ובסופו של דבר לקריסת מצב.

3. אסטרטגיות התמודדות נגד SC:

טכניקות הת ללא אימון של SC:

שכחה מבוקרת: הגדלת דעיכת ייצוג מצב(חבוי) על ידי שינוי גורם הדעיכה α_t והפחתת ״עוצמת הכנסה״ של מידע חדש B_i (ייצוג של טוקן) . צעדים אלו גורמים למודל לשכוח טוקנים ישנים באופן אפקטיבי, מונע מייצוג הזכרון להגיע לרוויה (ערכים גבוהים מדי).

נרמול מצב: החלת אילוץ מבוסס נורמה על ייצוג המצב החבוי (מחלקים את וקטור הייצוג בנורמה שלו אם היא גדולה מדי):

זה מונע התפוצצויות של ייצוג מצב חבוי אך מכניס אי-לינאריות, המשפיעה על יעילות האימון (לא ניתן למקבל את החישובים).

$$\hat{h}_t = h_{t-1}\overline{A}_t + \overline{B}^T x_t$$
$$h_t = \begin{cases} \hat{h}_t p/\|\hat{h}_t\| & \text{if } \|\hat{h}_t\| > p \\ \hat{h}_t & \text{if } \|\hat{h}_t\| \leq p \end{cases}$$

עדכון וקטור ייצוג המצב עם sliding window: ניסוח מחדש של כלל עדכון ייצוג המצב לסימולציה של מנגנון sliding window**:

$$h_t^{(r)} = \sum_{i=t-r+1}^{t} \alpha_{i:t}\hat{R}_i = \sum_{i=1}^{t} \alpha_{i:t}\overline{B}_i^T x_i - \alpha_{t-r+1:t}\sum_{i=1}^{t-r} \alpha_{i:t-r}\overline{B}_i^T x_i = h_t - \alpha_{t-r+1:t}h_{t-r}$$

זה מסיר טוקנים ישנים באופן אפקטיבי מבלי לחשב מחדש מאפס. ישים לארכיטקטורות אחרות כמו RWKV ו-RetNet.

המשך אימון על רצפים ארוכים יותר: המחברים מרחיבים את אורכי דאטה האימון מעבר ל"קיבולת ייצוג המצב" כדי לאלץ את המודל ללמוד כיצד לשכוח בהדרגה. הם מאמתים אמפירית שעבור כל גודל ייצוג מצב S, קיים סף אורך אימון שבו SC לא מתרחש.

4. סיכום:

- המחקר השיטתי הראשון של קריסת ייצוג מצב (SC) ברשתות "דמויות" RNN עם אורך הקשר ארוך.SC מתבטא בכך שוקטור ייצוג המצב מגיע לרוויה (ערכים גבוהים) וזה גורם להידרדרות רצינית בביצועי המודל. המאמר מציע 3 שיטות מיטיגציה ללא אימון לביטול SC עד מיליון טוקנים. המחברים הציעו ביסוס אמפירי לקשר בין **גודל ייצוג המצב לקיבולת המודל. לבסוף הם אימנו מודל Mamba2 בעל 370M פרמטרים עם אחזור מושלם של 256K טוקנים - הרבה מעבר ליכולות של מודל סטנדרטי מסוג זה.

המאמר היומי של מייק - 13.02.25
One Initialization to Rule them All: Fine-tuning via Explained Variance Adaptation

היום נסקור קצרות מאמר המציע שיטת LoRa לשיפור של טכניקת טיוב (fine-tuning) של LLMS. כמו שאתם בטח זוכרים LoRA מוסיפה למשקלי המודל (בשכבות מסוימות) מטריצה נלמדת בעלת ראנק משמעותית נמוך יותר מהמימד של מטריצת המשקולות. משקולות המודל נשארות קבועות (לא מאומנות) במהלך הטיוב.

המחברים כאומר מציע גישה לשכלול של LoRA המכיל שלב מקדים שנקרא במאמר אתחול Date-Driven. מטרת אתחול זה היא "להתאים את הראנק של מטריצות של LoRA לכל שכבה של המודל". הרי אם אנו מאמנים תוספת משקלים מסוימת (במהלך אנו יכולים לפזר אותם בצורה "אופטימלית" בין שכבות המודל. האופטימליות כאן נמדדת באמצעות השונות של האקטיבציות של השכבה (כלומר הפלט של שכבת FFN) עבור הדאטה שאנו מאמנים עליו.

הרי אם שונות האקטיבציות על דאטה האימון היא נמוכה זה אומר שערכי השכבה פחות או יותר קבועים ולא כדאי לבזבז עליה את המשקלים של LoRA. כלומר אפשר להשתמש ב-LoRA בעלת ראנק נמוך מאוד (אם בכלל) לשכבה זו. אבל איך ניתן למדוד את השונות הזו באמצעות ערכים סינגולריים של מטריצות האקטיציות המחושבים באמצעות פירוק SVD של מטריצה זו. מימדים מטריצת האקטיבציה כאן היא המימד החבוי של המודל וגודל הבאץ'.

אז מחשבים את הערכים הסינגולריים של מטריצת האקטיבציות על דאטהסט האימון עד שהוקטורים הסינגולריים (הימניים מתיצבים). וקטורים אלו מתעדכנים במהלך הרצות הבאצ'ים(המודל מתאמן) ויצירתם(של הוקטורים) נעצרת כאשר הם מתייצבים ומפסיקים להשתנות באופן מהותי (המאמר מודד את הדמיון באמצעות מרחק קוסיין - אם הוא גבוה מדי עבור שכבה מסוימת מפסיקים את עדכון הוקטורים עבור שכבה זו(אימון זה המתבצע לפני LoRA).

לאחר שהוקטורים הסינגולריים התכנסו עבור כל השכבות, לוקחים את הערכים העצמיים ומחשבים את אחוז השונות המוסבר על ידי כל שכבה (מחושב על ידי סכום הריבועים של הערכים הסינגולריים שלהם) ביחס לשונות המוסברת על ידי כל המודל (שהיא סכום הריבועים של הערכים הסינגולריים עבור אקטיבציות של כל שכבות המודל).

בשלב הבא מקצים את הראנקים של מטריצות LoRA לשכבות שפונקציות של השונות המוסברת על ידי. כלומר ככל השונות המוסברת של שכבה עולה, מקצים יותר ראנקים שלי LoRa. בשלב האחרון מאמנים LoRa עם הקצאה "אופטימלית" של ראנקי מטריצות LoRA בהתבסס על דאטה האימון. רעיון די מעניין שמראה תוצאות לא רעות.

https://arxiv.org/abs/2410.07170

המאמר היומי של מייק - 1502.25
A Spectral Condition for Feature Learning

1. מבוא

המאמר מציג מסגרת תיאורטית להבנת למידת מאפיינים(feature learning) ברשתות נוירונים עמוקות דרך חקר הסקאלת הנורמה הספקטרלית של משקולות ואקטיבציות הרשת. המחברים מציגים תנאים עבור סקאלה ספקטרלי השולט בהתפתחות המאפיינים המופקים על ידי הרשת במהלך האימון, ומספקים אסטרטגיה לבחירת סקאלות של משקולות וקצב למידה המבוססות על אינטואיציה בלבד.

המוטיבציה המרכזית של עבודה זו היא להתמודד עם אתגר מרכזי באימון רשתות רחבות (ועמוקות): הבטחת למידת מאפיינים אפקטיבית בכל השכבות, תוך מניעת דעיכת או התפוצצות הגרדיאנטים. המחברים טוענים כי באמצעות סקאלת נורמה ספקטרלית מדויקת של מטריצות המשקולות ועדכוניהן, ניתן לשמר למידת מאפיינים גם בגבול עבור רשתות בעלי מימדים חבויים מאוד גבוהים. מסגרת זו מספקת גישה מבוססת יותר(מתמטית) בהשוואה לאופן אתחול משקולות מסורתיות המבוססות על נורמת פרובניוס או בחירתם פר משקל (כמו .

המאמר תורם הן להיבטים התיאורטיים והן להיבטים הפרקטיים של אימון רשתות נוירונים בכך שהוא מדגים כיצד שיקולי נורמה ספקטרלית מובילים באופן טבעי לשיטה μP – Maximal Update Parametrization, אסטרטגיית אתחול וסקאלת קצב למידה המאפשרת העברת היפרפרמטרים ממודלים קטנים לרחבים. בשונה ממחקרים קודמים שהסיקו את μP באמצעות ניתוחים טנזוריית מורכבים, המאמר מספק הוכחה פשוטה יותר המבוססת על אלגברה לינארית, מה שהופך אותו לנגיש יותר עבור קהילת למידת העומק.

2. תרומות מרכזיות וייסוד תיאורטי

2.1 תנאי הסקאלה הספקטרלי

הממצא המרכזי של המאמר הוא תנאי סקאלה על הנורמה הספקטרלית של מטריצות המשקל ועדכוני הגרדיאנט שלהן:

$$||W_l||_* = \Theta(\sqrt{n_l/n_{l-1}}), \quad ||\Delta W_l||_* = \Theta(\sqrt{n_l/n_{l-1}})$$

כאשר {n_{l-1} ו- n_l מסמנים גודל הקלט והפלט (fan-out ו-fan-in) בשכבה l ו- * מסמן הנורמה הספקטרלית של W. תנאי זה חייב להתקיים עבור כל שכבות הרשת

תנאי זה מבטיח כי גם גודל הפיצ'רים החבויים h_l וגם עדכוניהם Δh_l (כתוצאה ממורד הגרדיאנט) יישארו בסקאלה מתאימה:

$$||h_l||_2 = \Theta(\sqrt{n_l}), \; ||\Delta h_l||_2 = \Theta(\sqrt{n_l})$$

ובכך נמנעות הן דעיכה והן התפוצצות של מאפיינים, תוך שימור דינמיקת למידה יציבה לאורך כל שכבות הרשת.

המוטיבציה לתנאי זה נובעת מהאופן שבו מידע "זורם" ברשתות נוירונים. בשיטות אתחול מסורתיות כמו Kaiming או Xavier, נעשה שימוש בנורמת פרובניוס לשליטה בגודל האקטיבציות. אולם, המחברים טוענים כי דווקא הנורמה הספקטרלית – המודדת את הערך הסינגולרי הגדול ביותר של המטריצה – מספקת אינדיקציה מדויקת יותר להשפעת השכבות על אותות הקלט.

2.2 ביסוס מתמטי של למידת מאפיינים

תנאי הסקאלה הספקטרלי נגזר מתכונה יסודית של רשתות עמוקות: כל שכבה מבצעת טרנספורמציה המגבירה או מחלישה את אותות הקלט בהתאם לערכים הסינגולריים של מטריצת המשקל שלה. הערך הסינגולרי הגדול ביותר (הנורמה הספקטרלית) קובע עד כמה השכבה מסוגלת למתוח או לכווץ את האקטיבציות לאורך כיוונים מסוימים במרחב התכונות.

המאמר מוכיח כי כאשר הנורמה הספקטרלית מקיימת את תנאי הסקאלה שהוגדרו קודם, מתקיימים התנאים הבאים:

- עוצמת פיצ'רים נשמרת לאורך השכבות, מונעת דעיכה או התפוצצות.
- התפתחות המאפיינים במהלך האימון נשארת משמעותית, ומונעת קריסה לייצוגים טריוויאליים.

להוכחת טענה זו, המחברים מבצעים ניתוח מתמטי מעמיק של עדכוני הגרדיאנט ב-MLPs (שזה multi-layer perceptron). נקודת מפתח היא שעדכוני המשקולות ברשתות עמוקות הם בעלי רנאק נמוך הנובע מהיות הגרדיאנטים מכפלה חיצונית (outer product) של וקטורים:

$$\Delta W_l = -\eta_l \Delta_{W_l} L = -\eta_l \cdot (\text{error signal}) \cdot (\text{input features})^T$$

מבנה זה מוביל לתובנה חשובה: עדכוני המשקל מתיישרים באופן טבעי עם הווקטורים הסינגולריים הדומיננטיים של מטריצות המשקולות, מה שמדגיש את חשיבות הנורמה הספקטרלית בקביעת דינמיקת הרשת.

2.3 קשר לשיטת פרמטריזציית μP

אחת התרומות המרכזיות של המאמר היא החיבור לשיטת μP. פרמטריזציה זו קובעת כללי אתחול וסקאלת קצב למידה המאפשרים העברת היפרפרמטרים ממודלים צרים לרחבים מבלי לדרוש כיול מחדש. המאמר מוכיח כי μP שקולה ליישום תנאי הסקאלה הספקטרלי, עם סקאלות אתחול וקצב למידה מהצורה:

$$\sigma_\ell = \Theta\left(\frac{1}{\sqrt{n_{\ell-1}}}\min\left\{1, \sqrt{\frac{n_\ell}{n_{\ell-1}}}\right\}\right), \quad \eta_\ell = \Theta\left(\frac{n_\ell}{n_{\ell-1}}\right)$$

כלומר, במקום להשתמש בחוקים מבוססי אינטואיציה, ניתן לגזור את μP מתוך שיקולי נורמה ספקטרלית. יתרה מכך, המחברים מציעים גישה מאוחדת שאינה מצריכה כללים מיוחדים לשכבות קלט, חבויות או פלט, ובכך מפשטים את היישום של μP.

3. מסקנות

המאמר מספק תובנות מעניינות בנושא למידת מאפיינים ברשתות רחבות באמצעות ניתוח נורמה ספקטרלית. התנאי הספקטרלי שהוצג מספק מסגרת מאוחדת המסבירה ומשפרת פרמטריזציות קיימות כמו μP. המחקר מצביע על כך שסכמות אתחול מסורתיות עשויות להפיק תועלת משמעותית מהסתמכות על נורמה ספקטרלית, דבר שעשוי לשפר את יציבות האימון ואת הביצועים של רשתות נוירונים עמוקות.

https://arxiv.org/abs/2310.17813

המאמר היומי של מייק - 16.02.25

Representation Alignment for Generation: Training Diffusion Transformers is Easier than you Think

לוקחים פסק זמן קטן מ-LLMs וסוקרים מאמר על מודלי דיפוזיה גנרטיביים. המאמר מציע שיטה די אינטואיטיבית לשיפור ביצועים של מודלים אלו על ידי הוספת איבר רגולריזציה ה"מיישר" את הייצוגים הפנימיים של המודל עם אלו של אנקודרים חזקים כמו DiNOV2. יישור זה משפר את איכות התמונות שהמודל מגנרט.

נתחיל מרקע קצרצר על מודלי דיפוזיה גנרטיביים. מודלים אלו מאומנים לגנרט תמונות (למשל בהינתן תיאור טקסטואלי) על ידי הסרה הדרגתית של הרעש. המודל מתחיל מרעש טהור (בד"כ גאוסי) ולאט לאט הופכים אותו לתמונה (או פיסת דאטה מדומיין אחר). המודל מאומן על תמונות מורעשות עם רמות שונות של רעש(=איטרציות) כאשר באימון המודל לומד להסיר כמות קטנה של רעש (מאיטרציה t לאיטרציה t -1). בחירה של הייפר-הפרמטרים α_t של תהליך ההרעשה היא מרכיב קריטי לאיכות גנרוט של המודל המאומן.

תהליך זה(הרעשה) ניתן לתאר באמצעות משוואות דיפרנציאלית של זרימה הסתברותית (probability flow) המתאר השתנות (גרדיאנט) הדאטה המורעש עם קצב/מהירות הרעשה (velocity) שנסמן אותו v_t (הפתרון של משוואה זו מפולג לפי ההתפלגות של הדאטה המורעש). קצב הרעשה ניתן לשערך עם המודל (=רשת) בהתבסס

על דגימות הדאטה המורעש ו-α_t. לאחר מכן ניתן לפתור את משוואות הזרימה ההסתברותית עם השערך של v_t (בכיוון ההפוך - כלומר החל מרעש טהור) עם שיטת איולר למשל. שיטות אלו נקראות stochastic interpoland. נציין שיש שיטות המבוססות על פתרון נומרי של משוואה דיפרנצאלית סטוכסטית שמתארת את השתנות הדאטה כפונקציה של פונקציית score שהיא לוגריתם של פונקציית התפלגות של דאטה מורעש.

אוקיי, אחרי הסיבוך הזה החיים נהיים קצת יותר קלים. מודלי דיפוזיה היום הם לרוב מודלים לטנטים מהגנרוט מתרחש במרחב הייצוג של הדאטה. כלומר המודל מאומן לשחזר ייצוג לטנטי מרעש ואז מפעילים את הדקודר כדי לבנות תמונה מהייצוג המשוחזר. הייצוג של התמונה ההתחלתית נוצר על ידי האנקודר. המחברים טוענים שהייצוגים הלטנטיים המורעשים אינם "חזקים מספיק" כלומר פחות משקפים את האספקטים הסמנטיים של התמונה.

המחברים מציעים להעשיר את הייצוגים האלו על ידי הוספה של איבר רגולריזציה המטרתו לקרב ייצוגים אלה (של התמונת המרועשות) לייצוג המופק על ידי אנקודר חזק (כמו DINOV2). לוס זה מתווסף ללוס הרגיל של מודל דיפוזיה והטענה שזה משפר את איכות התמונות המגונרטות וגם תורם ליציבות האימון.

https://arxiv.org/abs/2410.06940