המאמר היומי של אביב ומייק: 20.06.25  
Evolving Deeper LLM Thinking

המאמר מציג שיטה לשיפור ביצועי מודלי שפה בזמן אינפרנס(test-time compute). השיטה ממירה את בעיית החיפוש במרחב של פתרונות טקסטואליים לתהליך אבולוציוני מונחה־ביקורת, שנבנה כולו סביב יכולות הגנרטיבית וה"רפלקטיבי״ של ה-LLM עצמו. אין כאן fine-tuning או עדכון משקולות המודל אלא השיפור מתבצע באופק החישובי של ה-inference בלבד.

הנחת המוצא היא שבעיות רבות, כגון תכנון מסלול טיול או לוח זמנים לפגישות, לא ניתנות לפורמליזציה מלאה, אך כן ניתן לבחון עבורן את איכות הפתרון בעזרת פונקציית הערכה חיצונית. זה יוצר תרחיש שבו לא ניתן לייצר פתרונות ישירות על ידי אופטימיזציה מסורתית, אך כן ניתן לבצע חיפוש מונחה־הערכה. המאמר מבצע זאת באמצעות מנגנון גנטי שמתממש כולו בשפה טבעית.

### רכיב האלגוריתם 1: אוכלוסייה טקסטואלית

כל פתרון מיוצג כטקסט כלומר תיאור מילולי של תוכנית פעולה. המרחב שבו מתבצע החיפוש אינו מרחב וקטורי ואינו בעל מבנה טופולוגי ברור. אין מרחק מוגדר בין שני פתרונות, ואין דרך לקבוע "כיוון שיפור". השיפור מבוצע באמצעות recombination(מושג מהמאמר) לשוני, כלומר כתיבה מחדש של טקסט על בסיס טקסטים קודמים.

#### רכיב האלגוריתם 2: מבנה אבולוציוני עם איים

במקום אוכלוסייה אחת, האלגוריתם מחלק את מרחב הפתרונות למספר אוכלוסיות נפרדות הנקראים איים במאמר. כל אי עובר תהליך אבולוציוני עצמאי, אך כל כמה איטרציות מתבצעת "הגירה" של פתרונות מוצלחים בין האיים. כך נשמר איזון בין חיפוש מקומי (exploitation) לחיפוש גלובלי (exploration).

רכיב האלגוריתם 3: בחירה מבוססת סלקציה רכה

הבחירה של אילו פתרונות ישמשו הבסיס(הורים) לדור הבא אינה דטרמיניסטית. האלגוריתם בוחר פתרונות עם הסתברות שתלויה באיכותם, אך משמר גם סיכוי לבחירת פתרונות בינוניים, כדי למנוע התכנסות מוקדמת. זה יוצר מנגנון של סלקציה רכה שמאפשר לאוכלוסייה לשמור על גיוון מבני ורעיוני(קצת דומה לMCTS אבל בלי עצים).

#### רכיב האלגוריתם 4: recombination באמצעות שיח ביקורתי

במקום לבצע recombination באמצעות תהליכים סינתטיים כמו דילוג על שורות או חיבור משפטים, האלגוריתם מייצר שיח פנימי בין שני ישויות קונספטואליות, מבקר ומחבר, אשר לומדים מהפידבק של פונקציית ההערכה. התוצאה היא טקסט חדש, שלא בהכרח בנוי כשילוב כלשהו של פתרונות קודמים, אלא כפרשנות מחודשת עליהם. תהליך זה חוזר על עצמו מספר פעמים בכל דור.

התהליך כולו מסתמך על פונקציית הערכה חיצונית שיכול להיות קוד, תוכנה או מודל נוסף שמספקת גם ציון איכות וגם פידבק טקסטואלי מפורש. חשוב להדגיש: המשוב אינו בהכרח מספרי בלבד, אלא יכול לכלול תיאור מפורט של תקלות או סטיות מהאילוצים, מה שמאפשר למודל להשתמש בו כחומר גלם לרפלקסיה.

### יתרונות מבניים

סקלביליות לבעיות לא מוגדרות היטב: מאחר והאלגוריתם פועל על טקסטים ולא על מבנים פורמליים, ניתן להפעילו גם כאשר אין תיאור פורמלי של הבעיה.

הפרדה בין גנרוט לאבלואציה: בניגוד לגישות המבוססות על התקדמות ליניארית כמו Chain-of-Thought או Reflexion, כאן יש חלוקה ברורה: המודל מייצר, ההערכה בוחנת, ואז מתבצע רה-קונפיגורציה של הפתרון.  
מניעת התכנסות מוקדמת: בזכות האיים, ההגרלות הרכות, וה-reset התקופתי, נמנעת קריסה מוקדמת לפתרונות לוקליים.

האלגוריתם מאפשר ל-LLMs לחשוב לעומק לא דרך ניתוח סמנטי או לוגי של השפה, אלא דרך דינמיקה של תחרות, ביקורת, רפלקסיה והתמרה. זהו תהליך חישובי שמשתמש בשפה עצמה כחומר גלם לבניית פתרונות, ומוביל לשיפור איכותי של היכולות התכנוניות של המודל גם במצבים שבהם לא ניתן להגדיר מראש את מהות "הפתרון הנכון".

אם נביט בזה כתשתית רעיונית, המאמר מציע גישה כללית ל-meta-reasoning של מודלים: מערכת שמארגנת את החשיבה של המודל לא רק דרך פרומפט אלא דרך שילוב של רעיונות מתחרים שמתפתחים בהכוונת ביקורת. מדובר בתפיסה לא ליניארית של אינפרנס, כזו שמניחה שמחשבה טובה נולדת לא בבת אחת, אלא דרך אקספלורציה, שגיאות, ותיקון מצטבר.

https://arxiv.org/abs/2501.09891