大規模言語モデルに対するプロービングによる複合動詞の意味理解の分析

小野聡　河原大輔
早稲田大学理工学術院
s.ono@toki.waseda.jp　dkw@waseda.jp

概要

本研究は，大規模言語モデル (LLM) が日本語複合動詞の意味関係および成立可否をどの程度捉えているかをプロービングにより検証する．(A) 複合動詞の意味による分類，(B) 複合動詞として妥当であるかの判定という 2 つのタスクで評価を行う．評価は，凍結 LLM の層別に線形分類器を学習し，複合動詞を一まとまりとして入力する条件と，それぞれを分割して表現を統合する条件を比較する．結果として，いずれのタスクでも一まとまりでの入力が分割入力を上回り，複合動詞の理解には構成動詞の関係性が寄与する可能性が示唆された．さらに，日本語の大規模コーパスで学習された LLM ではこの差が大きい傾向が見られた．

1　はじめに

日本語には，「走り回る」や「書き記す」のように，二つの動詞が結合して一語の述語として振る舞う**複合動詞**が存在する．本論文では複合動詞の前要素を V_1，後要素を V_2 と記す．V_1–V_2 の結合は任意ではなく，「立ち食う」や「走り転ぶ」のように一般に用いられない組合せもある．これは，複合動詞の解釈が V_1 と V_2 の単純な合成のみではなく，結合形式そのものに対しても依存しうるためである．

この点を「形式と意味の対応」という**コンストラクション**（特定の形式が特定の意味・機能と結び付いたペア）の観点から捉える立場では，V_1–V_2 の形式と意味関係の対応が整理される．陳らはこの観点に基づき，複合動詞を意味関係で分類・体系化した [1]．例えば「連れ出す」は「連れる」を手段，「出す」を目的とする**手段-目的**に分類される．

大規模言語モデル（LLM）は，大規模コーパスから日本語の語彙・構文・意味に関する知識を獲得している．一方で，複合動詞の意味関係が LLM の知識にどの程度反映されているかは明確でない．本研究は，LLM が複合動詞の意味関係をどの程度捉えているかを検証し，複合動詞を一まとまりとして扱うことが，V_1 と V_2 を分割して扱う場合に比べて意味関係の識別に有利かを問う．LLM が成分の意味に加えて結合形式に関わる手掛かりを利用しているなら，一まとまりで入力した条件がより高い性能を示すと予想される．

本研究では，学習済み LLM を凍結し，内部表現から言語情報がどの程度読み出せるかを評価するプロービングにより分析する．具体的には，(a) 複合動詞を一単位として入力する条件と，(b) V_1 と V_2 に分割して入力する条件で意味関係分類の精度を比較し，あわせて層別の表現から意味関係が読み出されやすい傾向も検討する．

2　関連研究

日本語複合動詞の分類と成立条件・意味解釈　日本語の複合動詞（V_1–V_2）は二動詞が結合して一述語となるが，その結合は任意ではない．影山や由本により，クオリア構造を用いて，「結果」や「目的」，「原因」のような背景知識や動詞の概念との関連事象を含めた説明がなされた [2, 3]．クオリア構造とは，ある語彙項目の意味を最もよく説明できる，その語彙項目と関連する属性や事象の集合である．

限界とコンストラクションによる整理　しかし，$V1$ と $V2$ の意味を単純に合成するだけでは説明しにくい複合動詞もあり，複合動詞の意味には，従来の合成的アプローチでは説明できない**全体的な性質**があるという指摘がある．陳らはそれらを解決するために，複合動詞の一部に全体的な側面があるとし，合成的立場と全体的立場の双方を説明するために，コンストラクション形態論を用い，複合動詞を分類した [1]．本研究は，陳らの意味関係体系を評価軸として，LLM 内部表現からその情報がどの程度読

表 1　複合動詞の意味分類（説明・例・件数）

分類ラベル	説明	例	件数
a. 原因—結果	V1 が原因となり V2 が生起する.	生き残る	291
b. 手段—目的	V1 が V2 の目標達成の手段を表す.	投げ入れる	1291
c. 準備—目的	V1 の成立に近接して V2 が成立する.	割り入れる	29
d. 背景—実現	V2 の背景として V1 がある.	売れ残る	74
e. 様態—移動	V2 の様子／状態を V1 が示す.	滑り降りる	198
f. 付帯事象—主事象	V1 と V2 の別事象が時間的に共起する.	迎え撃つ	295
g. 並列関係	V1 と V2 が同一事象を表す.	飛び跳ねる	55
h. 事象対象	V1 が V2 事象を参照対象に取る.	出し惜しむ	38
i. 比喩的様態	一方が他方の比喩的様態を表す.	咲き狂う	64
j.V1 希薄化	V1 の意味が希薄化している.	ぶっ壊れる	132
k.V2 補助動詞化	V2 が V1 に要素を付加／補助する.	こね回す	669
l. その他	上記に当てはまらない.	出歩く	147
m. 不透明	意味構造が判別しにくい.	ひっつく	81
n. 派生	V1 と V2 の主語が異なる.	濡れ広がる	121
o. 不適当	複合動詞でない（非複合動詞）.	悩み透ける	949
		合計	**4434**

み出せるかをプロービングで検証する.

LLM 内部表現とプロービング　プロービングは，事前学習済みモデルを**凍結**したまま中間表現を特徴量として取り出し，補助タスクを解く分類器（プローブ）を学習することで，内部表現からどの情報が**読み出し可能**かを測る枠組みである [4, 5]. Transformer 系モデルに対しては，層ごとに得られる表現に線形分類器などを当てる層別プロービングにより，言語情報がどの層に分布するかを検証する研究が進展している [6, 7].

本研究では，凍結 LLM の層別表現に**線形プローブ**を適用して，複合動詞の意味関係がどの程度読み出しやすいかを比較する.

3　データセットとタスク

本研究では，(A) LLM が複合動詞 V_1–V_2 の意味関係ラベルを内部表現に保持しているか（タスク A）と，(B) V_1–V_2 が複合動詞として妥当かを識別できるか（タスク B）を分けて評価する.

これらの評価に，陳らの『日本語語彙的複合動詞の意味と体系』の複合動詞データセットを用いる. 本データセットは，複合動詞（V_1–V_2）に対して意味関係ラベルを付与したものであり，タスク A ではこのラベルを意味理解の評価対象とする. そのために負例（不適当）を除外し，意味関係ラベル（表 1 の a–n）の 14 クラス分類を行う.

タスク B のために，データセットに負例（不適当）を自動生成して付与する. 具体的には，V_1 を一様に選び，V_2 は既存データセットにおける V_2 の出現分布（経験分布）に従いサンプリングをし，候補の V_1–V_2 を作る. 既存データセットに含まれる組合せは棄却し，再サンプリングすることで，「既存にない組合せ」を負例として追加する. タスク B は，適当（複合動詞）／不適当（負例）の 2 値分類を行う.

表 1 に負例を加えた複合動詞データセットにおける意味分類とその例，各意味分類の件数を示す.

学習・評価のために，各分類ラベル内で V_2 の分布を保ちつつ train:test=7:3 に分割し，train のうち 30% を validation とした.

4　手法

本研究では複合動詞データセットを用い，凍結 LLM の層表現から意味関係がどの程度読み出せるかを線形プローブで評価する.

入力形式　凍結 LLM への入力は，(a) 複合動詞を一まとまりとして入力（CV (Compound Verb) 入力）する方法と，(b) 複合動詞を V_1 と V_2 に分割して独立に入力（独立入力）する方法とする. なお，(b) でのそれぞれの動詞の活用は，元の複合動詞に合わせ V_1 は連用形，V_2 は終止形とする.

内部表現抽出　(a)，(b) 双方の入力形式においても，トークナイズした後のトークン数 T と，各層 ℓ における 1 トークンあたりの埋め込みベクトルの次元数 d を用い，ℓ のトークン表現 $H^{(\ell)} \in \mathbb{R}^{T\times d}$ を取得する. 入力全トークン（特殊トークンを含む）に対して以下の式を用いて平均プーリングを行うこと

で，系列表現 $\mathbf{s}^{(\ell)} \in \mathbb{R}^d$ を得る：

$$\mathbf{s}^{(\ell)} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{h}_t^{(\ell)}.$$

なお，表現抽出は計算節約のために全奇数層 ℓ についてのみ行う．

線形プローブに渡す特徴量 $\mathbf{z}^{(\ell)}$ を以下の方法で作成する．CV 入力では複合動詞全体を 1 系列として上記の平均プーリングを行い，得られた $\mathbf{s}_{\mathrm{CV}}^{(\ell)}$ をそのまま $\mathbf{z}^{(\ell)}$ とする（$\mathbf{z}^{(\ell)} = \mathbf{s}_{\mathrm{CV}}^{(\ell)}$）．独立入力では，各層 ℓ において，同様に平均プーリングにより $\mathbf{s}_{V_1}^{(\ell)}, \mathbf{s}_{V_2}^{(\ell)}$ を得て，以下の 2 つの方法で $\mathbf{z}^{(\ell)}$ を構成する．

- **Concatenation** (Concat)：$\mathbf{z}^{(\ell)} = [\mathbf{s}_{V_1}^{(\ell)}; \mathbf{s}_{V_2}^{(\ell)}] \in \mathbb{R}^{2d}$．各ベクトル由来の特徴を区別できるが次元が増加する．
- **Element-wise addition** (EWA)：$\mathbf{z}^{(\ell)} = \mathbf{s}_{V_1}^{(\ell)} + \mathbf{s}_{V_2}^{(\ell)} \in \mathbb{R}^d$．次元は増えないが各ベクトルの値が混合し，相殺が起こりうる．

線形プローブと評価　各層・各入力条件で $\mathbf{z}^{(\ell)}$ からロジスティック回帰を学習する．多クラス分類では，validation の macro-F1 により重みを調整する．

5　実験

5.1　実験設定

使用モデル　複合動詞の意味関係に関するプロービング実験として，LLM-jp-3-instruct3 (980m，1.8b，3.7b，7.2b，13b)，Qwen3 (1.7b，8b)，Llama-3-Instruct (1b，8b)，Llama-3-Swallow-Instruct-v0.1 (8b) の 4 モデルを比較した．LLM-jp と Swallow は日本語能力を強化したモデル，Qwen3 は多言語（日本語も含む），Llama は英語で事前学習されたモデルである．

トークナイズ設定　入力テキストは各モデル付属のトークナイザを用いてトークナイズした．

ベースライン　モデルの内部表現を用いず，データセットの情報のみでロジスティック回帰を行うものを表層特徴ベースラインとし，以下の 4 種を用いる．

- **V1-only One-Hot**：複合動詞を（V_1, V_2）に分解し，V_1 の語彙サイズに等しい次元のベクトルを用意する．そして，観測された V_1 に対応する 1 次元のみを 1，それ以外を 0 とする One-Hot ベクトルに符号化する．
- **V2-only One-Hot**：V_2 のみを用いて上記と同様に One-Hot ベクトルに符号化する．

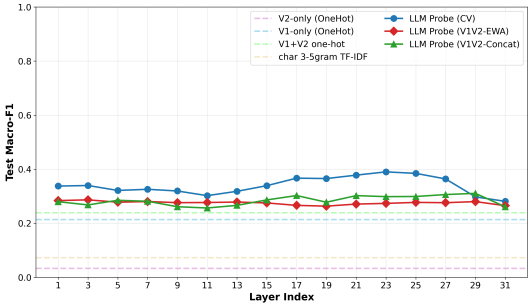

図 1　Llama 8B のタスク A の F1 スコア

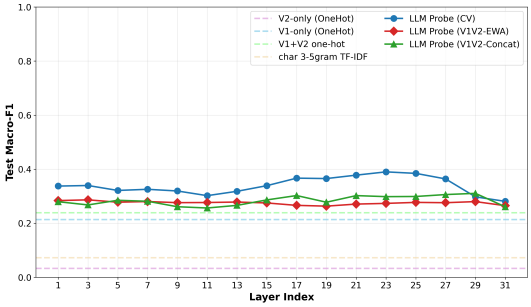

図 2　Llama 8B のタスク B の F1 スコア

- **V1–V2 One-Hot**：(V_1, V_2) の組を 1 つのカテゴリとして One-Hot ベクトルに符号化する．
- **文字 3–5gram TF-IDF**：表記（文字列）から連続する文字列（3, 4, 5 文字）を抽出し，それらを特徴として TF-IDF で重み付けする．

線形プローブ学習設定　また，タスク A/B ともにクラス不均衡に対応するために，各クラスの重みを「全サンプル数 ÷ (クラス数 × そのクラスのサンプル数)」とし，少数クラスの誤分類に大きなペナルティを付与する手法を適用した．

評価指標　評価は test セットで行い，クラス不均衡であるため macro-F1 を報告する．

5.2　結果

実験結果の一部を図 1 から 6 に示す．図中の横線は表層特徴ベースラインである．表層特徴ベースラインは層に依存しないため，横線として表示する．各図は層ごとの macro-F1 を示し，層番号が大きいほど出力層に近い．

タスク A：14 クラス複合動詞意味分類　図 1，3 より，タスク A では，線形プローブ結果は全モデルで表層特徴ベースラインを上回り，CV が Concat/EWA より安定して高かった．特に，Swallow の CV と Concat/EWA，ベースライン との差は 19 層目で 0.1 程度となった．性能は中〜後段層で最大となり，出力層近傍 (27 層) で低下した．

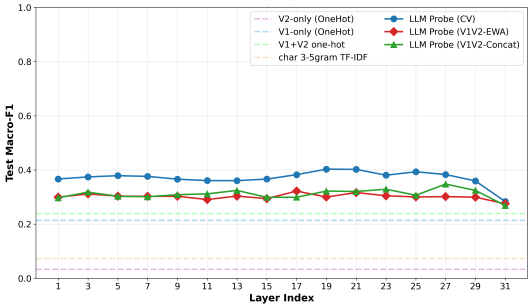

図 3　Swallow 8B のタスク A の F1 スコア

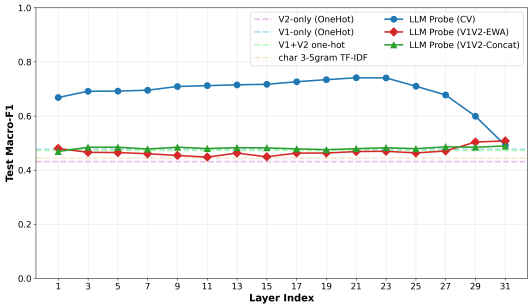

図 4　Swallow 8B のタスク B の F1 スコア

タスク B：存在する複合動詞かの分類　図 2，4 より，タスク B では線形プローブ結果はベースラインを大きく上回り，Swallow の CV と Concat/EWA，ベースライン との差は 23 層目で 0.25 程度となった．また，こちらでも出力層近傍 (25 層) で性能が低下し，性能は中〜後段層で最大となった．

モデルサイズによる比較　図 5，6 より，LLM-jp 系列では，CV 入力はモデルサイズの増加に伴い，タスク A では 0.06，タスク B では 0.15 ほど性能が上昇した．一方，Concat/EWA はモデルサイズに対して概ね横ばいであった．

5.3　考察

入力形式の差（CV vs 独立入力）　CV 入力が有利である理由は，複合動詞の意味が V_1 と V_2 の単純合成のみではなく，**コンストラクション**として成立するためと考えられる．$V1$ と $V2$ の関係が「一まとまり」の表現として LLM 内部に符号化されている可能性があり，CV 入力には成分間の相互作用や結合タイプに関する手掛かりが反映されたため，線形プローブが比較的容易に読み出せたと解釈できる．一方，Concat/EWA は成分を分割して統合するため，結合形式による情報を後段で補う必要がある．線形プローブではその統合を十分表現できず，CV 条件がより高い性能を示したと考えられる．

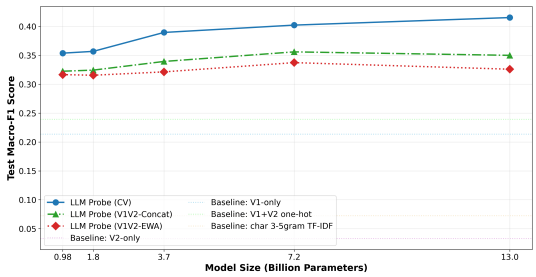

図 5　LLM-jp 各モデルサイズのタスク A の F1 スコア

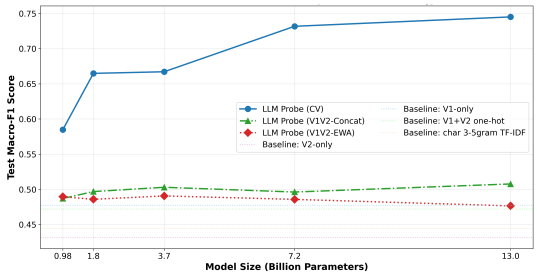

図 6　LLM-jp 各モデルサイズのタスク B の F1 スコア

日本語学習量と差分の関係　Llama を追加学習した Swallow で CV 入力の F1 が Llama より高いのは，**日本語での追加学習量**の差が大きいと考えられる．複合動詞の意味は V_1 と V_2 の単純合成だけではなく，V_1–V_2 のコンストラクションとしても成立するため，日本語コーパスから十分に学習しているほど，複合動詞を「まとまり」として表現しやすい．日本語コーパスへの露出が相対的に少ない Llama ではこの情報が CV 表現に入りにくく，線形プローブでの読み出し可能性が低下したと解釈できる．

最終層付近で性能が落ちる理由　最終層付近では生成・指示追従に適した表現へ変換が進み，分類に有効な汎用特徴が弱まる可能性がある．その結果，線形プローブでの分離性が低下し，中〜後段層より性能が下がったと考えられる．

6　結論

プロービング結果より，LLM は複合動詞の意味関係情報を内部表現にある程度保持していた．特に日本語に強く，大規模な日本語コーパスで学習した LLM では一定の精度に読み出せた．また，複合動詞表層を入力する条件が分割入力より安定して高く，多くのモデルで中〜後段層が最も有効であった．これらは，複合動詞のコンストラクションに関わる情報が LLM 表現に含まれることを示唆する．今後の課題として，より大規模な日本語事前学習コーパスによって学習した LLM により精度が向上するかを検証することなどが挙げられる．

謝辞

本研究は JSPS 科研費 JP23K25326，JP24H00727 および JST CREST JPMJCR2565 の支援を受けた．また，産総研及び AIST Solutions が提供する ABCI 3.0 を「ABCI 3.0 開発加速利用」の支援を受けて利用した。

参考文献

[1] 陳奕廷, 松本曜. 日本語語彙的複合動詞の意味と体系 : コンストラクション形態論とフレーム意味論, ひつじ研究叢書（言語編）, 第 152 巻. ひつじ書房, 東京, 2018.

[2] 影山太郎. 辞書的知識と語用論的知識—語彙概念構造とクオリア構造の融合にむけて. レキシコンフォーラム, No. 1, pp. 66–101, 2005.

[3] 由本陽子. 語彙的複合動詞の生産性と 2 つの動詞の意味関係. 影山太郎（編）, 複合動詞研究の最先端 : 謎の解明に向けて, pp. 109–140. ひつじ書房, 東京, 2013.

[4] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In **International Conference on Learning Representations (ICLR) Workshop Track**, 2017.

[5] Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)**, 2018.

[6] Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)**, 2019.

[7] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)**, 2019.

A　結果画像

Qwen，LLM-jp での実験結果を図 7–図 14 に示す．

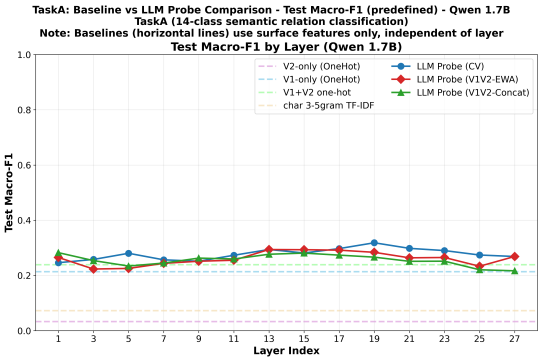

図 7　Qwen3-1.7B のタスク A の F1 スコア

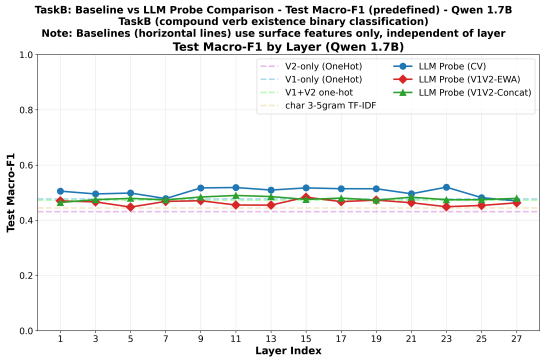

図 8　Qwen3-1.7B のタスク B の F1 スコア

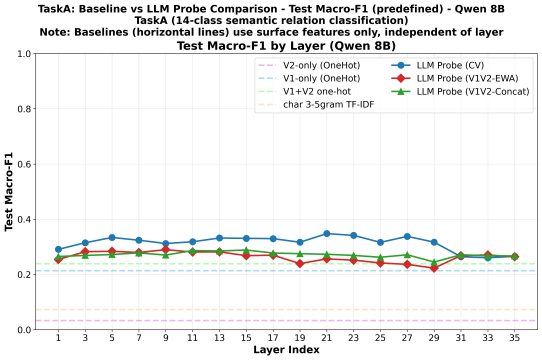

図 9　Qwen3-8B のタスク A の F1 スコア

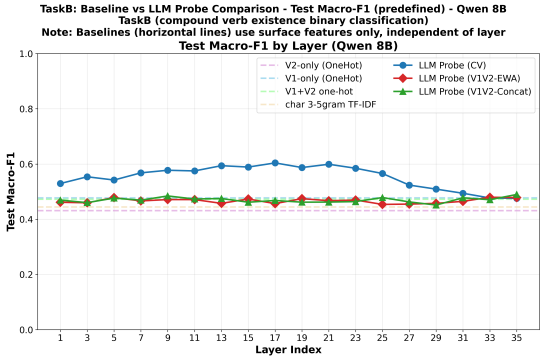

図 10　Qwen3-8B のタスク B の F1 スコア

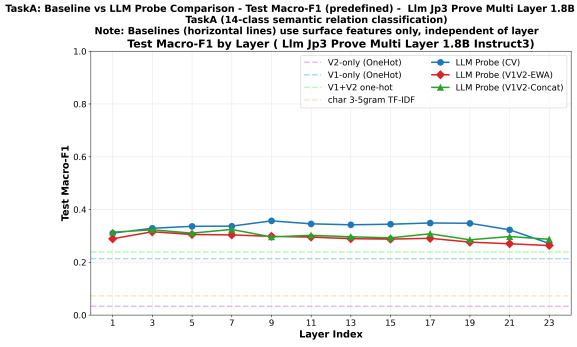

図 11　LLM-jp1.8B のタスク A の F1 スコア

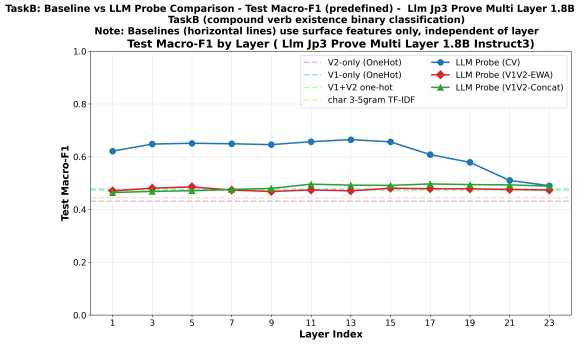

図 12　LLM-jp1.8B のタスク B の F1 スコア

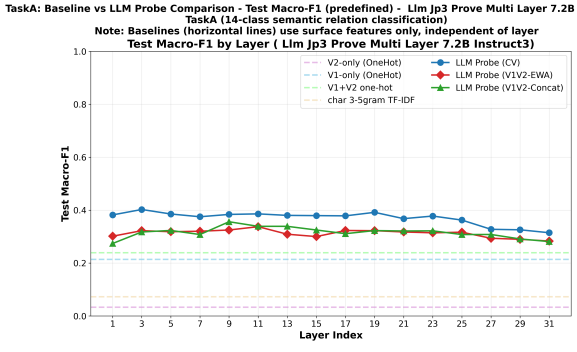

図 13　LLM-jp7.2B のタスク A の F1 スコア

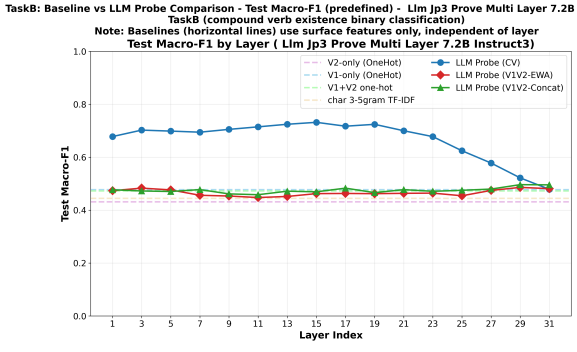

図 14　LLM-jp7.2B のタスク B の F1 スコア