訓練不要レビュー生成のための会話形式プロンプト

草野 元紀
日本電気株式会社
g-kusano@nec.com

概要

本研究では、ユーザの商品レビュー履歴が少なく学習も困難な環境において、高性能なレビューを生成することに焦点を当てる。既存手法は長い履歴データや教師あり学習を前提とするが、実環境では数件のレビューしか得られない状況が往々にして存在する。このような状況においては LLM の活用が有効であるが、そのとき、ユーザのレビュー履歴を単一のプロンプトで表現するのではなく、会話形式に再構成することが有効であることを 8 種類のデータセットと 5 種類の LLM で実証した。非会話型の単一のプロンプトは汎用的なレビューであり対象ユーザらしさが反映できてない一方、提案手法ではわずか 2 件しか過去履歴がなくとも対象ユーザに近いレビューを生成できている。

1 はじめに

ユーザの商品へのレビューは、ニーズ把握やマーケティング、商品開発における重要情報である。従来は人手によるアンケートやインタビューで商品レビューを収集してきたが、時間も費用もかさむという課題があるため、その課題を解消するためにレビュー生成の研究が行われてきた [1, 2, 3, 4, 5]。これらの多くは教師有り手法であるため、モデルを作る前に多くのユーザレビューが必要であり、レビューが集まらないという課題に再び直面する。

近年の大規模言語モデル（LLM）を用いると、訓練不要でも商品レビューを生成できることが知られている [4]。特定の人物になりきらせる Role-Play 技術 [6, 7, 8, 9] を用いると高精度なレビューが出来上がるが、そのユーザに対して多くのレビュー履歴など豊富なプロフィール情報 [8] が必要であり、再びデータ不足問題に直面する。本研究は、過去レビューが少ない一般消費者という、現実のマーケティングでよくある条件に焦点を当てる。

このような状況に対し、本研究では、会話形式プロンプトの活用を提案する。具体的には過去レビューにおける、アイテム情報を LLM における `user` ロールからの入力に、対応する過去レビューを `assistant` ロールからの応答として会話に変換する（図 1）。他ユーザのレビューがあれば、誤応答として挿入し、その不一致を指摘するメッセージを追加することで、正誤比較を通じてユーザの嗜好や文体を LLM に学習させる。他ユーザのレビューを負例として用いる手法を **Contrastive Conversational Prompting**（**CCP**）と呼ぶ。負例を用いないものは **Simple Conversational Prompting**（**SCP**）と呼び、CCP は外部レビューを要し、SCP は不要という違いがある。ただし、ここでの外部レビュー自体も LLM で作ることができるため訓練不要の状況は変えずに実行することができる。

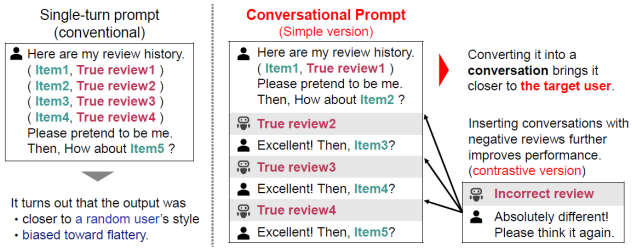

図 1 従来法（左）ではランダムなレビューに近い出力をしがちであった。提案法（右）はレビュー履歴を会話化することで、LLM が対象ユーザの文体を反映しやすくする。誤ったレビューの挿入で品質がさらに向上。

数値実験では、8 つの実世界データセットで **SCP** と **CCP** を評価し、履歴長 2〜10、5 種の LLM を比較した。結果として、非会話形式プロンプトはランダムなユーザレビューに近いことが多い一方、**SCP** と **CCP** は対象ユーザにより近いレビューを生成した。 **CCP** は一部で **SCP** を上回るが、負例収集コストとのトレードオフがある。そのため予算に応じて適切な手法を使い分けることができる。この研究の内容は [10] に基づくものである。

2 商品レビュー生成

問題設定 ユーザ u とアイテム i_k に対して、r_k^u を u が i_k に対して書いたレビュー、$\{(i_k, r_k^u)\}_{k=1}^n$ を u の時系列レビュー履歴（最新は i_n）とする。本研究の目的は、未レビューのアイテム i_{n+1} に対するレビュー $\hat{r}_{n+1}^u$ を生成し、u が実際に書くであろう内容に近づけることである。

会話形式プロンプト ここでは、LLM のプロンプトをマルチターン会話として構築する。メッセージは $M(role, text) := \{\text{'role'} : role, \text{'content'} : text\}$ とし、$role \in \{\mathsf{user}, \mathsf{assistant}\}$ が話者、$text$ が内容を表す。このようなメッセージの系列が会話形式プロンプトを形成する[1)]。

SCP レビュー履歴 $\{(i_k, r_k^u)\}_{k=1}^{n-\ell}$ に基づいてアイテム $i_{n-\ell+1}$ のレビュー生成を依頼するテキストを $T_{n-\ell}$（$0 \le \ell < n$）とする（Appendix の図 7 参照）。従来の in-context learning では過去レビュー n 件すべてを 1 メッセージ $[M(\mathsf{user}, T_n)]$ に連結させて推論するが、*Simple Conversational Prompting*（SCP）は同じ情報を ℓ 回の正のフィードバック会話に分割する。会話は $M(\mathsf{user}, T_{n-\ell})$ で開始し、assistant が実レビュー $r_{n-\ell+1}^u$ を提示、user が受理して次の $i_{n-\ell+2}$ へのレビューを依頼する。これを繰り返し、最後に対象アイテム i_{n+1} のレビューを求める。

CCP SCP は教師なしだが、アイテム i_k に他ユーザのレビューがあれば負例として利用できる。具体的には、user が i_k のレビューを求めた際、assistant は真の r_k^u ではなく他ユーザの r'_k（$r'_k \neq r_k^u$）を返し、user がそれを拒否して再生成を指示する。その後は SCP と同様に、assistant が正例 r_k^u を提示し、user が受理して次のアイテム i_{k+1} に進む（Appendix の図 7 参照）。負のフィードバックにより対象ユーザと他ユーザのレビューの差異を強調することを狙う対照学習 [11] に由来するめた、これを *Contrastive Conversational Prompting*（CCP）と呼ぶ。

3 数値実験

データセット 数値実験では Amazon Reviews Dataset [12] の 8 カテゴリを用いた。それぞれで、少なくとも 6 件のレビューを持つユーザと、他ユーザのレビューが 5 件以上あるアイテムを選定し、レビューは 20 から 300 トークンのものに制限した。各カテゴリからランダムに 200 ユーザ（計 1,600）を抽出し、最新レビューを評価用、残り 5 件を LLM への入力プロンプトに使用する。

品質評価には ROUGE-L [13] と BERTScore [14] を用いた。Appendix の表 2 に、同一アイテムに対する他ユーザのレビューとのスコアの統計量を示している。スコアの解釈として、ROUGE-L の約 0.12 は中央値約 60 人のランダムレビューに相当し、0.22 超で最も一致度の高い他ユーザのレビューに匹敵する。BERTScore はランダムレビューが約 0.84、最も似ているレビューが 0.87 であり、これらの絶対値を今後の結果の解釈に用いる。

比較手法 過去 n 件のレビューを単一会話 $[M(\mathsf{user}, T_n)]$ に連結するプロンプトを **Baseline** とする。これは Review-LLM [4] と同じである。

本実験では、CCP の負例の種類が性能に与える影響を比較するために、5 種類の負例を用意する。ROUGE または BERTScore が最大の他ユーザレビューを負例とするものを **CCP(R)** と **CCP(B)** とする。また、他ユーザレビューの質が低いものとして、対応する指標が最小となるものを **CCP(R)**$^-$ と **CCP(B)**$^-$ とする。そして、LLM 自身が負例を生成するケース **CCP(G)** を用意する。これは、$0 < \ell < n$ を設定し、初回は $[M(\mathsf{user}, T_{k-1})]$ から負例 r'_k（$k = n - \ell + 1$）を生成し、それ以降は、拒否フィードバック、正例レビュー r_k^u、受理メッセージ、次アイテム i_{k+1} へのレビュー生成依頼で会話を構成する。**CCP(G)** は実ユーザのレビューを要さないが推論コストが増える。

3.1 レビュー品質の自動評価

本節では、提案手法である SCP と CCP の有効性を 4 つの research question (RQ) に分けて検証する。

RQ1: 会話形式プロンプトの有効性 以降は LLM を `gpt-4.1-mini` とし、ユーザ履歴の最初以外にフィードバックを付与する設定（$\ell = 4$）でのスコアを表 1 に示す。分析結果としては次の通りとなった。(1) **Baseline** のスコア（ROUGE 0.12-0.15, BERTScore 0.843-0.852）は、表 2 におけるランダムな他ユーザのレビューに相当し、非会話形式プロンプトは一般的意見は捉えるが、対象ユーザの特徴は捉えられていないことが判明。(2) **SCP** と **CCP** は全データセットで **Baseline** を大幅に上回っており、会話形式プロンプトの有効性が確認された。(3) **CCP** の負例は性能を左右しており、高品質な他ユーザの

1) 話者の {user, assistant} は OpenAI や Anthropic によるものだが、Llama などの他の LLM では対応するものを使用。単一ターン $[M(user, text)]$ は通常のプロンプトに相当。

表 1　右端以外のすべての列は BERTScore（「Avg」は平均）を示し、右端は全データセットで平均した ROUGE-L を示す。各列で太字と下線は、それぞれ最良と次点の手法を表す。 **SCP** より有意に良いプロンプトには $*$ を付した（片側 Wilcoxon 検定、$p < 0.01$）。

	Movies	Music	Books	Kindle	Groceries	Games	Sports	Electronics	Avg	ROUGE
Baseline	0.844	0.847	0.852	0.852	0.846	0.850	0.848	0.843	0.848	0.133
SCP	0.854	0.859	0.864	0.860	0.854	0.860	0.857	0.850	0.857	0.154
CCP(B)	**0.860***	0.864*	0.866*	**0.863***	**0.857***	**0.863***	**0.858***	**0.853***	**0.861***	0.161*
CCP(R)	0.857*	0.863*	0.866*	0.863*	0.857*	0.861*	0.857	0.852*	0.860*	0.160*
CCP(G)	0.858*	**0.865***	**0.867***	0.863*	0.856	0.861	0.857	0.850	0.860*	**0.164***
CCP(B)$^-$	0.854	0.862	0.863	0.861	0.854	0.861	0.857	0.851*	0.858*	0.154
CCP(R)$^-$	0.857*	0.862	0.866*	0.862*	0.856*	0.861	0.857	0.851*	0.859*	0.157

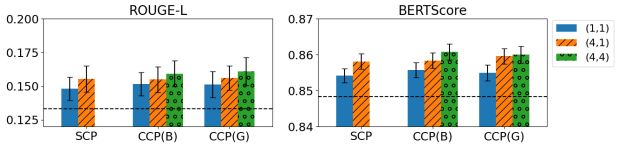

図 2　会話設定別スコア。エラーバーは t 分布に基づく 95% 信頼区間、破線は **Baseline**。

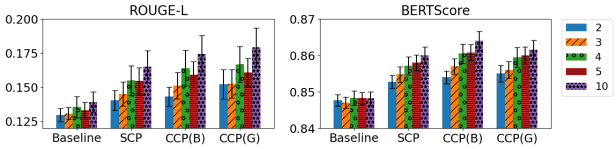

図 3　レビュー数別スコア。エラーバーは t 分布に基づく 95% 信頼区間。

レビューを用いると多くの場合で **SCP** を優位に超えており、低品質（$^-$）なものだと改善幅は小さくなった。これは決定境界近傍のハードネガティブが有効であるためだと考えられる。また、負例を選択する基準としては ROUGE より BERTScore が有利という結果になった。(4) **CCP(G)** は統計的に **SCP** を上回るケースが多く、性能は **CCP(B)** に近い。そのため、LLM が生成するレビューであっても高品質な負例として機能することが確認できた。

RQ2: 会話回数と負例数　RQ1 では $n = 5$ の履歴のうち、最新の $\ell = n - 1$ 件を `assistant` 応答として会話化し、残り 1 件を最初の指示に用いた。ここでは、会話回数 ℓ と負例数 m が精度に与える影響を、(4,4)：毎ターン負例の会話を用いる、(4,1)：毎ターン会話するが負例を用いるのは最後のみ、(1,1)：初回の指示に $n - 1$ 件のレビュー履歴を用いて、会話は最後の 1 回のみ、の 3 設定で比較する。

推論コストを抑えるため、以降は各データセットで 200 ユーザから 50 ユーザに縮小し、8 データセットを統合して計 400 ユーザで評価する。

図 2 よると、プロンプト種別や指標に依らず $(1,1) < (4,1) < (4,4)$ の順で精度が向上した[2)]。これにより各レビューを会話形式に変換し、各ターンで負例を挿入することが重要であるとわかった。**CCP** は負例収集に追加コスト（実レビューの取得または LLM 生成）が必要となるため、最小の追加コストである $(4,1)$ を導入したが、このスコアは $(1,1)$ である **SCP** と差が小さい。そのため、追加コストを許容できるなら $(\ell, m) = (n-1, n-1)$ が最も効果的で、コスト制約のある状況では $\ell = n - 1$ の **SCP** が性能と費用の良好なトレードオフを達成する。

RQ3: レビュー数　LLM は十分な履歴があるとユーザレビューの特徴を捉えられるが [8, 9]、実運用では多くのレビュー収集は難しい。そこで過去レビュー数を $n \in \{2, 3, 4, 5, 10\}$ に変化させて、生成されるレビューの品質がどう変化するかを評価する。

図 3 によると、**SCP** と **CCP** はレビュー数が増えるに従って精度が向上したが、**Baseline** は増加による改善が小さかった。これは入力例が増えるほど性能が上がるという in-context learning のスケーリング則 [15] と反する結果であった。$n = 2$ でも **SCP** と **CCP** は ROUGE > 0.14, BERTScore > 0.85 を達成しランダムレビューを上回っている。つまり、1 回の会話と 1 つの負例だけでも品質が向上する。

RQ4: LLM による性能差　これまでは `gpt-4.1-mini` を用いたが、ここでは SCP や CCP が他 LLM でも安定的に機能するかを調べるために、`gpt-4.1`, `o4-mini`, `llama3.3-70b`, `claude-sonnet-4` を調査対象に加える。設定は RQ2 の $n = 5$ に戻す。

図 4 がその結果であり、全 LLM で **SCP** と **CCP** が **Baseline** を上回った。ただし、推論モデルである `o4-mini` は低スコアあり、overthinking [16, 17] の可能性があることから、以降の分析では除外する。

Baseline の中では `gpt-4.1` が最良である一方、

2)　SCP は負例数 m に依存しないため、$(4,4)$ と $(4,1)$ は同スコア。図では $(4,4)$ を省略。

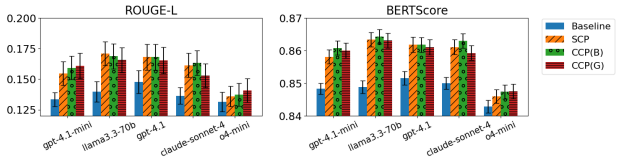

図 4　異なる LLM におけるスコア。エラーバーは t 分布に基づく 95% 信頼区間。

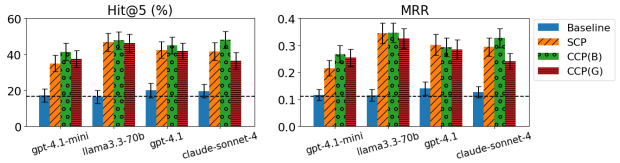

図 5　Hit@5 と MRR。エラーバーはブートストラップによる 95% 信頼区間、破線はランダムのスコア。

SCP は `o4-mini` 以外の全 LLM でより高スコアを示し、会話形式が LLM を問わず性能を向上させることを示した。**CCP** は、**CCP(B)** が多くの場合で **SCP** を上回り、**CCP(G)** は一部で劣ることがあるが差は小さく信頼区間も重なる。負例が用意できる場合は **CCP(B)**、難しい場合は **SCP** を使うことが良いという結果になった。LLM 間の比較では `llama3.3-70b` が概して高いが、信頼区間は他モデルと重なり、会話形式プロンプトの効果は各 LLM で安定している。

3.2　下流タスクでの評価

本研究ではレビュー生成の応用に焦点を当て、二つの下流タスクで有用かを検証する。

ユーザ同一性判定　生成レビューが対象ユーザらしさをどれだけ反映するかを測るため、User Indentity Linkage [18, 19] に着想を得たマッチングタスクを設計した。各アイテムについて、真のレビュー、生成レビュー、他ユーザのレビューを用意し、真のレビューとの BERTScore に基づいてランク付けし、生成レビューの順位を評価する。指標は Hit@5 と MRR（mean reciprocal rank）を用い、ブートストラップ法（1000 再標本）で信頼区間を算出した。比較として、他ユーザのレビューからのランダム選択を含める[3)]。

図 5 によると、**Baseline** は全 LLM でランダムと同程度で、非会話形式ではユーザらしさを捉えられていない。一方で、会話形式プロンプトはいずれも両指標を改善し、なかでも **CCP(B)** が一番良かった。この傾向は図 4 と整合的である。

感情分析タスク　生成レビューが対象ユーザの感情傾向をどの程度反映するかを、RoBERTa ベースの感情分類器 [20][4)] で評価する。分類器はテキストからポジティブ, ニュートラル, ネガティブの 3 次元ベクトルを出力する。各ユーザについて最大のラベルを集計してヒストグラムを作り、真の分布との類似度を KL ダイバージェンス [21] で測定した[5)]。

3)　各アイテムは他ユーザのレビューが多数（中央値 60、平均 285；表 2）のため Hit@5 を採用。

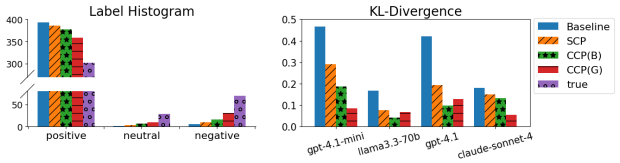

図 6　`gpt-4.1-mini` による感情ラベルのヒストグラムと KL ダイバージェンス。

図 6 によると、**Baseline** はポジティブに強く偏っている。対照的に会話形式プロンプトは批判的視点も含むよりバランスの取れたレビューを生成し、KL ダイバージェンスが小さく、ユーザの感情傾向をより偏りなく再現した。

4　考察

本研究は、少数事例かつ訓練不要の設定において、会話形式プロンプトがレビュー生成に有効であることを実証的に示した。しかし、理論的分析は本論文の範囲外であり、今後の課題である。近年、単一プロンプトにおける in-context learning のメカニズムが研究されている [22, 23, 24]。これらの知見を会話形式に適用することは有望な方向性である。

もう 1 つの方向性は推論コストである。精度向上の一案として、**CCP(G)** を改変して複数レビューを生成し、類似度が最も高いものを選択する方法が考えられる。いくつかの場合で **CCP(B)** が **CCP(G)** を上回ったため、この改変は性能改善に寄与する可能性がある。しかし、複数出力の生成はコストを増加させ、**SCP** の数十倍にもなりうる。大規模シミュレーション [25, 26] のように多くのユーザを処理する場面では、効率性が重要となる。負例としてウェブからアイテムレビューを検索することも一案だが、サイトの利用規約がこのような利用を制限する場合が多い。今後は、コストを抑えつつ精度を維持または改善できる会話形式プロンプトの探究を進める。

4)　https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest

5)　Amazon データセットの実評価分布（例：{5: 259, 4: 67, 3: 35, 2: 16, 1: 23}）は、分類器ラベルの傾向と概ね整合する。

参考文献

[1] Junyi Li, Wayne Xin Zhao, Ji-Rong Wen, and Yang Song. Generating long and informative reviews with aspect-aware coarse-to-fine decoding. In **ACL (1)**, pp. 1969–1979. ACL, 2019.

[2] Pan Li and Alexander Tuzhilin. Towards controllable and personalized review generation. In **EMNLP/IJCNLP (1)**, pp. 3235–3243. ACL, 2019.

[3] Jianmo Ni and Julian J. McAuley. Personalized review generation by expanding phrases and attending on aspect-aware representations. In **ACL (2)**, pp. 706–711. ACL, 2018.

[4] Qiyao Peng, Hongtao Liu, Hongyan Xu, Qing Yang, Minglai Shao, and Wenjun Wang. Review-LLM: Harnessing large language models for personalized review generation. **arXiv preprint**, Vol. arXiv:2407.07487, , 2024.

[5] Zhouhang Xie, Sameer Singh, Julian J. McAuley, and Bodhisattwa Prasad Majumder. Factual and informative review generation for explainable recommendation. In **AAAI**, pp. 13816–13824. AAAI Press, 2023.

[6] Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, Aili Chen, Nianqi Li, Lida Chen, Caiyu Hu, Siye Wu, Scott Ren, Ziquan Fu, and Yanghua Xiao. From persona to personalization: A survey on role-playing language agents. **TMLR**, pp. 2835–8856, 2024.

[7] Nuo Chen, Yan Wang, Haiyun Jiang, Deng Cai, Yuhan Li, Ziyang Chen, Longyue Wang, and Jia Li. Large language models meet harry potter: A dataset for aligning dialogue agents with characters. In **EMNLP (Findings)**, pp. 8506–8520. ACL, 2023.

[8] Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-LLM: A trainable agent for role-playing. In **EMNLP**, pp. 13153–13187. ACL, 2023.

[9] Noah Wang, Zhongyuan Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, Man Zhang, Zhaoxiang Zhang, Wanli Ouyang, Ke Xu, Wenhao Huang, Jie Fu, and Junran Peng. RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In **ACL (Findings)**, pp. 14743–14777. ACL, 2024.

[10] Genki Kusano. Few-shot and training-free review generation via conversational prompting. **arXiv**, Vol. abs/2509.20805, , 2025.

[11] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In **CVPR (2)**, pp. 1735–1742. IEEE Computer Society, 2006.

[12] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian J. McAuley. Bridging language and items for retrieval and recommendation. **arXiv preprint**, Vol. arXiv:2403.03952, , 2024.

[13] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In **Text Summarization Branches Out**, pp. 74–81. ACL, 2004.

[14] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In **ICLR**, 2020.

[15] Rishabh Agarwal, Avi Singh, Lei Zhang, Bernd Bohnet, Luis Rosias, Stephanie C. Y. Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, John D. Co-Reyes, Eric Chu, Feryal M. P. Behbahani, Aleksandra Faust, and Hugo Larochelle. Many-shot in-context learning. In **NeurIPS**, Vol. 37, pp. 76930–76966, 2024.

[16] Ryan Liu, Jiayi Geng, Addison J. Wu, Ilia Sucholutsky, Tania Lombrozo, and Thomas L. Griffiths. Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse. In **ICML**, 2025.

[17] Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models. **Trans. Mach. Learn. Res.**, Vol. 2025, , 2025.

[18] Kai Shu, Suhang Wang, Jiliang Tang, Reza Zafarani, and Huan Liu. User identity linkage across online social networks: A review. **SIGKDD Explor.**, Vol. 18, No. 2, pp. 5–17, 2016.

[19] Reza Zafarani and Huan Liu. Connecting corresponding identities across communities. In **ICWSM**, Vol. 3, pp. 354–357. The AAAI Press, 2009.

[20] Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In **EMNLP (Findings)**, pp. 1644–1650. ACL, 2020.

[21] Christopher M. Bishop. **Pattern recognition and machine learning**. Springer, 2006.

[22] Hakaze Cho, Mariko Kato, Yoshihiro Sakai, and Naoya Inoue. Revisiting in-context learning inference circuit in large language models. In **ICLR**, 2025.

[23] Benoit Dherin, Michael Munn, Hanna Mazzawi, Michael Wunder, and Javier Gonzalvo. Learning without training: The implicit dynamics of in-context learning. **arXiv preprint**, Vol. arXiv:2507.16003, , 2025.

[24] Hong Jun Jeon, Jason D. Lee, Qi Lei, and Benjamin Van Roy. An information-theoretic analysis of in-context learning. In **ICML 2024 Workshop on In-Context Learning**, 2024.

[25] Haoxiang Guan, Jiyan He, Liyang Fan, Zhenzhen Ren, Shaobin He, Xin Yu, Yuan Chen, Shuxin Zheng, Tie-Yan Liu, and Zhen Liu. Modeling earth-scale human-like societies with one billion agents. **arXiv preprint**, Vol. arXiv:2506.12078, , 2025.

[26] Joon Sung Park, Carolyn Q. Zou, Aaron Shaw, Benjamin Mako Hill, Carrie J. Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S. Bernstein. Generative agent simulations of 1,000 people. **arXiv preprint**, Vol. arXiv:2411.10109, , 2024.

```
input_conversational_prompt = [
    {"role" : "user", "content" : "You are an AI assistant. As an AI assistant, I want you to impersonate me. I will provide you with
    product information, and I would like you to generate a review that I might write. # My Information Here is my previous item
    information. The items are listed in chronological order from 1 (oldest) to 3 (latest). ## Item history 1 : {'itemInfo': Item1,
    'review': Review1}, 2 : {'itemInfo': Item2, 'review': Review2}, 3 : {'itemInfo': Item3, 'review': Review3} # Task Please predict
    the impressions I might have and generate a review for the target item. ## Target item. Item4  ## Output format Your
    response should only be the generated review. Do not include any irrelevant information."},  # instruction text
    {"role" : "assistant", "content" : Incorrect_review},
    {"role" : "user", "content" : "Absolutely different! That's not how I would answer. Please think it over carefully and generate a
    review for the target item that I might actually write."},  # rejection message
    {"role" : "assistant", "content" : Review4},
    {"role" : "user", "content" : "Excellent! It really feels like something I would write. Now, I will provide the next product. Please
    generate a review that I might write in the same way. ## Target item. Item5"}  # acceptance message
]

output = Generated_review  # output = LLM(input_conversational_prompt)
```

図 7　CCP の例。指示 T_3 に続き、誤レビュー r'_4、拒否、真のレビュー r''_4、次アイテム i_5 の依頼へと進む。

表 2　データセットの統計。最左列では「S」はスコア、「U」はユーザを表す。スコアについては、右端以外の列はすべて BERTScore（「Avg」は平均）を示し、右端は全データセットで平均した ROUGE-L を示す。ユーザについては、各アイテムにレビューを残したユーザ数を示す。

		Movies	Music	Books	Kindle	Groceries	Games	Sports	Electronics	Avg	ROUGE
S	Max	0.868	0.871	0.876	0.876	0.874	0.877	0.871	0.872	0.873	0.226
	Mean	0.841	0.845	0.849	0.848	0.844	0.849	0.846	0.842	0.845	0.121
	Min	0.805	0.810	0.816	0.815	0.788	0.804	0.803	0.783	0.803	0.036
U	Median	39	31	54	60	124	62	63	196	60	
	Mean	76.9	58.9	141.9	135.8	552.4	225.6	307.6	785.2	285.5	
	Std	106.6	81.1	244.5	195.3	1572.8	710.9	703.9	1475.3	880.7	