

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 ON THE ELIGIBILITY OF LLMS FOR COUNTERFACTUAL REASONING: A DECOMPOSITIONAL STUDY

Anonymous authors

Paper under double-blind review

ABSTRACT

Counterfactual reasoning has emerged as a crucial technique for generalizing the reasoning capabilities of large language models (LLMs). By generating and analyzing counterfactual scenarios, researchers can assess the adaptability and reliability of model decision-making. Although prior work has shown that LLMs often struggle with counterfactual reasoning, it remains unclear which factors most significantly impede their performance across different tasks and modalities. In this paper, we propose a decompositional strategy that breaks down the counterfactual generation from causality construction to the reasoning over counterfactual interventions. To support decompositional analysis, we investigate 11 datasets spanning diverse tasks, including natural language understanding, mathematics, programming, and vision-language tasks. Through extensive evaluations, we characterize LLM behavior across each decompositional stage and identify how modality type and intermediate reasoning influence performance. By establishing a structured framework for analyzing counterfactual reasoning, this work contributes to the development of more reliable LLM-based reasoning systems and informs future elicitation strategies.

1 INTRODUCTION

Large language models (LLMs) have exhibited remarkable proficiency across a diverse range of tasks, including natural language understanding (Devlin et al., 2019; Kuang et al., 2025) and multimodal reasoning (Hu et al., 2017; Lu et al., 2022; Yang et al., 2023). Despite these advancements, concerns persist regarding their reasoning and generalization capabilities. A particularly challenging aspect of model evaluation is **Counterfactual Reasoning**, i.e., the ability to adjust responses when presented with modified premises (Pearl & Mackenzie, 2018) (e.g., *What is the outcome in a hypothetical condition?*). Investigating the counterfactual reasoning of LLMs provides an interpretable step to understand their adaptability under hypothetical alterations to input conditions (Gat et al., 2024; Huang et al., 2024).

Prior studies have demonstrated that LLMs often struggle with counterfactual reasoning and frequently fail to maintain logical consistency or adjust to context shifts (Li et al., 2023; Nguyen et al., 2024; Wang et al., 2024). While these works highlight notable performance gaps, they lack a standardized framework for systematically analyzing and understanding counterfactual behaviors in LLMs. Consequently, it remains unclear what factors most significantly impact LLM performance in counterfactual scenarios. Furthermore, counterfactual reasoning has often been evaluated in a direct and monolithic manner, primarily by introducing interventions and assessing model responses (Li et al., 2023), without grounding the analysis in the underlying causal structure that gives rise to such interventions. This overlooks the foundational role of causal modeling. Specifically, the identification of causal variables and their dependencies are essential for understanding counterfactuals.

To address these gaps, we are motivated by the structural formulation of counterfactual reasoning under the Structural Causal Model (SCM) formalism (Pearl, 2009), in which counterfactual reasoning must proceed through a sequence of regularized steps including inferring latent variables from observations, modifying the SCM via intervention, and computing the updated outcome. Accordingly, we outline a **Decompositional Strategy** that breaks down the analysis of counterfactual reasoning into distinct stages. Our approach departs from prior work that focuses solely on counterfactual

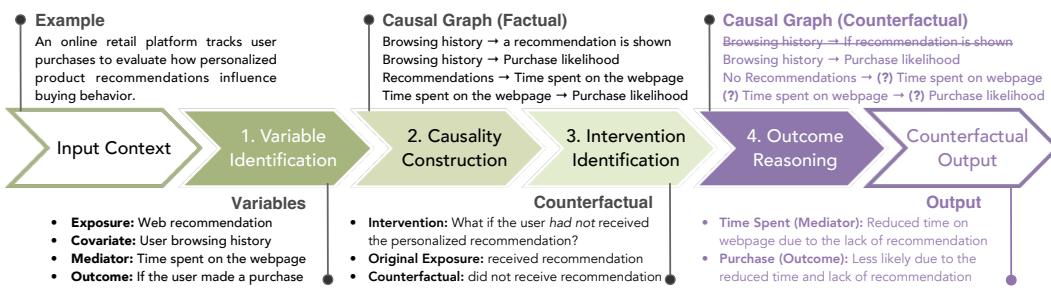


Figure 1: A workflow and illustrative example that decomposes LLM-based counterfactual reasoning into four stages: (1) identifying causal variables (e.g., *whether web recommendation is shown*), (2) constructing the causal graph (e.g., *browsing history → a recommendation is shown*), (3) specifying the counterfactual intervention (e.g., *no recommendation shown*), and (4) reasoning about the counterfactual outcome (e.g., *less likely to purchase a product online*).

generation. Instead, we begin by examining the causal structure of factual conditions, which serves as the necessary foundation for valid counterfactual reasoning.

As illustrated in Figure 1, our methodology is outlined into four stages. First, we assess (i) whether LLMs can accurately identify the four variable groups critical to causal reasoning: Exposure, Covariate, Mediator, and Outcome. Next, we evaluate (ii) whether LLMs can correctly construct a corresponding causal graph in the form of a directed acyclic graph (DAG). Building on this causality modeling, we then study LLMs’ counterfactual reasoning abilities by evaluating (iii) whether they can identify the correct intervened variable (i.e., the Exposure), and (iv) whether they can accurately infer the counterfactual mediators and final outcomes by reasoning over the updated causal graph.

To support our decompositional study, we construct a benchmark by collecting and curating 11 counterfactual datasets across diverse tasks, including natural language understanding, mathematics, programming, and vision-language reasoning. We curate each dataset by extracting factual and counterfactual variables, identifying causal elements, and constructing corresponding causal graphs as reference structures for evaluation purpose. In experiments, we test the performance of leading LLMs across each decompositional stage to analyze their sufficiency in handling individual reasoning components. Based on the observed performance among these decomposed evaluations, we propose targeted improvements, such as integrating modality-specific function-calling interfaces within a tool-augmented learning paradigm, to address critical reasoning bottlenecks. Additionally, we evaluate the impact of different elicitation (prompting) strategies, including Chain-of-Thought (CoT) (Wei et al., 2022), Chain-of-Thought with Self-Consistency (CoT-SC) (Wang et al., 2022), and Tree-of-Thought (ToT) (Yao et al., 2023) reasoning. Collectively, our evaluations provide a solid step for understanding and enhancing LLMs in complex reasoning tasks and imaginative scenarios 1.

In summary, we make the following contributions:

- Decompositional Framework**—We propose a decompositional strategy that spans from causal modeling to counterfactual reasoning, enabling a systematic evaluation of LLMs’ capabilities in understanding and performing counterfactual tasks.
- Benchmark Construction**—We construct a comprehensive evaluation benchmark by curating causal structures and counterfactual instances across multiple domains. This benchmark standardizes decompositional evaluations and supports consistent analysis across tasks and modalities.
- Evaluation and Improvement Strategy**—We evaluate leading LLMs under diverse tasks. By identifying LLMs’ capabilities in specific decompositional stage, we propose actionable strategies to improve LLMs’ counterfactual adaptability.

¹Codes available at: https://anonymous.4open.science/r/Counterfactual_NeurIPS-2025-D8E6/.

108
109

2 RELATED WORK

110
111
112
113
114
115
116
117
118

Counterfactual Reasoning. A fundamental component of causal inference is to examine hypothetical scenarios, which addresses the question: *What would have occurred had a particular factor or decision differed?* (Pearl & Mackenzie, 2018) This method facilitates causal analysis by comparing observed outcomes with those projected under alternative conditions. Empirical research has demonstrated the broad applicability of counterfactual reasoning across multiple domains, including healthcare, business, and fairness (Gvozdenović et al., 2021; Kyrimi et al., 2025; Gow et al., 2016; Kasirzadeh & Smart, 2021; Koonce et al., 2011). This stands in contrast to many contemporary AI systems, which predominantly rely on statistical correlations while lacking robust capacities for abstract reasoning and causal inference (Jiao et al., 2024).

119
120
121
122
123
124
125
126

Counterfactual Reasoning in AI and NLP. Counterfactual reasoning has emerged as a powerful framework for enhancing model interpretability and causal understanding in AI and NLP. In medical AI, SyncTwin (Qian et al., 2021) proposed a counterfactual estimation framework that constructs synthetic patient data to predict potential outcomes under alternative treatments. In NLP, a Counterfactual Reasoning Model (CRM) (Feng et al., 2021) is developed using LLMs to generate contrastive samples, improving sentiment analysis and inference tasks. There are also order-faithfulness metrics (Gat et al., 2024) to evaluate causal explanations in black-box models. These contributions demonstrate the versatility of counterfactual methods in improving model transparency and reliability across domains.

127
128
129
130
131
132

LLMs and Elicitation. Recent advances have significantly enhanced LLMs' reasoning capabilities through several elicitation (prompting) approaches. The introduction of Chain-of-Thought (CoT) prompting (Wei et al., 2022) and its extensions, the Self-Consistency CoT (Wang et al., 2022) and Tree-of-Thought (ToT) (Yao et al., 2023), have enabled more structured and reliable multi-step reasoning. These innovations collectively represent a paradigm shift from simple pattern recognition to deliberate, verifiable reasoning in LLMs.

133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161

Evaluation of Counterfactual Reasoning. Evaluating counterfactual reasoning in LLMs has garnered growing attention. However, most prior works assess counterfactual through end-to-end evaluations such as contrastive counterfactuals (e.g., “What would happen if X didn’t occur?”) (Huang et al., 2023; Frohberg & Binder, 2021; Zhang et al., 2024b; Le et al., 2023). Other studies such as MalAlgoQA (Sonkar et al., 2024) introduce the concept of algorithms to assess LLMs' ability to reason about flawed hypothetical paths. Their setup focuses on identifying distractor rationales in multiple-choice formats, revealing LLM struggles in understanding student misconceptions. Other efforts like DICE (Shrivastava & Aoyagi, 2025) and CausalProbe (Chi et al., 2024) create diagnostic benchmarks to evaluate causal sensitivity or counterfactual faithfulness in static question-answering formats, without decomposing the reasoning process into interpretable modules. Similarly, synthetic datasets have been used to analyze RNN inductive biases in agreement prediction (Ravfogel et al., 2019) but do not extend to structured counterfactual inference tasks. Compared to these works, our study introduces a modular evaluation aligned with Pearl's structural causal model (Pearl & Mackenzie, 2018). We decompose counterfactual reasoning into four interconnected sub-tasks to open up fine-grained attribution of model failures and provide a more diagnostic and interpretable assessment of LLM reasoning capabilities.

158
159
160
161

3 METHODOLOGY: DECOMPOSING COUNTERFACTUAL REASONING

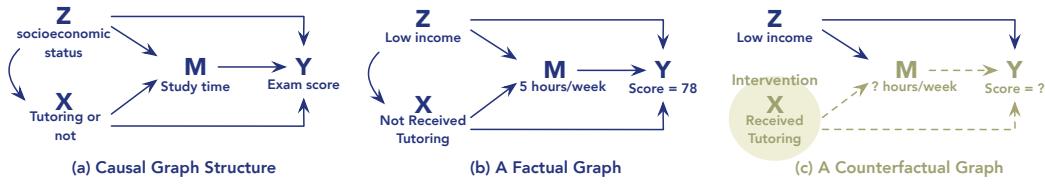
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1370
1371
1372
1373
1374
1375
1376
1377
1378
1

162 that may influence both X and Y ; (4) *Mediator* (M) lies on the causal pathway from X to Y ,
 163 representing intermediate mechanisms through which the exposure exerts its influence.
 164

165 **Example 1** Consider a dataset that records students' academic performance in the presence of a
 166 tutoring tool. Here, the exposure X indicates whether a student used the tool. The outcome Y
 167 corresponds to the student's final exam score. The covariate Z may include socioeconomic factors
 168 (e.g., parental income), which could influence both the use of tutoring and academic outcomes.
 169 The mediator M is the number of hours the student spends studying per week.
 170

171 **Causal Graph.** The relationships among exposure, covariate(s), mediator(s), and outcome(s) can
 172 be formally represented using a directed acyclic graph (DAG), commonly named a *causal graph*
 173 ([Pearl & Mackenzie, 2018](#)) that captures causal relationships, where the exposure X influences the
 174 outcome Y both directly and indirectly through a mediator M . Covariate Z may affect X , M , and
 175 Y , as illustrated in Figure 2.

176 **Example 2** The corresponding causal graph, illustrated in Figure 2-(a), would include: an
 177 arrow from $X \rightarrow M$ (e.g., tutoring influences study time), $M \rightarrow Y$ (study time influences
 178 exam performance), $X \rightarrow Y$ (direct effect of tutoring on scores), $Z \rightarrow X$ and $Z \rightarrow Y$ (e.g.,
 179 socioeconomic status affects both tutoring usage and academic performance).
 180



189 Figure 2: (a) Causal graph structure and (b)(c) A factual/counterfactual example.
 190

191 **Counterfactual Reasoning.** Counterfactual reasoning aims to answer:
 192

193 **Given an observed instance ($X = x, Z = z, M = m, Y = y$), what would the outcome Y be
 194 if the exposure X were set to a different value x' , while keeping the covariate Z fixed?**

195 The observed instance (x, z, m, y) is also known as the **factual** case. In contrast, counterfactual
 196 reasoning seeks to determine the outcomes under an alternate intervention, that is, when $X = x'$.
 197

198 In causal graph, we assume a two-stage causal mechanism: (1) Mediator Function: $M = f_M(X, Z)$
 199 and (2) Outcome Function: $Y = f_Y(X, M, Z)$. Then, the counterfactual outcome under an alternative
 200 exposure x' can be computed via: $Y_{x'} = f_Y(x', f_M(x', z), z)$, where we simulate a new mediator
 201 value $M_{x'} = f_M(x', z)$ based on the counterfactual exposure x' and observed covariate z . Then, we
 202 predict the counterfactual outcome $Y_{x'}$ using the counterfactual exposure x' , the simulated mediator
 203 $M_{x'}$, and the same covariate z .
 204

205 **Example 3** As exemplified in Figure 2-(b), a student with $z = \text{LOW-INCOME}$ socioeconomic
 206 status did not receive tutoring ($x = 0$), studied 5 hours per week ($m = 5$), and scored 78 on the
 207 exam ($y = 78$).

208 **We now ask:** *What would the student's score have been if they had received tutoring ($x' = 1$)?*

209 **We compute:** (i) Simulated study time: $m' = f_M(x' = 1, z = \text{LOW-INCOME}) = 9$ and (ii)
 210 counterfactual score: $Y_{x'=1} = f_Y(x' = 1, m' = 9, z = \text{LOW-INCOME}) = 85$.

211 **We conclude:** *The tutoring would have increased the student's score from 78 to 85 (Figure 2-(c)).*

212 3.2 DECOMPOSITIONAL EVALUATION TASK

213 Counterfactual reasoning is often described as a structured chain of analysis from identifying vari-
 214 ables to modeling causal relations, specifying interventions, and simulating outcomes ([Pearl, 2009](#);
 215 [Bareinboim et al., 2022](#)). Recent work in ML and LLMs further emphasizes the importance of

Table 1: Summary of counterfactual benchmarks including data source, use case, presence of causal variables (●: present, ○: partially present), the definition of counterfactual condition, included modalities, and number of instances. Concrete examples are shown in Appendix A.

Data	Use Case	Causal Variable				Counterfactual Condition	Modality	Num
		X	Z	M	Y			
CRASS [Frohberg & Binder, 2021]	Question answering	●	○	●	●	“What if ...” condition	Text	274
CLOMO [Huang et al., 2023]	Text logic parsing	●	●	●	●	New premise for textual statement	Text	1,100
RNN-Typology [Ravfogel et al., 2019]	Text syntax parsing	●	●	●	●	New syntactic structure of sentence	Text	584
CVQA-Bool [Zhang et al., 2024b]	Question answering	●	●	●	●	Hypothetical behavioral pattern	Text,Image	1,130
CVQA-Count [Zhang et al., 2024b]	Numerical reasoning	●	●	●	●	Hypothetical numerical pattern	Text,Image	2,011
COCO [Le et al., 2023]	Text-image matching	●	●	●	●	“What if ...” condition	Text,Image	17,410
Arithmetic [Wu et al., 2024]	Mathematical reasoning	●	●	●	●	Change number base	Symbol	6,000
MalAlgoQA [Sonkar et al., 2024]	Question Answering	●	●	●	●	“What if ...” condition	Text,Symbol	807
HumanEval-Exe [Chen et al., 2021]	Code Execution simulation	●	○	●	●	Hypothetical coding criterion	Text,Code	981
Open-Critic [Vezora, 2024]	Code generation	●	●	●	●	Hypothetical descriptive functions	Text,Code	8,910
Code-Preference [Vezora, 2024]	Code summarization	●	○	●	●	Hypothetical code structures	Text,Code	9,389

disentangling these steps to evaluate reasoning capacity (Kiciman et al., 2023; Chi et al., 2024). Motivated by the need for decomposition, we design four evaluation tasks that reflect the full pipeline of counterfactual reasoning, with each task targeting a distinct capability in counterfactual analysis.

- **Task I: Causal Variable Identification.** Given inputs containing factual information, a model is required to identify the values of the causal variables (X, Z, M, Y). This step serves as the foundation for subsequent causal modeling and counterfactual reasoning.
- **Task II: Causal Graph Construction.** Given the identified variables, the model is tasked with constructing a DAG that captures the causal relationships among them. This step evaluates the model's ability to discover causal dependencies.
- **Task III: Counterfactual Identification.** Given a counterfactual query (e.g., “*What if variable X had been different?*”), the LLM must identify the new value of (i.e., the intervention). This task evaluates whether the model can detect intervention in the counterfactual condition.
- **Task IV: Outcome Reasoning.** Based on the constructed causal graph and identified intervention, the model is prompted to predict the counterfactual outcome. This step measures whether the model can simulate the hypothetical scenario while respecting the underlying causal mechanisms.

3.3 BENCHMARKING COUNTERFACTUALS

Next, we introduce the datasets we leverage for the decompositional evaluations:

Data Sources and Use Cases. As shown in Table 1, we collect a diverse set of datasets to ensure broad coverage across various NLP tasks and modalities. The included use cases are: (1) **Question Answering**, evaluated using CRASS (Frohberg & Binder, 2021), CVQA-Bool (Zhang et al., 2024b), and MalAlgoQA (Sonkar et al., 2024), which involve answering general-purpose textual or visually grounded questions; (2) **Text Parsing**, using CLOMO (Huang et al., 2023) for logical structure reconstruction and RNN-Typolog (Ravfogel et al., 2019) for syntactic structure understanding; (3) **Reasoning Tasks**, with CVQA-Count (Zhang et al., 2024b) for numerical reasoning and Arithmetic (Wu et al., 2024) for symbolic arithmetic computation; (4) **Multimodal Matching**, represented by the COCO dataset (Le et al., 2023) for image-text alignment; (5) **Code-based Tasks**, including HumanEval-Exe (Chen et al., 2021) for execution simulation, Open-Critic (Vezora, 2024) for generation, and Code-Preference (Vezora, 2024) for summarization. These datasets are therefore intentionally positioned as a prerequisite stage to support safe and informed downstream application.

These datasets span four modalities, natural language text, images, mathematical symbols, and code, that encompass diverse definitions of counterfactual interventions tailored to each task. Collectively, they support a comprehensive multimodal evaluation of LLMs' abilities to reason under varied counterfactual settings and data types.

Our Preprocessing. To support our decompositional evaluations, we curate those datasets to augment each instance with three additional aspects of information relevant to Tasks I–III (Section 3.2). Specifically, we begin by identifying and annotating the causal variables (X, Z, M, Y) from the original data, questions, or descriptions. Using these annotations, we construct a DAG to represent

270
 271 Table 2: LLMs’ performance in causal variable identification, we report means of F1 across all
 272 instances for each variable. Each value is scaled to 100%. The standard deviation is in Table 8.

273 274 275 Dataset	276 GPT-5		277 GPT-o4		278 Qwen3		279 Llama4-S		280 Llama4-M		281 Gemini2.5		282 DeepSeek	
	283 v_1	284 v_2	285 v_1	286 v_2	287 v_1	288 v_2	289 v_1	290 v_2	291 v_1	292 v_2	293 v_1	294 v_2	295 v_1	296 v_2
$v_1 = X$ (Exposure), $v_2 = Z$ (Covariate)														
CRASS	92.3	91.1	91.0	89.2	87.3	85.4	88.5	86.9	90.6	89.1	88.6	86.2	89.5	87.1
CLOMO	89.8	87.6	88.1	85.6	83.5	81.9	87.1	85.3	88.8	87.0	84.3	82.8	86.4	84.2
RNN-Topo	87.9	85.4	85.9	83.7	80.4	78.6	84.3	82.6	85.7	84.2	81.7	80.1	83.8	82.0
CVQA-Bool	79.4	76.2	79.8	76.5	72.3	70.5	77.9	75.8	79.1	76.9	68.5	66.9	70.7	68.3
CVQA-Count	74.7	72.3	74.3	72.9	68.9	67.2	73.6	71.9	74.4	72.5	65.8	63.7	67.4	65.1
COCO	72.8	70.2	73.2	71.1	67.2	65.4	72.5	70.8	73.6	71.7	62.6	60.9	65.9	63.2
Arithmetic	88.2	86.5	84.9	82.8	75.7	73.8	80.3	78.6	81.6	79.8	76.9	74.5	78.3	76.1
MalAlgoQA	84.1	81.3	81.5	78.9	72.9	70.6	79.5	77.1	80.6	78.2	73.5	71.2	75.9	73.4
HumanEval-Exe	69.3	66.9	71.4	69.2	63.7	61.9	67.8	65.7	68.9	66.7	59.6	57.3	62.1	59.8
Open-Critic	71.7	69.4	70.1	67.3	61.8	59.7	66.5	64.7	67.6	65.8	57.3	55.9	60.4	58.1
Code-Preference	49.6	68.4	80.2	69.0	72.9	60.5	73.6	61.9	75.2	63.4	68.4	66.5	61.2	59.3
$v_1 = M$ (Mediator), $v_2 = Y$ (Outcome)														
CRASS	87.4	91.7	84.1	89.3	72.8	79.9	81.2	87.4	82.4	88.6	74.1	81.6	76.2	83.5
CLOMO	83.1	89.4	81.3	87.9	68.9	77.4	77.2	84.9	78.6	86.0	71.2	78.9	73.5	80.7
RNN-Topo	81.7	86.3	79.4	85.6	67.1	75.8	75.3	83.1	76.6	84.4	69.3	76.4	71.5	78.9
CVQA-Bool	73.5	79.3	73.9	80.1	59.7	66.9	71.8	78.3	72.9	79.4	57.3	63.2	60.5	68.4
CVQA-Count	69.6	75.2	70.1	76.2	56.8	63.6	68.9	74.8	70.2	75.9	54.7	60.4	57.9	65.1
COCO	67.3	73.4	68.0	74.4	54.2	61.8	66.2	72.1	67.4	73.5	52.8	58.3	55.7	62.6
Arithmetic	82.1	85.6	82.3	73.4	63.6	71.9	79.1	85.1	80.3	86.0	62.8	73.5	65.7	74.9
MalAlgoQA	79.2	83.4	76.8	80.4	61.5	69.3	76.6	81.2	77.8	82.4	60.2	70.5	63.8	72.3
HumanEval-Exe	63.2	67.4	66.0	70.3	51.9	59.7	61.9	66.3	63.0	67.5	49.7	57.0	53.6	60.5
Open-Critic	66.3	70.6	64.1	68.9	49.7	57.2	64.7	69.0	65.6	70.0	47.3	54.8	51.3	58.7
Code-Preference	65.9	75.3	66.3	77.0	50.4	78.6	64.5	79.3	65.7	80.4	48.2	76.1	52.5	59.8

293
 294 the underlying causal structure of each data instance, which enrich original instances with causal and
 295 counterfactual structures. A running example in Figure 2.

296
 297 **Preprocessing Feasibility.** Notably, all datasets are built upon “what-if” conditions or hypothetical
 298 scenarios (as outlined in Table 1) and intervention-style narratives, thus naturally supporting counter-
 299 factual interventions. For each instance, we parse and extract the intervened variables and, guided by
 300 the previously constructed DAG, annotate the corresponding counterfactual outcomes and construct a
 301 matched counterfactual graph. We provide the curated instances in Appendix A.

302 4 EXPERIMENT

303 We aim to empirically answer two research questions: **RQ₁**: How well do LLMs perform when
 304 their counterfactual reasoning is decomposed into distinct reasoning tasks? **RQ₂**: What auxiliary
 305 techniques can improve LLMs’ counterfactual reasoning? We defer the experimental settings into
 306 Appendix B.1 and additional results into Appendix B.2.

307 **LLMs.** We evaluate reasoning-centric and multimodal LLMs due to the nature of counterfactual
 308 reasoning tasks. Specifically, we leverage GPT-5, GPT-o4-mini-high, Qwen3-VL-235B-A22B-
 309 Thinking, Llama-4-Scout-17B, Llama-4-Maverick-17B-128E, Gemini2.5-Pro, DeepSeek-VL.

310 **Metrics.** We use the *F1 score* for Tasks I, II, and IV, as they involve multiple instances (e.g., M ,
 311 Z , or graph edges) that require set-level evaluation. For Task III, which typically involves a single
 312 intervention on X , we use *accuracy* to assess whether the LLM correctly identifies the intervened X .

313 4.1 LLM PERFORMANCE ON DECOMPOSITIONAL TASKS (RQ₁)

314 **Setting.** We evaluate LLMs independently on each decompositional task. For each task, we explicitly
 315 provide the ground-truth outputs from the preceding tasks to isolate and measure LLMs’ capabilities
 316 specific to that task. For example, when assessing models’ ability to construct causal graphs, we
 317 supply the original inputs along with the ground-truth causal variables.

318 **Task I: Causal Variable Identification.** Table 2 presents the performance of LLMs on identifying
 319 causal variables (X, Z, M, Y). We observe that model performance is strongly influenced by the
 320 modality of the dataset. Specifically, datasets involving more complex modalities (e.g., images,

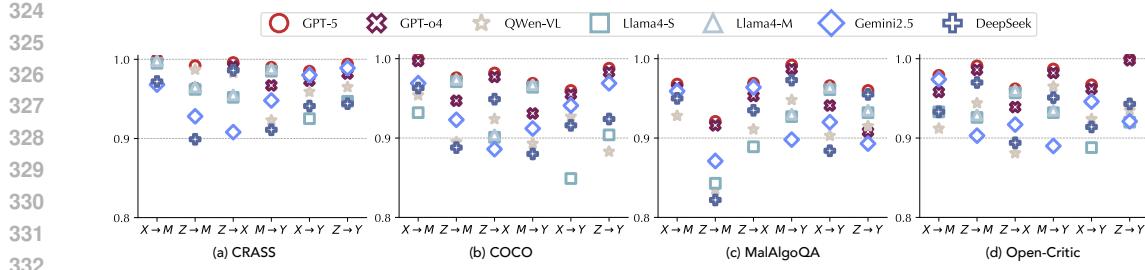


Figure 3: Evaluation on causal graph construction. We evaluate F1 score to balance (i) whether the constructed edges under one category (e.g., $X \rightarrow M$) is correctly constructed if the (X, Z, M, Y) are already given. Additional results for all other datasets at Figure 5

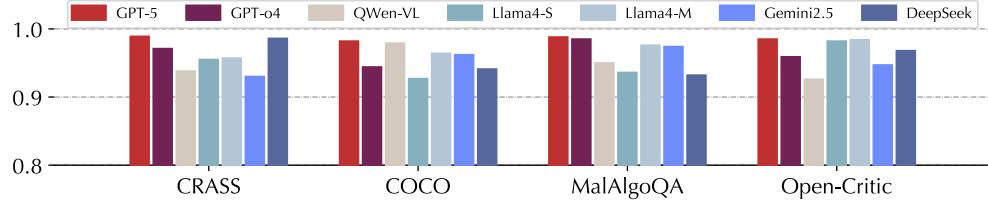


Figure 4: Evaluation of LLMs' accuracy in identifying the correct intervention (i.e., the counterfactual value of X). Additional results for all other datasets are provided in Figure 6

mathematical symbols, codes) tend to reduce LLM accuracy (e.g., <0.7 F1 on Open-Critic), even when variables like X, Z, Y are explicitly present in the context.

Interestingly, even within the text modality, LLMs show notable difficulty in identifying the implicit mediator M , which often requires reasoning about the underlying causal pathways connecting X, Z , and Y . This suggests that the challenge lies not only in the complexity of the input modality but also in the abstractness and inferential nature of the variable type itself. Together, these findings highlight the need for improved methods that enhance LLMs' capacity to handle both cross-modal complexity and deeper causal reasoning.

Task II: Causal Graph Construction. As described in our experimental setup, we isolate each decompositional step by providing the ground-truth outputs of preceding steps as inputs. For causal graph construction, we supply the identified variables X, Z, M, Y and prompt LLMs to construct the corresponding counterfactual graph. The results are presented in Figures 3 and 5. Notably, the overall performance mostly exceeds 0.9 F1 scores, indicating that LLMs can accurately construct graph edges. Moreover, the impact of dataset modality and variable types (e.g., explicit Z vs. implicit M) appears to be minimal in this step. We attribute this to the rule-based nature of causal graph construction: since causal graph structures are well-defined (as shown in Figure 2), it is relatively less challenging for LLMs to apply construction rules and generate the correct causal relationships.

Insights from causality modeling.

The major challenge in causal modeling lies in causal variable identification, where (1) LLMs are highly sensitive to the complexity and structure of the input modality, and (2) implicit variables (i.e., the mediator M) reveal a critical gap in LLMs' causal reasoning capabilities.

Task III: Counterfactual Identification. Next, we evaluate LLMs' capability in identifying interventions, i.e., determining the counterfactual values of X (i.e., X'). As shown in Figures 4 and 6, the experimental results indicate that LLMs are generally effective at recognizing the counterfactual values of X across most datasets and modalities. This demonstrates that LLMs have a solid grasp of pinpointing intervention points within the context. However, note that the task remains relatively

378

379
380
Table 3: LLM performance (F1 mean) in reasoning the counterfactual mediator (M') and outcome
(Y'). Standard deviation is in Table 9.

Dataset	GPT-5		GPT-o4		Qwen3		Llama4-S		Llama4-M		Gemini2.5		DeepSeek	
	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'
CRASS	92.1	88.0	90.5	86.2	80.5	73.9	70.1	63.5	84.9	79.5	81.7	75.2	82.9	77.1
CLOMO	90.2	85.3	88.7	83.9	77.8	71.6	67.2	60.9	82.9	77.2	79.3	72.8	80.5	74.3
RNN-Topo	88.9	83.4	87.9	81.6	75.6	69.4	65.2	58.7	80.5	75.0	77.1	70.6	78.3	72.0
CVQA-Bool	81.2	74.5	77.1	70.2	65.4	58.6	54.8	48.1	70.9	64.3	63.2	56.8	66.7	59.8
CVQA-Count	79.2	72.0	76.2	69.3	62.2	55.7	51.7	45.2	67.5	61.0	60.1	53.8	63.5	56.9
COCO	77.8	70.1	75.3	66.9	60.1	53.4	49.3	42.7	65.4	58.7	57.8	51.5	61.3	54.6
Arithmetic	87.8	82.7	85.8	80.9	69.8	63.2	59.1	52.4	76.3	70.6	72.1	65.4	74.0	67.5
MalAlgoQA	85.1	79.6	83.6	77.8	67.5	60.9	57.4	50.7	74.0	68.2	69.6	62.9	71.8	65.1
HumanEval-Exe	75.7	71.5	73.4	66.5	58.2	51.5	47.7	41.2	63.6	56.9	55.8	49.4	59.4	52.7
Open-Critic	75.3	69.4	73.8	67.5	56.0	49.4	45.8	39.2	61.5	54.7	53.7	47.3	57.2	50.6
Code-Preference	77.0	71.0	74.4	66.8	57.1	50.4	46.6	40.0	62.7	55.9	54.7	48.3	58.3	51.6

390

391

392
393
Table 4: Improvement in LLM performance (comparing with Table 2) for identifying explicit causal
variables. Results are reported on six representative datasets spanning all major modalities.

Dataset	GPT-o4			Qwen3			Llama4-S			Gemini2.5		
	X	Z	Y	X	Z	Y	X	Z	Y	X	Z	Y
CRASS	+6.0	+6.6	+5.5	+10.8	+9.4	+9.7	+15.2	+11.5	+11.9	+6.6	+5.3	+7.1
CLOMO	+4.7	+5.1	+7.9	+12.7	+11.3	+12.1	+21.0	+14.4	+16.8	+6.2	+9.4	+15.5
CVQA-Count	+17.7	+15.9	+18.2	+21.9	+21.4	+22.7	+32.0	+26.1	+24.1	+18.1	+15.7	+19.3
COCO	+5.5	+3.1	+4.4	+8.9	+7.6	+8.2	+7.2	+6.0	+9.7	+6.9	+3.8	+5.2
MalAlgoQA	+4.2	+5.9	+2.6	+9.6	+8.1	+6.8	+12.5	+10.2	+6.5	+12.4	+8.5	+8.3
Open-Critic	+12.8	+14.5	+7.0	+15.4	+10.9	+8.3	+17.4	+5.1	+1.2	+7.3	+6.1	+9.8

401

402

403
404
isolated and does not challenge the model’s ability to propagate the effects of the intervention through
downstream variables (e.g., M', Y'), which we address in Task IV.

405

406
407
408
409
410
411
412
Task IV: Outcome Reasoning. In this final task, we evaluate LLMs’ ability to infer the mediator
413 (M') and outcome (Y') under a counterfactual intervention. As shown in Table 3, LLMs consistently
414 exhibit insufficient performance in inferring these implicit variables across all datasets. Notably,
415 since both M' and Y' are implicit under the counterfactual condition (whereas only M is implicit in
416 the factual condition), this result suggests that LLMs lack sufficient capacity to reason over causal
417 chains, even when the underlying structure is explicitly provided.418
419
420
421
422
423
424
Insights from counterfactual reasoning.425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1230
1231
1232
123

432

433

Table 5: Improvement of LLM performance (F1 score) in reasoning implicit variables.

434

435

Dataset	Elicitation	GPT-o4			Qwen3			Llama4-S			Gemini2.5		
		M	M'	Y'	M	M'	Y'	M	M'	Y'	M	M'	Y'
CRASS	CoT	+5.6	+4.0	+3.1	+7.4	+5.6	+4.1	+7.9	+5.8	+4.2	+6.8	+4.5	+3.2
	CoT-SC	+5.3	+5.0	+5.5	+8.6	+7.3	+6.0	+10.5	+8.2	+5.7	+9.4	+6.8	+5.0
	ToT	+6.8	+5.2	+4.1	+8.9	+6.7	+5.0	+8.9	+7.4	+4.7	+8.2	+5.9	+4.3
CLOMO	CoT	+6.4	+4.9	+3.6	+8.3	+6.5	+4.8	+8.2	+6.9	+5.0	+7.6	+5.3	+3.9
	CoT-SC	+7.0	+5.2	+4.2	+9.6	+7.6	+5.4	+9.9	+7.7	+5.5	+8.9	+6.5	+4.7
	ToT	+10.1	+3.8	+2.9	+8.7	+5.2	+3.5	+7.2	+5.1	+3.9	+12.4	+4.2	+3.0
CVQA-Count	CoT	+5.7	+4.4	+3.5	+7.9	+6.1	+4.4	+8.2	+6.4	+4.5	+7.3	+5.4	+3.8
	CoT-SC	+4.1	+4.2	+2.4	+7.2	+6.6	+4.8	+5.9	+6.8	+5.7	+12.2	+7.6	+4.5
	ToT	+4.9	+3.8	+3.1	+6.9	+5.3	+4.0	+6.5	+5.2	+4.0	+6.1	+4.5	+3.1
COCO	CoT	+3.8	+2.9	+2.1	+6.0	+4.6	+3.2	+5.4	+4.0	+2.6	+4.9	+3.4	+2.3
	CoT-SC	+5.4	+4.1	+3.1	+7.2	+5.7	+4.3	+7.6	+5.9	+3.8	+6.9	+5.1	+3.6
	ToT	+4.5	+3.5	+2.7	+6.8	+5.1	+3.6	+6.2	+5.0	+3.5	+5.6	+4.3	+3.0
MalAlgoQA	CoT	+4.6	+3.5	+2.7	+7.0	+5.5	+4.0	+6.6	+5.0	+3.3	+5.8	+4.0	+2.7
	CoT-SC	+5.5	+4.3	+3.4	+7.7	+6.1	+4.6	+7.9	+6.3	+4.1	+7.2	+5.2	+3.7
	ToT	+6.2	+4.8	+3.8	+8.5	+6.7	+5.0	+8.6	+7.1	+4.6	+8.1	+6.0	+4.3
Open-Critic	CoT	+3.5	+2.7	+2.0	+5.3	+4.2	+3.0	+5.0	+3.7	+2.5	+4.5	+3.1	+2.1
	CoT-SC	+4.2	+3.3	+2.6	+6.1	+4.9	+3.6	+5.8	+4.5	+3.1	+5.0	+3.5	+2.7
	ToT	+5.0	+3.8	+2.9	+6.7	+5.2	+3.8	+6.1	+5.3	+3.6	+6.3	+4.7	+3.4

448

GRAPHCODEBERT (Guo et al., 2020) to extract functions, variables, and control structures for programming tasks.

451

After identifying candidate entities across each modality, we prompt the LLM to refine and filter the final set of explicit variables (X, Z, Y) according to the formal definitions provided in Section 3.1

452

453

Experimental Results. We randomly select three representative LLMs and conduct experiments across multiple datasets. As shown in Table 4, tool-augmented execution consistently improves LLM performance in identifying explicit causal variables (X, Z, Y) across all modalities. For example, by leveraging GRAPHCODEBERT to parse code structures and forward the results to Llama, which gains a clearer understanding of programming logic and achieves an F1 improvement up to 0.189. Similarly, in the vision modality, the object detector GROUNDING-DINO-BASE assists by generating a set of candidate visual objects, which GPT-o4 can then contextualize and compose into a coherent factual variable. For instance, detected objects like “Woman,” “Knife,” and “Apple” can be effectively integrated by GPT-o4 into the causal expression: “A woman cuts an apple with a knife.”

462

These results demonstrate that tool-augmented learning effectively mitigates modality-specific bottlenecks by offloading low-level entity recognition to specialized models, allowing the LLM to focus on higher-level reasoning. Looking forward, there is potential to explore alternative tool configurations that may yield comparable or even superior performance. Additionally, future work may explore multi-agent frameworks with specialized agents to collaboratively handle different variable categories.

463

4.2.2 ADVANCED ELICITATION STRATEGIES FOR REASONING OVER IMPLICIT VARIABLES

464

465

466

467

Settings. To enhance LLM reasoning over implicit variables, particularly factual M and counterfactual M', Y' , we implement advanced elicitation strategies that guide the model through more structured reasoning. Specifically, we apply Chain-of-Thought (CoT) (Wei et al., 2022), CoT with Self-Consistency (CoT-SC) (Wang et al., 2022), and Tree-of-Thought (ToT) (Yao et al., 2023):

468

- **CoT:** given pre-determined explicit variables, LLMs are encouraged to infer intermediate variables step-by-step.
- **CoT-SC:** LLMs are prompted to generate multiple reasoning paths and select the final answer via majority voting or consensus.
- **ToT:** LLMs are prompted to explore multiple parallel reasoning paths in a branching structure and evaluate candidate outputs based on intermediate criteria.

469

470

471

472

473

In implementation, both CoT-SC and ToT are executed with $k=5$ sampled reasoning paths. ToT further evaluates candidate outputs by scoring their textual similarity using BERTScore (Zhang et al., 2019), specifically assessing how well the intermediate results align with the original task statement.

485

Experimental Results. Our experimental results in Table 5 show that advanced elicitation strategies generally lead to improved performance in reasoning over implicit variables (M, M', Y') despite

486
 487 **Table 6: Overall change of LLM performance (F1 score) with different improvement strategies.**
 488 The performance change is compared with Table 3, where results of Y' demonstrate the final
 489 counterfactual reasoning outcomes.

Dataset	Improve Explicit Variable (§4.2.1)				Improve Implicit Variable (§4.2.2)				Improve Both			
	G5	QW3	LM4S	GM2.5	G5	QW3	LM4S	GM2.5	G5	QW3	LM4S	GM2.5
CRASS	+1.8	+2.3	+3.5	+2.0	+6.2	+7.4	+7.9	+6.8	+9.0	+10.1	+10.5	+9.3
CLOMO	+2.1	+3.1	+4.2	+2.6	+5.7	+6.3	+6.1	+6.5	+9.2	+10.4	+11.0	+9.8
COCO	+1.2	+1.5	+1.9	+1.4	+4.1	+5.1	+5.9	+5.1	+5.8	+6.4	+6.7	+6.2
Open-Critic	+1.6	+2.3	+2.7	+2.0	+3.3	+8.3	+11.2	+4.3	+5.3	+11.5	+14.2	+5.5

495
 496 factual or counterfactual cases. However, we also observe that more complex prompting strategies
 497 (e.g., CoT-SC and ToT) can sometimes perform slightly worse than simpler approaches (e.g., CoT).
 498 While these advanced methods encourage more exhaustive exploration of reasoning paths, they may
 499 also induce overthinking behavior in LLMs, leading the model to introduce unnecessary causal links
 500 or misinterpret the underlying problem structure. For instance, consider the following context “A
 501 *person is running a marathon and collapses.*” with expected mediator “*Dehydration*”. While CoT
 502 and CoT-SC strategies correctly identify the mediator, ToT leads LLMs to overanalyze and identify
 503 “*lack of training*” or “*overexertion*” as the mediator. These choices, although related, are not directly
 504 supported by the input data and reflect an over-extension of the reasoning process.

505 The over-qualification of elicitation strategies (e.g., ToT) highlights that, while advanced prompting
 506 techniques can improve reasoning capabilities, they may also introduce complexities that divert the
 507 model from the most straightforward and contextually supported causal pathways. Therefore, it’s
 508 crucial to balance the reasoning depth while maintaining alignment with the input data.

510 4.2.3 OVERALL PERFORMANCE IMPROVEMENT

511 In an end-to-end setting, we incorporate the prior improvements for explicit variable identification
 512 (Section 4.2.1), implicit variable reasoning (Section 4.2.2), and their combinatorial strategy to
 513 evaluate the overall change in counterfactual reasoning performance. Table 6 presents the results
 514 across representative datasets and LLMs.

515 We have two observations: (i) In general, improving implicit variable reasoning (i.e., for mediators
 516 and outcomes) yields more substantial gains in end-to-end performance, as these variables directly
 517 influence the final counterfactual predictions (Task IV). In contrast, improvements in explicit variable
 518 identification (e.g., for X, Z, Y) primarily strengthen the early stages of the pipeline, offering
 519 moderate but necessary support. (ii) Notably, the combined strategy achieves the highest overall
 520 improvement, although the gains are not strictly additive. This is due to accumulative reasoning
 521 errors across the decompositional steps, where inaccuracies in earlier predictions may propagate and
 522 compound through the pipeline. These findings highlight both the opportunities and challenges of
 523 modular improvements.

525 5 CONCLUSION

527 This work provides a decompositional framework for evaluating counterfactual reasoning in LLMs.
 528 We collect a set of datasets across multimodalities, and then curate them with reference causal
 529 variables, structured graphs, and counterfactual intervention. Next, we use our curated dataset
 530 to study the reasoning process into distinct stages from causal variable identification to outcome
 531 inference. We uncover the qualifications of current LLMs in counterfactual reasoning and where
 532 they fall short. Based on experimental insights, we further propose improvements to offer actionable
 533 insights for enhancing LLM reasoning, particularly in multimodality and implicit reasoning settings.

540 ETHICS STATEMENT

541

542 This work does not raise ethical concerns. All experiments were conducted on publicly available
 543 datasets spanning text, image, code, and symbolic reasoning tasks (e.g., CRASS, CLOMO, COCO,
 544 HumanEval). No private, personal, or sensitive information was accessed or processed during this
 545 research. The methodology and evaluations strictly comply with the licensing terms and intended
 546 academic usage of the benchmark datasets.

547

548 REPRODUCIBILITY STATEMENT

549

550 To support reproducibility, we have curated and released the complete benchmark construction
 551 process, together with evaluation scripts, model prompts, and experimental configurations, through an
 552 anonymous GitHub repository released in Introduction section. This repository provides detailed in-
 553 structions for dataset preprocessing, task decomposition, and model evaluation, enabling independent
 554 researchers to reproduce and extend our experiments.

555

556 REFERENCES

557

Elias Bareinboim, Juan D Correa, Duligur Ibeling, and Thomas Icard. On pearl’s hierarchy and the
 foundations of causal inference. In *Probabilistic and causal inference: the works of judea pearl*,
 pp. 507–556. 2022.

559

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared
 Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large
 language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

564

Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han.
 Unveiling causal reasoning in large language models: Reality or mirage? *Advances in Neural
 Information Processing Systems*, 37:96640–96670, 2024.

568

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep
 bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL
<http://arxiv.org/abs/1810.04805>

571

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep
 bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of
 the North American chapter of the association for computational linguistics: human language
 technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

576

Fuli Feng, Jizhi Zhang, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. Empowering language
 understanding with counterfactual reasoning. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto
 Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*,
 pp. 2226–2236, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/
 v1/2021.findings-acl.196. URL <https://aclanthology.org/2021.findings-acl.196/>

582

Jörg Frohberg and Frank Binder. Crass: A novel data set and benchmark to test counterfactual
 reasoning of large language models. *arXiv preprint arXiv:2112.11941*, 2021.

585

Yair Ori Gat, Nitay Calderon, Amir Feder, Alexander Chapanin, Amit Sharma, and Roi Reichart.
 Faithful explanations of black-box nlp models using llm-generated counterfactuals. In *The Twelfth
 International Conference on Learning Representations*, 2024.

588

Ian D Gow, David F Larcker, and Peter C Reiss. Causal inference in accounting research. *Journal of
 Accounting Research*, 54(2):477–523, 2016.

591

Daya Guo, Shuo Ren, Shuai Lu, Zhangyin Feng, Duyu Tang, Shujie Liu, Long Zhou, Nan Duan,
 Alexey Svyatkovskiy, Shengyu Fu, et al. Graphcodebert: Pre-training code representations with
 data flow. *arXiv preprint arXiv:2009.08366*, 2020.

594 Emilia Gvozdenović, Lucio Malvisi, Elisa Cinconze, Stijn Vansteelandt, Phoebe Nakanwagi, Em-
 595 manuel Aris, and Dominique Rosillon. Causal inference concepts applied to three observational
 596 studies in the context of vaccine development: from theory to practice. *BMC Medical Research
 597 Methodology*, 21:1–10, 2021.

598 Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to
 599 reason: End-to-end module networks for visual question answering. In *Proceedings of the IEEE
 600 international conference on computer vision*, pp. 804–813, 2017.

601 Yinya Huang, Ruixin Hong, Hongming Zhang, Wei Shao, Zhicheng Yang, Dong Yu, Changshui
 602 Zhang, Xiaodan Liang, and Linqi Song. Clomo: Counterfactual logical modification with large
 603 language models. *arXiv preprint arXiv:2311.17438*, 2023.

604 Yinya Huang, Ruixin Hong, Hongming Zhang, Wei Shao, Zhicheng Yang, Dong Yu, Changshui
 605 Zhang, Xiaodan Liang, and Linqi Song. Clomo: Counterfactual logical modification with large
 606 language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of
 607 the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long
 608 Papers)*, pp. 11012–11034, Bangkok, Thailand, August 2024. Association for Computational
 609 Linguistics. doi: 10.18653/v1/2024.acl-long.593. URL <https://aclanthology.org/2024.acl-long.593/>.

610 Licheng Jiao, Yuhang Wang, Xu Liu, Lingling Li, Fang Liu, Wenping Ma, Yuwei Guo, Puhua Chen,
 611 Shuyuan Yang, and Biao Hou. Causal inference meets deep learning: A comprehensive survey.
 612 *Research*, 7:0467, 2024.

613 Atoosa Kasirzadeh and Andrew Smart. The use and misuse of counterfactuals in ethical machine learn-
 614 ing. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*,
 615 pp. 228–236, 2021.

616 Emre Kiciman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language
 617 models: Opening a new frontier for causality. *Transactions on Machine Learning Research*, 2023.

618 Lisa Koonce, Karen K Nelson, and Catherine M Shakespeare. Judging the relevance of fair value for
 619 financial instruments. *The Accounting Review*, 86(6):2075–2098, 2011.

620 Jiayi Kuang, Ying Shen, Jingyou Xie, Haohao Luo, Zhe Xu, Ronghao Li, Yinghui Li, Xianfeng
 621 Cheng, Xika Lin, and Yu Han. Natural language understanding and inference with mllm in visual
 622 question answering: A survey. *ACM Comput. Surv.*, 57(8), March 2025. ISSN 0360-0300. doi:
 623 10.1145/3711680. URL <https://doi.org/10.1145/3711680>.

624 Evangelia Kyrimi, Somayyeh Mossadegh, Jared M Wohlgemut, Rebecca S Stoner, Nigel RM Tai,
 625 and William Marsh. Counterfactual reasoning using causal bayesian networks as a healthcare
 626 governance tool. *International Journal of Medical Informatics*, 193:105681, 2025.

627 Tiep Le, Vasudev Lal, and Phillip Howard. Coco-counterfactuals: Automatically constructed
 628 counterfactual examples for image-text pairs. *Advances in Neural Information Processing Systems*,
 629 36:71195–71221, 2023.

630 Jiaxuan Li, Lang Yu, and Allyson Ettinger. Counterfactual reasoning: Testing language models'
 631 understanding of hypothetical scenarios. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki
 632 (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
 633 (Volume 2: Short Papers)*, pp. 804–815, Toronto, Canada, July 2023. Association for Computational
 634 Linguistics. doi: 10.18653/v1/2023.acl-short.70. URL <https://aclanthology.org/2023.acl-short.70/>.

635 Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang,
 636 Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for
 637 open-set object detection, 2023.

638 Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord,
 639 Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for
 640 science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521,
 641 2022.

648 Van Bach Nguyen, Paul Youssef, Christin Seifert, and Jörg Schlötterer. LLMs for generating and
 649 evaluating counterfactuals: A comprehensive study. In Yaser Al-Onaizan, Mohit Bansal, and Yun-
 650 Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp.
 651 14809–14824, Miami, Florida, USA, November 2024. Association for Computational Linguistics.
 652 doi: 10.18653/v1/2024.findings-emnlp.870. URL <https://aclanthology.org/2024-findings-emnlp.870/>.
 653

654 Judea Pearl. *Causality*. Cambridge university press, 2009.
 655

656 Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic books,
 657 2018.
 658

659 Zhaozhi Qian, Yao Zhang, Ioana Bica, Angela Wood, and Mihaela van der Schaar. SyncTwin:
 660 Treatment effect estimation with longitudinal outcomes. In M. Ranzato, A. Beygelz-
 661 imer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural
 662 Information Processing Systems*, volume 34, pp. 3178–3190. Curran Associates, Inc.,
 663 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/19485224d128528da1602ca47383f078-Paper.pdf.
 664

665 Shauli Ravfogel, Yoav Goldberg, and Tal Linzen. Studying the inductive biases of rnns with synthetic
 666 variations of natural languages. *arXiv preprint arXiv:1903.06400*, 2019.
 667

668 Aryan Srivastava and Paula Akemi Aoyagui. Dice: A framework for dimensional and contextual
 669 evaluation of language models. *arXiv preprint arXiv:2504.10359*, 2025.
 670

671 Shashank Sonkar, Naiming Liu, MyCo Le, and Richard Baraniuk. Malalgoqa: Pedagogical evaluation
 672 of counterfactual reasoning in large language models and implications for ai in education. In
 673 *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15554–15567, 2024.
 674

675 Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task:
 676 Language-independent named entity recognition. In *Proceedings of the Seventh Conference on
 677 Natural Language Learning at HLT-NAACL 2003*, pp. 142–147, 2003. URL <https://www.aclweb.org/anthology/W03-0419>.
 678

679 Vezora. Code preference pairs. <https://huggingface.co/datasets/Vezora/Code-Preference-Pairs>, 2024.
 680

681 Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdh-
 682 ery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models.
 683 *arXiv preprint arXiv:2203.11171*, 2022.
 684

685 Yongjie Wang, Xiaoqi Qiu, Yu Yue, Xu Guo, Zhiwei Zeng, Yuhong Feng, and Zhiqi Shen. A
 686 survey on natural language counterfactual generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-
 687 Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp.
 688 4798–4818, Miami, Florida, USA, November 2024. Association for Computational Linguistics.
 689 doi: 10.18653/v1/2024.findings-emnlp.276. URL <https://aclanthology.org/2024-findings-emnlp.276/>.
 690

691 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny
 692 Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in
 693 neural information processing systems*, 35:24824–24837, 2022.
 694

695 Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim,
 696 Jacob Andreas, and Yoon Kim. Reasoning or reciting? exploring the capabilities and limitations
 697 of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the
 698 North American Chapter of the Association for Computational Linguistics: Human Language
 699 Technologies (Volume 1: Long Papers)*, pp. 1819–1862, 2024.
 700

701 Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu,
 Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning
 and action. *arXiv preprint arXiv:2303.11381*, 2023.

702 Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik
 703 Narasimhan. Tree of thoughts: deliberate problem solving with large language models. In
 704 *Proceedings of the 37th International Conference on Neural Information Processing Systems*,
 705 NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

707 Chunhui Zhang, Yiren Jian, Zhongyu Ouyang, and Soroush Vosoughi. Working memory iden-
 708 tifies reasoning limits in language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-
 709 Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural*
 710 *Language Processing*, pp. 16896–16922, Miami, Florida, USA, November 2024a. Associa-
 711 *tion for Computational Linguistics*. doi: 10.18653/v1/2024.emnlp-main.938. URL <https://aclanthology.org/2024.emnlp-main.938/>.

713 Letian Zhang, Xiaotong Zhai, Zhongkai Zhao, Yongshuo Zong, Xin Wen, and Bingchen Zhao. What
 714 if the tv was off? examining counterfactual reasoning abilities of multi-modal language models.
 715 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.
 716 21853–21862, 2024b.

718 Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating
 719 text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

722 A COMPLEMENTARY INFORMATION OF CAUSALITY AND CAUSAL GRAPH

724 This appendix presents a concise overview of each dataset, followed by its causal structure and
 725 graphical representation, alongside a concrete example. For each dataset, we identify the four variable
 726 types—Exposure, Covariate, Mediator, and Outcome—and distinguish their roles in both factual and
 727 counterfactual scenarios, illustrating each with directed edges in the corresponding causal graph. In
 728 addition, we include a sample prompt used to generate responses on a simple text-parsing Q&A task
 729 dataset.

730 In evaluation, we rely on a universal prompt template as shown below, wherein only the task-specific
 731 contents are replaced for each dataset.

732 **Prompt Template.**

```

734 {
735  **Task Description**
736  You are asked to perform [Task I / Task II / Task III / Task IV] in
737  decompositional counterfactual reasoning. Follow the definitions of
738  causal variables and causal relations strictly.

739  **Input Context**
740  Here is the factual instance from the dataset:
741  [Insert factual context or multimodal description]

742  **Intermediate Outputs**
743  If applicable, use the following ground-truth results from previous tasks
744  :
745  Exposure (X): [...]
746  Covariate(s) (Z): [...]
747  Mediator(s) (M): [...]
748  Outcome (Y): [...]

749  **Instruction for the Current Task**
750  Task I (Variable Identification): [description about domain specific
751  meanings of causal variables X, Z, M, and Y and the identification
752  task]
753  Task II (Graph Construction): [instruction about constructing the causal
754  graph by listing all directed edges among X, Z, M, Y.
755  Task III (Intervention Identification): [instruction about identifying
    which variable is intervened on in the counterfactual query.]
```

```
756 | Task IV (Outcome Reasoning): [instruction about inferring] the
757 | counterfactual mediator  $M'$  and outcome  $Y'$  under the specified
758 | intervention.
759 |
760 | **Output Format**
761 | Provide the answer using the following structure:
762 | Exposure (X): [...]
763 | Covariate(s) (Z): [...]
764 | Mediator(s) (M): [...]
765 | Outcome (Y): [...]
766 | Causal Edges: [...]
767 | Intervention: [...]
768 | Counterfactual Mediator ( $M'$ ): [...]
769 | Counterfactual Outcome ( $Y'$ ): [...]
770 | }
```

CRASS Example. The Counterfactual Reasoning Assessment for Structured Scenarios (CRASS) dataset is designed to evaluate whether language models can reason about hypothetical alternatives to factual events. Each example in CRASS presents a factual scenario (e.g., “A woman opens a treasure chest”) followed by a counterfactual question (e.g., “What would have happened if the woman had not opened the treasure chest?”). Models are asked to select the most logically consistent outcome from multiple-choice options, such as “The treasure chest would have remained closed”, which is labeled as correct. The following displays a full example:

```
777 {
778     "input": "A woman opens a treasure chest. What would have happened if
779         the woman had not opened the treasure chest?",
780     "target_scores": {
781         "The treasure chest would have been open.": 0,
782         "That is not possible.": 0,
783         "The treasure chest would have remained closed.": 1,
784         "I don't know.": 0
785     }
786 }
```

787 CRASS Causality & Causal Graph.

```
788 {
789     "factual_roles": {
790         "Exposure": ["act of opening treasure chest"],
791         "Covariate": ["key possession", "physical capability"],
792         "Mediator": ["lock mechanism release"],
793         "Outcome": ["chest opened"]
794     },
795     "counterfactual_roles": {
796         "Exposure": ["omission of opening action"],
797         "Covariate": ["key possession", "physical capability"],
798         "Mediator": ["lock state preservation"],
799         "Outcome": ["chest remains closed"]
800     },
801     "causal_graph": {
802         "factual_edges": [
803             ["key possession", "act of opening treasure chest"],
804             ["key possession", "lock mechanism release"],
805             ["key possession", "chest opened"],
806             ["physical capability", "act of opening treasure chest"],
807             ["physical capability", "lock mechanism release"],
808             ["physical capability", "chest opened"],
809             ["act of opening treasure chest", "lock mechanism release"],
810             ["lock mechanism release", "chest opened"],
811             ["act of opening treasure chest", "chest opened"],
812         ],
813         "counterfactual_edges": [
814             ["key possession", "lock state preservation"],
815             ["key possession", "lock state preservation"]
816         ]
817     }
818 }
```

```

810     ["key possession", "chest remains closed"],
811     ["physical capability", "lock state preservation"],
812     ["physical capability", "chest remains closed"],
813     ["omission of opening action", "lock state preservation"],
814     ["lock state preservation", "chest remains closed"],
815     ["omission of opening action", "chest remains closed"],
816   ]
817 }
818

```

819 **CLOMO Example.** The Counterfactual Logical Modification (**CLOMO**) dataset is designed
 820 to evaluate whether large language models can perform controlled, counterfactual edits to natural
 821 language arguments in a logically coherent way. Each example presents a base argument and two
 822 premises: Premise 1 has a logical sensitivity with the original argument, while Premise 2 does not.
 823 The model is instructed to modify the argument such that Premise 2 has a logical sensitivity with
 824 the original argument, while Premise 1 no longer is. For instance, given an argument attributing the
 825 rise in gasoline prices fully to government policies, the model must produce a revised version (e.g.,
 826 changing “fully responsible” to “partly leads”) of an argument that shifts logical sensitivity from one
 827 premise to another without introducing new claims. The following displays a full example:
 828

```

828 {
829   "instruction": "In the following, you will see an argument and 2
830   premises, where Premise 1 provides a necessary assumption to the
831   Argument. Please modify the Statements in the Argument until
832   Premise 2 provides a necessary assumption to the Argument instead,
833   while Premise 1 fails to provide a necessary assumption to the
834   Argument. Note that no additional statement should be added. ",
835   "input": "Argument: Statement1: Consumer advocate : there is no doubt
836   that the government is responsible for the increased cost of
837   gasoline, because the government's policies have significantly
838   increased consumer demand for fuel, and as a result of increasing
839   demand, the price of gasoline has risen steadily.Premise1: The
840   government can bear responsibility for that which it indirectly
841   causes.Premise2: Consumer demand for gasoline cannot increase
842   without causing gasoline prices to increase.Please write the
843   modified argument below: ",
844   "output": "Statement1: Consumer advocate : there is no doubt that the
845   government partly leads to the increased cost of gasoline, because
846   the government's policies have significantly increased consumer
847   demand for fuel, and as a result of increasing demand, the price
848   of gasoline has risen steadily undoubtedly."
849 }

```

847 **CLOMO Causality & Causal Graph.**

```

849 {
850   "factual_roles": {
851     "Exposure": ["Premise 1 as necessary assumption"],
852     "Covariate": [
853       "Government's policy impact on demand",
854       "Demand-price relationship assumption"
855     ],
856     "Mediator": ["Causal attribution mechanism (direct vs indirect)"],
857     "Outcome": ["Full responsibility attribution to government"]
858   },
859   "counterfactual_roles": {
860     "Exposure": ["Premise 2 as necessary assumption"],
861     "Covariate": [
862       "Government's policy impact on demand",
863       "Demand-price relationship assumption"
864     ],
865     "Mediator": ["Responsibility attribution modifier (partial vs full)"]
866   },
867   "Outcome": ["Partial responsibility attribution to government"]
868 }

```

```

864 },
865 "causal_graph": {
866   "factual_edges": [
867     ["Government's policy impact on demand", "Premise 1 as necessary
868       assumption"],
869     ["Government's policy impact on demand", "Causal attribution
870       mechanism"],
871     ["Government's policy impact on demand", "Full responsibility
872       attribution to government"],
873     ["Demand-price relationship assumption", "Premise 1 as necessary
874       assumption"],
875     ["Demand-price relationship assumption", "Causal attribution
876       mechanism"],
877     ["Demand-price relationship assumption", "Full responsibility
878       attribution to government"],
879     ["Premise 1 as necessary assumption", "Causal attribution mechanism
880       "],
881     ["Causal attribution mechanism", "Full responsibility attribution
882       to government"],
883     ["Premise 1 as necessary assumption", "Full responsibility
884       attribution to government"]
885   ],
886   "counterfactual_edges": [
887     ["Government's policy impact on demand", "Responsibility
888       attribution modifier"],
889     ["Government's policy impact on demand", "Partial responsibility
890       attribution to government"],
891     ["Demand-price relationship assumption", "Responsibility
892       attribution modifier"],
893     ["Demand-price relationship assumption", "Partial responsibility
894       attribution to government"],
895     ["Premise 2 as necessary assumption", "Responsibility attribution
896       modifier"],
897     ["Responsibility attribution modifier", "Partial responsibility
898       attribution to government"],
899     ["Premise 2 as necessary assumption", "Partial responsibility
900       attribution to government"]
901   ]
902 }
903 }
904

```

899 **RNN-Typology Example.** This is a synthetic dataset contains sentence pairs that reflect syntactic
900 alterations to word orders (e.g., converting English from subject-verb-object (SVO) to subject-object-
901 verb (SOV) order). For example, the factual sentence “*Tim saw Lucas.*” (SVO) is transformed to its
902 SOV equivalent “*Tim Lucas saw.*”

```

903 "tim saw lucas.": "tim lucas saw."
904

```

905 RNN-Typology Causality & Causal Graph.

```

906 {
907   "factual_roles": {
908     "Exposure": ["subject-verb-object order"],
909     "Covariate": ["syntactic rule", "Lexical items (Tim, saw, Lucas)"],
910     "Mediator": ["SOV reordering operation"],
911     "Outcome": ["tim saw lucas."]
912   },
913   "counterfactual_roles": {
914     "Exposure": ["subject-object-verb order"],
915     "Covariate": ["syntactic rule", "Lexical items (Tim, saw, Lucas)"],
916     "Mediator": ["SVO restoration operation"],
917     "Outcome": ["tim lucas saw."]
918   },
919   "causal_graph": {

```

```

918     "factual_edges": [
919         ["syntactic rule", "subject-verb-object order"],
920         ["syntactic rule", "SOV reordering operation"],
921         ["syntactic rule", "tim saw lucas."],
922         ["Lexical items (Tim, saw, Lucas)", "subject-verb-object order"],
923         ["Lexical items (Tim, saw, Lucas)", "SOV reordering operation"],
924         ["Lexical items (Tim, saw, Lucas)", "tim saw lucas."],
925         ["subject-verb-object order", "SOV reordering operation"],
926         ["SOV reordering operation", "tim saw lucas."],
927         ["subject-verb-object order", "tim saw lucas."]
928     ],
929     "counterfactual_edges": [
930         ["syntactic rule", "SVO restoration operation"],
931         ["syntactic rule", "tim lucas saw."],
932         ["Lexical items (Tim, saw, Lucas)", "SVO restoration operation"],
933         ["Lexical items (Tim, saw, Lucas)", "tim lucas saw."],
934         ["subject-object-verb order", "SVO restoration operation"],
935         ["SVO restoration operation", "tim lucas saw."],
936         ["subject-object-verb order", "tim lucas saw."]
937     ]
938 }

```

CVQA-Bool Example. Counterfactual Visual Question Answering(CVQA) is designed to assess the ability of vision-language models to perform counterfactual reasoning over images. Each example presents a factual visual query-answer pair (e.g., *“Is there a red sandal here?”* → *yes*) grounded in a COCO image, along with a corresponding counterfactual query that modifies a key visual condition (e.g., *“Would there be a red sandal here if all shoes were removed?”* → *no*). The task requires the model to infer changes in object presence or relationships under hypothetical alterations to the scene. The dataset focuses on a boolean query type. The following displays an example:

real image	query	answer	new query	new answer	type
	Is there a red sandal here?	yes	Would there be a red sandal here if all shoes were removed?	no	boolean

CVQA-Bool Causality & Causal Graph.

```

953 {
954     "factual_roles": {
955         "Exposure": ["presence of red sandal"],
956         "Covariate": ["original shoe collection in cart", "visual recognition
957                         capability"],
958         "Mediator": ["sandal-as-shoe categorical inclusion"],
959         "Outcome": ["yes"]
960     },
961     "counterfactual_roles": {
962         "Exposure": ["removal of all shoes"],
963         "Covariate": ["original shoe collection in cart", "visual recognition
964                         capability"],
965         "Mediator": ["sandal-shoe categorical dependency"],
966         "Outcome": ["no"]
967     },
968     "causal_graph": {
969         "factual_edges": [
970             ["original shoe collection in cart", "presence of red sandal"],
971             ["original shoe collection in cart", "sandal-as-shoe categorical
972                         inclusion"],
973             ["original shoe collection in cart", "yes"],
974             ["visual recognition capability", "presence of red sandal"],
975             ["visual recognition capability", "sandal-as-shoe categorical
976                         inclusion"]
977         ]
978     }
979 }

```

```

972     ["visual recognition capability", "yes"],
973     ["presence of red sandal", "sandal-as-shoe categorical inclusion"],
974     ["sandal-as-shoe categorical inclusion", "yes"],
975     ["presence of red sandal", "yes"]
976   ],
977   "counterfactual_edges": [
978     ["original shoe collection in cart", "sandal-shoe categorical
979       dependency"],
980     ["original shoe collection in cart", "no"],
981     ["visual recognition capability", "sandal-shoe categorical
982       dependency"],
983     ["visual recognition capability", "no"],
984     ["removal of all shoes", "sandal-shoe categorical dependency"],
985     ["sandal-shoe categorical dependency", "no"],
986     ["removal of all shoes", "no"]
987   ]
988 }

```

CVQA-Count Example. **Visual Counterfactual Query Dataset (CVQA)** also evaluates whether language models can perform a direct or indirect numerical counterfactual reasoning grounded in visual inputs. Each example consists of a factual visual question (e.g., *“How many plates are there?”* → 1) paired with a corresponding counterfactual query that modifies the quantity in a clearly defined way (e.g., *“How many plates would there be if 2 more plates were added?”* → 3). The model must integrate visual perception (e.g., detecting a single white plate in an image) with numerical logic (e.g., adding 2) to produce the correct answer. The dataset focuses on a counting query type. The following displays an example:

real image	query	answer	new query	new answer	type
	How many plates are there	1	How many plates would there be if 2 more plates were added?	3	direct counting

CVQA-Count Causality & Causal Graph.

```

1008 {
1009   "factual_roles": {
1010     "Exposure": ["current plate presence (1 unit)"],
1011     "Covariate": ["original plate count (1)", "visual counting capability
1012       "],
1013     "Mediator": ["visual plate detection mechanism"],
1014     "Outcome": ["1"]
1015   },
1016   "counterfactual_roles": {
1017     "Exposure": ["addition of 2 plates"],
1018     "Covariate": ["original plate count (1)", "visual counting capability
1019       "],
1020     "Mediator": ["numerical addition operation"],
1021     "Outcome": ["3"]
1022   },
1023   "causal_graph": {
1024     "factual_edges": [
1025       ["original plate count (1)", "current plate presence (1 unit)"],
1026       ["original plate count (1)", "visual plate detection mechanism"],
1027       ["original plate count (1)", "1"],
1028       ["visual counting capability", "current plate presence (1 unit)"],
1029       ["visual counting capability", "visual plate detection mechanism"],
1030       ["visual counting capability", "1"],
1031     ]
1032   }
1033 }

```

```

1026     ["current plate presence (1 unit)", "visual plate detection
1027         mechanism"],
1028     ["visual plate detection mechanism", "1"],
1029     ["current plate presence (1 unit)", "1"]
1030 ],
1031 "counterfactual_edges": [
1032     ["original plate count (1)", "numerical addition operation"],
1033     ["original plate count (1)", "3"],
1034     ["visual counting capability", "numerical addition operation"],
1035     ["visual counting capability", "3"],
1036     ["addition of 2 plates", "numerical addition operation"],
1037     ["numerical addition operation", "3"],
1038     ["addition of 2 plates", "3"]
1039 ]
1040 }
1041

```

COCO Example. Common Objects in Context(COCO) dataset provides automatically constructed counterfactual examples for evaluating multimodal reasoning in image-text pairs. Each instance contains two images and two near-identical captions that differ only in a key noun (e.g., “A *big burly grizzly bear* is shown with *grass* in the background” vs. “A *big burly grizzly bear* is shown with *deer* in the background”). The dataset is designed to test whether models can detect minimal semantic changes and determine whether the new image visually aligns with the counterfactual caption. The goal is to assess visual-textual consistency and a model’s sensitivity to causal or identity-based alterations in structured multimodal contexts. The following displays an example:

Factual Caption	Image 0	Counterfactual Caption	Image 1
A big burly grizzly bear is shown with grass in the background.		A big burly grizzly bear is shown with deer in the background.	

COCO Causality & Causal Graph.

```

1058
1059
1060 {
1061     "factual_roles": {
1062         "Exposure": ["original image (bear with grass)"],
1063         "Covariate": ["bear presence", "background context"],
1064         "Mediator": ["grass visual detection"],
1065         "Outcome": ["caption with 'grass'"]
1066     },
1067     "counterfactual_roles": {
1068         "Exposure": ["modified image (bear with deer)"],
1069         "Covariate": ["bear presence", "background context"],
1070         "Mediator": ["deer-grass substitution mechanism"],
1071         "Outcome": ["caption with 'deer'"]
1072     },
1073     "causal_graph": {
1074         "factual_edges": [
1075             ["bear presence", "original image (bear with grass)"],
1076             ["bear presence", "grass visual detection"],
1077             ["bear presence", "caption with 'grass'"],
1078             ["background context", "original image (bear with grass)"],
1079             ["background context", "grass visual detection"],
1080             ["background context", "caption with 'grass'"],
1081             ["original image (bear with grass)", "grass visual detection"],
1082             ["grass visual detection", "caption with 'grass'"],
1083             ["original image (bear with grass)", "caption with 'grass'"]
1084         ],
1085     }
1086 }
1087

```

```

1080     "counterfactual_edges": [
1081         ["bear presence", "deer-grass substitution mechanism"],
1082         ["bear presence", "caption with 'deer'"],
1083         ["background context", "deer-grass substitution mechanism"],
1084         ["background context", "caption with 'deer'"],
1085         ["modified image (bear with deer)", "deer-grass substitution
1086             mechanism"],
1087         ["deer-grass substitution mechanism", "caption with 'deer'"],
1088         ["modified image (bear with deer)", "caption with 'deer'"]
1089     ]
1090 }

```

1091 **Arithmetic Example. Base-computation Arithmetic** dataset evaluates counterfactual numerical reasoning by testing arithmetic operations across multiple numeral systems(e.g. Base-8, 9, 10, 11, 16). Each example pairs a factual base-10 calculation with a counterfactual alternate-base computation (e.g., base-8: $14_8 + 57_8 = 73_8$, base-16: $EC_{16} + DD_{16} = 1C9_{16}$). The dataset includes inputs (num1, num2), the numeral system (e.g., "8" for octal, "16" for hexadecimal), and the base-specific result (addrst). It assesses models' ability to adapt numeral system transitions and consistency in counterfactual reasoning. The following display a base-8 computation example and a base-16 example:

```

1099 {
1100     "8": {
1101         "num1": "14",
1102         "num2": "57",
1103         "addrst": "73"
1104     },
1105     "16": {
1106         "num1": "EC",
1107         "num2": "DD",
1108         "addrst": "1C9"
1109     }
1110 }

```

1111 **Base-8 Arithmetic Causality & Causal Graph.**

```

1112 {
1113     "factual_roles": {
1114         "Exposure": ["10-based system"],
1115         "Covariate": ["14", "57"],
1116         "Mediator": ["base-10 arithmetic operation"],
1117         "Outcome": ["71"]
1118     },
1119     "counterfactual_roles": {
1120         "Exposure": ["8-based system"],
1121         "Covariate": ["14", "57"],
1122         "Mediator": [
1123             "base-8 to base-10 conversion",
1124             "base-10 sum conversion to base-8"
1125         ],
1126         "Outcome": ["73"]
1127     },
1128     "causal_graph": {
1129         "factual_edges": [
1130             ["14", "10-based system"],
1131             ["14", "base-10 arithmetic operation"],
1132             ["14", "71"],
1133             ["57", "10-based system"],
1134             ["57", "base-10 arithmetic operation"],
1135             ["57", "71"],
1136             ["10-based system", "base-10 arithmetic operation"],
1137             ["base-10 arithmetic operation", "71"],
1138             ["10-based system", "71"]
1139         ]
1140     }
1141 }

```

```

1134 ],
1135 "counterfactual_edges": [
1136     ["14", "base-8 to base-10 conversion"],
1137     ["14", "73"],
1138     ["57", "base-8 to base-10 conversion"],
1139     ["57", "73"],
1140     ["8-based system", "base-8 to base-10 conversion"],
1141     ["base-8 to base-10 conversion", "base-10 sum conversion to base-8"]
1142         ],
1143     ["base-10 sum conversion to base-8", "73"],
1144     ["8-based system", "73"]
1145 ]
1146 }
1147 }
1148

```

Base-16 Arithmetic Causality & Causal Graph.

```

1149 {
1150     "factual_roles": {
1151         "Exposure": ["10-based system"],
1152         "Covariate": ["EC", "DD"],
1153         "Mediator": ["N.A."],
1154         "Outcome": ["N.A."]
1155     },
1156     "counterfactual_roles": {
1157         "Exposure": ["16-based system"],
1158         "Covariate": ["EC", "DD"],
1159         "Mediator": [
1160             "hex-to-decimal conversion",
1161             "decimal-to-hex reversion"
1162         ],
1163         "Outcome": ["1C9"]
1164     },
1165     "causal_graph": {
1166         "factual_edges": [
1167             ["EC", "10-based system"],
1168             ["DD", "10-based system"]
1169         ],
1170         "counterfactual_edges": [
1171             ["EC", "hex-to-decimal conversion"],
1172             ["hex-to-decimal conversion", "decimal-to-hex reversion"],
1173             ["EC", "1C9"],
1174             ["DD", "hex-to-decimal conversion"],
1175             ["DD", "1C9"],
1176             ["16-based system", "hex-to-decimal conversion"],
1177             ["decimal-to-hex reversion", "1C9"],
1178             ["16-based system", "1C9"]
1179         ]
1180     }
1181 }
1182

```

1178 **MalAlgoQA Example (Malformed Algorithmic Question Answering (MalAlgoQA) dataset**
1179 is designed intentionally including factual and counterfactual rationales between multiple-choice
1180 question answering to validate a language model's ability to discern sound reasoning in the presence
1181 of rationales(factual or counterfactual). Each question is presented alongside a factual rationale that
1182 supports the correct answer (e.g., “*Correctly ordered the values from greatest to least: 276, 254,*
1183 *237, 235.* → C), and is paired with counterfactual rationales (e.g., “*Ordered least to greatest*”) that
1184 correspond to plausible but incorrect or altered answers (e.g., A). Each example is decomposed into
1185 factual and counterfactual role pairs, allowing researchers to assess how changes in reasoning paths
1186 (rationales) lead to different answer choices. The following display an example of raw data and its
1187 decomposed data points:

```
{

```

```

1188 "Question":"Which list shows the following number in order from
1189     highest to lowest?",  

1190 "Answer":"C",  

1191 "Choice_A":" 235 237 254 276 ",  

1192 "Choice_B":" 237 276 235 254 ",  

1193 "Choice_C":" 276 254 237 235 ",  

1194 "Choice_D":" 276 254 235 237 ",  

1195 "Rationale_A":"Ordered least to greatest",  

1196 "Rationale_B":"Ordered greatest to least by ones place.",  

1197 "Rationale_C":"Correctly ordered the values from greatest to least:
1198     276, 254, 237, 235.",  

1199 "Rationale_D":"Switched last 2 numbers."  

1200 }  

1201 {  

1202     {  

1203         "Question":"Which list shows the following number in order from
1204             highest to lowest?",  

1205         "Answer": "C",
1206         "Counterfactual Answer": "A",
1207         "Choice_A": " 235 237 254 276 ",  

1208         "Choice_B": " 237 276 235 254 ",  

1209         "Choice_C": " 276 254 237 235 ",  

1210         "Choice_D": " 276 254 235 237 ",  

1211         "Counterfactual Rationale": "Ordered least to greatest",
1212         "Rationale_C": "Correctly ordered the values from greatest to
1213             least: 276, 254, 237, 235."
1214     },
1215     {  

1216         "Question":"Which list shows the following number in order from
1217             highest to lowest?",  

1218         "Answer": "C",
1219         "Counterfactual Answer": "B",
1220         "Choice_A": " 235 237 254 276 ",  

1221         "Choice_B": " 237 276 235 254 ",  

1222         "Choice_C": " 276 254 237 235 ",  

1223         "Choice_D": " 276 254 235 237 ",  

1224         "Counterfactual Rationale": "Ordered greatest to least by ones
1225             place",
1226         "Rationale_C": "Correctly ordered the values from greatest to
1227             least: 276, 254, 237, 235."
1228     },
1229     {  

1230         "Question":"Which list shows the following number in order from
1231             highest to lowest?",  

1232         "Answer": "C",
1233         "Counterfactual Answer": "D",
1234         "Choice_A": " 235 237 254 276 ",  

1235         "Choice_B": " 237 276 235 254 ",  

1236         "Choice_C": " 276 254 237 235 ",  

1237         "Choice_D": " 276 254 235 237 ",  

1238         "Rationale_C": "Correctly ordered the values from greatest to
1239             least: 276, 254, 237, 235.",
1240         "Counterfactual Rationale": "Switched last 2 numbers."
1241     }
1242 }
```

Take the first decomposed sample as an example showing **MalAlgoQA Causality & Causal Graph**.

```

1240 {
1241     "factual_roles": {
1242         "Exposure": ["Ordered from greatest to least"],
```

```

1242     "Covariate": ["Number set (276, 254, 237, 235)"],
1243     "Mediator": ["Descending comparison logic"],
1244     "Outcome": ["Choice C"]
1245   },
1246   "counterfactual_roles": {
1247     "Exposure": ["Ordered least to greatest"],
1248     "Covariate": ["Number set (276, 254, 237, 235)"],
1249     "Mediator": ["Ascending comparison logic"],
1250     "Outcome": ["Choice A"]
1251   },
1252   "causal_graph": {
1253     "factual_edges": [
1254       ["Number set (276, 254, 237, 235)", "Ordered from greatest to least
1255       ""],
1256       ["Number set (276, 254, 237, 235)", "Descending comparison logic"],
1257       ["Number set (276, 254, 237, 235)", "Choice C"],
1258       ["Ordered from greatest to least", "Descending comparison logic"],
1259       ["Descending comparison logic", "Choice C"],
1260       ["Ordered from greatest to least", "Choice C"]
1261     ],
1262     "counterfactual_edges": [
1263       ["Number set (276, 254, 237, 235)", "Ordered least to greatest"],
1264       ["Number set (276, 254, 237, 235)", "Ascending comparison logic"],
1265       ["Number set (276, 254, 237, 235)", "Choice A"],
1266       ["Ordered least to greatest", "Ascending comparison logic"],
1267       ["Ascending comparison logic", "Choice A"],
1268       ["Ordered least to greatest", "Choice A"]
1269     ]
1270   }
1271 }

```

HumanEval-Exe Example This dataset performs a programming-related task: code execution. It is designed to probe the ability of code-execution language models to perform counterfactual reasoning in the context of program behavior. Each example consists of a function definition and a test case input, and the model is asked to predict the output under a factual assumption (e.g., Python’s default 0-based indexing). The example is paired with a counterfactual version of the same test case, where a hypothetical condition is introduced—such as switching to 1-based indexing. The model must then predict the corresponding counterfactual output. For instance, given a function that checks for close floating-point elements in a list, the model is expected to reason whether the list and threshold would yield a different outcome if indexing conventions were altered. The full example is shown as follows:

```

1277 {
1278   "instruction": "from typing import List\n\n\nndef has_close_elements(
1279     numbers: List[float], threshold: float) -> bool:\n      \"\"\"\n      Check if in given list of numbers, are any two numbers closer to\n      each other than\n      given threshold.\n      >>> has_close_elements\n      ([1.0, 2.0, 3.0], 0.5)\n      False\n      >>> has_close_elements([1.
1280      0, 2.8, 3.0, 4.0, 5.0, 2.0], 0.3)\n      True\n      \"\"\"",
1281   "input": "[1.0, 2.0, 3.9, 4.0, 5.0, 2.2], 0.3",
1282   "output": "True",
1283   "counterfactual_output": "True"
1284 }

```

HumanEval-Exe Causality & Causal Graph.

```

1285 {
1286   "factual_roles": {
1287     "Exposure": ["0-based indexing"],
1288     "Covariate": [
1289       "List values [1.0, 2.0, 3.9, 4.0, 5.0, 2.2]",
1290       "Threshold 0.3",
1291       "Pairwise comparison algorithm"
1292     ],
1293     "Mediator": ["Range iteration logic (0 <= i < j < len(numbers))"]
1294   }
1295 }

```

```

1296     "Outcome": ["True"]
1297   },
1298   "counterfactual_roles": {
1299     "Exposure": ["1-based indexing"],
1300     "Covariate": [
1301       "List values [1.0, 2.0, 3.9, 4.0, 5.0, 2.2]",
1302       "Threshold 0.3",
1303       "Pairwise comparison algorithm"
1304     ],
1305     "Mediator": ["Range iteration logic (1 <= i < j <= len(numbers))"],
1306     "Outcome": ["True"]
1307   },
1308   "causal_graph": {
1309     "factual_edges": [
1310       ["List values [...]", "0-based indexing"],
1311       ["Threshold 0.3", "0-based indexing"],
1312       ["Pairwise comparison algorithm", "0-based indexing"],
1313       ["0-based indexing", "Range iteration logic (0 <= i < j < len(
1314         numbers))"],
1315       ["Range iteration logic (0 <= i < j < len(numbers))", "True"],
1316       ["List values [...]", "True"],
1317       ["Threshold 0.3", "True"]
1318     ],
1319     "counterfactual_edges": [
1320       ["List values [...]", "1-based indexing"],
1321       ["Threshold 0.3", "1-based indexing"],
1322       ["Pairwise comparison algorithm", "1-based indexing"],
1323       ["1-based indexing", "Range iteration logic (1 <= i < j <= len(
1324         numbers))"],
1325       ["Range iteration logic (1 <= i < j <= len(numbers))", "True"],
1326       ["List values [...]", "True"],
1327       ["Threshold 0.3", "True"]
1328     ]
1329   }
1330 }
```

Open-Critic Example This dataset performs a programming-related task: code generation. It is a synthetic code editing benchmark designed to train and evaluate large language models on their ability to identify and fix bugs in code. Each example consists of a natural language *task description*, a *correct code* solution that satisfies the task, and a *counterfactual explanation* that describes bugs introduced into a similar but faulty implementation. The objective is for the model to learn to generate or repair the *bugged version of the code* based on the bug descriptions as a counterfactual scenario. For instance, given a correct nested loop that prints all valid digit combinations excluding the number 5 and repetitions, the counterfactual explanation points out altered range values (e.g., using `range(100)` and `range(1)`), leading to an incomplete iteration logic. The full example is shown as follows:

```

1337 {
1338   "task": "Create a nested loop to print every combination of numbers
1339   between 0-9, excluding any combination that contains the number
1340   5. Additionally, exclude any combination that contains a
1341   repeating digit. Implement the solution without using any built-
1342   in functions or libraries to check for repeating digits.",
1343   "correct_code": "for i in range(10): # First digit\n    for j in
1344     range(10): # Second digit\n        for k in range(10): # Third digit\
1345           # Checking for the conditions\n            if i != 5 and j != 5 and k
1346           != 5 and i != j and i != k and j != k:\n              print(i, j, k)",
1347   "correct_explanation": "This code will generate and print every
1348   combination of three digits between 0-9 that do not contain the
1349   number 5 and do not have any repeating digits.",
1350   "bugged_code": "for i in range(10): \n    for j in range(100): \n
1351       for k in range(1): \n           if i != 5 and j != 5 and k != 5 and i !=
1352           j and i != k and j != k:\n               print(i, j, k)",
1353 }
```

```

1350
1351     "counterfactual_explanation": "Reading through this code, I notice
1352         that:\n\n* The second loop is set to 'range(100)' instead of 'range(10)', which means it will only iterate over the numbers
1353         from 0 to 99. This will cause the script to miss some
1354         combinations.\n* The third loop is set to 'range(1)', which means
1355             it will only iterate over the number 0. This will cause the
1356             script to only print combinations with one digit, instead of
1357             three.\n\nThese bugs will prevent the script from generating and
1358             printing all possible combinations of three digits between 0-9
1359             that do not contain the number 5 and do not have any repeating
1360             digits.\n\nTips for avoiding these mistakes:\n\n* Double-check
1361             the range values in each loop to ensure they are correct.\n* Make
1362             sure the loops iterate correctly over the desired range of
1363             values."
1364
1365 }
```

Open-Critic Causality & Causal Graph.

```

1366 {
1367     "factual_roles": {
1368         "Exposure": ["Correct explanation (valid ranges and checks)"],
1369         "Covariate": [
1370             "Task requirements (0-9 digits)",
1371             "Exclusion logic (no 5/repeats)",
1372             "Nested loop structure"
1373         ],
1374         "Mediator": ["Proper range initialization (range(10) x3)"],
1375         "Outcome": ["Correct triple-nested loop code"]
1376     },
1377     "counterfactual_roles": {
1378         "Exposure": ["Counterfactual explanation (invalid ranges)"],
1379         "Covariate": [
1380             "Task requirements (0-9 digits)",
1381             "Exclusion logic (no 5/repeats)",
1382             "Nested loop structure"
1383         ],
1384         "Mediator": [
1385             "Flawed range parameters (range(100)/range(1))",
1386             "Incomplete digit iteration"
1387         ],
1388         "Outcome": ["Bugged code with limited iterations"]
1389     },
1390     "causal_graph": {
1391         "factual_edges": [
1392             ["Task requirements", "Correct explanation"],
1393             ["Exclusion logic", "Correct explanation"],
1394             ["Nested loop structure", "Correct explanation"],
1395             ["Task requirements", "Proper range initialization"],
1396             ["Exclusion logic", "Proper range initialization"],
1397             ["Nested loop structure", "Proper range initialization"],
1398             ["Correct explanation", "Correct triple-nested loop code"],
1399             ["Task requirements", "Correct triple-nested loop code"],
1400             ["Exclusion logic", "Correct triple-nested loop code"],
1401             ["Correct explanation", "Proper range initialization"],
1402             ["Proper range initialization", "Correct triple-nested loop
1403             code"],
1404             ["Correct explanation", "Correct triple-nested loop code"]
1405         ],
1406         "counterfactual_edges": [
1407             ["Task requirements", "Flawed range parameters"],
1408             ["Exclusion logic", "Flawed range parameters"],
1409             ["Nested loop structure", "Flawed range parameters"],
1410             ["Task requirements", "Bugged code with limited iterations"],
1411             ["Exclusion logic", "Bugged code with limited iterations"]
1412         ]
1413     }
1414 }
```

```

1404     ["Nested loop structure", "Bugged code with limited iterations"
1405         ],
1406     ["Counterfactual explanation", "Flawed range parameters"],
1407     ["Flawed range parameters", "Incomplete digit iteration"],
1408     ["Incomplete digit iteration", "Bugged code with limited
1409         iterations"],
1410     ["Counterfactual explanation", "Bugged code with limited
1411         iterations"]
1412   ]
1413 }

```

1414

Code-Preference Example This dataset performs a programming-related task: code summarization. It contains pairs of duplicate code examples, with the only difference being the bugged code example has the bugged code 'surgically transplanted in' while the corrected code is left the same. Each example consists of a natural language *instruction*, a *correct code* solution that satisfies the instruction, and a *bug explanation* that describes bugs. The objective is for the model to learn to summarize and generate the bug descriptions as a counterfactual scenario. For instance, given a correct nested loop that prints all valid digit combinations excluding the number 5 and repetitions compared with a bugged loop, the bug description generated by the model will be able to point out altered range values (e.g., using `range(100)` and `range(1)`) leading to an incomplete iteration logic in its summarized response of bug explanation. The full example is shown as follows:

1424

1425

```

{
  "Instruction": 'Create a nested loop to print every combination of
  numbers between 0-9, excluding any combination that contains the
  number 5. Additionally, exclude any combination that contains a
  repeating digit. Implement the solution without using any built-
  in functions or libraries to check for repeating digits.',
  "bugged_code": 'What are the problems with this code? '''\npython\
  nfor i in range(10): \n    for j in range(100): \n        for k in
  range(1): \n            if i != 5 and j != 5 and k != 5 and i != j and i !=
  k and j != k:\n                print(i, j, k)\n''',
  "bug_explanation": 'Reading through this code, I notice that:\n\n*
  The second loop is set to `range(100)` instead of `range(10)`, which
  means it will only iterate over the numbers from 0 to 99. This
  will cause the script to miss some combinations.\n*
  The third loop is set to `range(1)`, which means it will only iterate
  over the number 0. This will cause the script to only print
  combinations with one digit, instead of three.\n\nThese bugs will
  prevent the script from generating and printing all possible
  combinations of three digits between 0-9 that do not contain the
  number 5 and do not have any repeating digits.\n\nTips for
  avoiding these mistakes:\n*
  Double-check the range values in each loop to ensure they are correct.\n*
  Make sure the loops
  iterate correctly over the desired range of values.\n\nHere is
  the corrected code:\n\n```python\nfor i in range(10): # First
  digit\n    for j in range(10): # Second digit\n        for k in range
  (10): # Third digit\n            # Checking for the conditions\n            if i
  != 5 and j != 5 and k != 5 and i != j and i != k and j != k:\n                print(i, j, k)\n```\n',
  "correct_code": 'Here is an example of a nested loop in Python to
  print every combination of numbers between 0-9, excluding any
  combination that contains the number 5 or repeating digits:\n\n```
  python\nfor i in range(10): # First digit\n    for j in range
  (10): # Second digit\n        for k in range(10): # Third digit\n            # Checking for the conditions\n            if i != 5 and j != 5 and k != 5
  and i != j and i != k and j != k:\n                print(i, j, k)\n```\n\nThis
  code will generate and print every combination of three
  digits between 0-9 that do not contain the number 5 and do not
  have any repeating digits.'
}

```

1457

1458 **Code-Preference Causality & Causal Graph.**
 1459

```

1460 {
1461   "factual_roles": {
1462     "Exposure": ["Correct code (range(10) loops)"],
1463     "Covariate": [
1464       "Task requirements (0-9 digits)",
1465       "Exclusion logic (no 5/repeats)",
1466       "Nested loop structure"
1467     ],
1468     "Mediator": ["Proper range initialization (range(10) x3)"],
1469     "Outcome": ["Correct explanation (valid ranges and checks)"]
1470   },
1471   "counterfactual_roles": {
1472     "Exposure": ["Bugged code (range(100)/range(1))"],
1473     "Covariate": [
1474       "Task requirements (0-9 digits)",
1475       "Exclusion logic (no 5/repeats)",
1476       "Nested loop structure"
1477     ],
1478     "Mediator": ["Flawed range parameters", "Incomplete digit iteration"]
1479   },
1480   "causal_graph": {
1481     "factual_edges": [
1482       ["Task requirements (0-9 digits)", "Correct code (range(10) loops)"],
1483       ["Exclusion logic (no 5/repeats)", "Correct code (range(10) loops)"],
1484       ["Nested loop structure", "Correct code (range(10) loops)"],
1485       ["Task requirements (0-9 digits)", "Proper range initialization (range(10) x3)"],
1486       ["Exclusion logic (no 5/repeats)", "Proper range initialization (range(10) x3)"],
1487       ["Nested loop structure", "Proper range initialization (range(10) x3)"],
1488       ["Task requirements (0-9 digits)", "Correct explanation (valid ranges and checks)"],
1489       ["Exclusion logic (no 5/repeats)", "Correct explanation (valid ranges and checks)"],
1490       ["Correct code (range(10) loops)", "Proper range initialization (range(10) x3)"],
1491       ["Proper range initialization (range(10) x3)", "Correct explanation (valid ranges and checks)"],
1492       ["Correct code (range(10) loops)", "Correct explanation (valid ranges and checks)"]
1493     ],
1494     "counterfactual_edges": [
1495       ["Task requirements (0-9 digits)", "Flawed range parameters"],
1496       ["Task requirements (0-9 digits)", "Incomplete digit iteration"],
1497       ["Task requirements (0-9 digits)", "Bug explanation (incorrect ranges analysis)"],
1498       ["Exclusion logic (no 5/repeats)", "Flawed range parameters"],
1499       ["Exclusion logic (no 5/repeats)", "Incomplete digit iteration"],
1500       ["Exclusion logic (no 5/repeats)", "Bug explanation (incorrect ranges analysis)"],
1501       ["Nested loop structure", "Flawed range parameters"],
1502       ["Nested loop structure", "Incomplete digit iteration"],
1503       ["Nested loop structure", "Bug explanation (incorrect ranges analysis)"],
1504       ["Bugged code (range(100)/range(1))", "Flawed range parameters"],
1505       ["Flawed range parameters", "Incomplete digit iteration"],
1506       ["Incomplete digit iteration", "Bug explanation (incorrect ranges analysis)"]
1507     ]
1508   }
1509 }
1510 
```

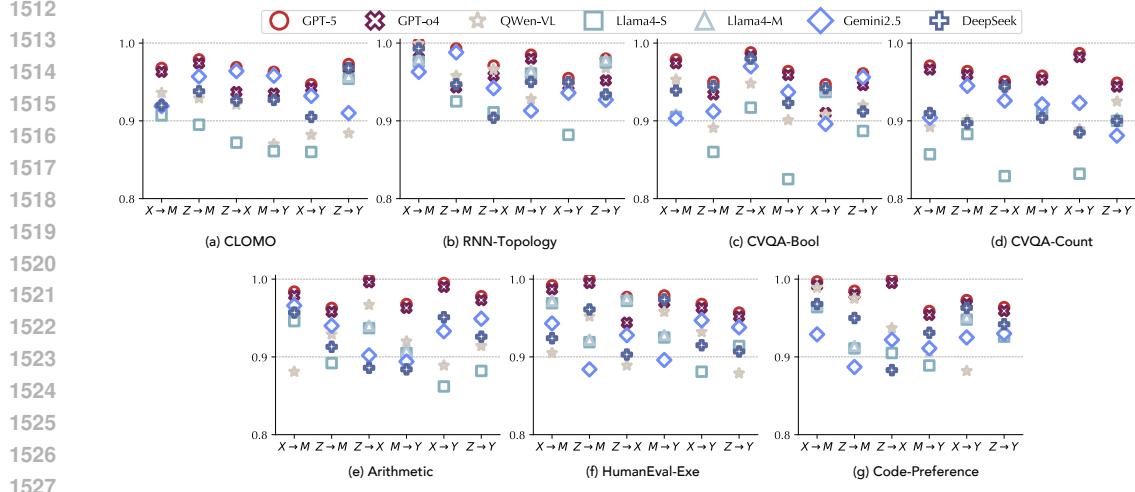


Figure 5: Additional evaluation on causal graph construction, complementing to Figure 3.

```

1531      ["Bugged code (range(100)/range(1))", "Bug explanation (incorrect
1532      ranges analysis)"]
1533  ]
1534  }
1535 }
```

B COMPLEMENTARY EXPERIMENT

B.1 SETTING

This section lists the experimental settings used in this study.

Table 7: LLM query hyperparameters used during all experiments.

Hyperparameter	Value	Description
Temperature	0.7	Controls randomness in generation
Top- p (nucleus sampling)	0.95	Probability mass for sampling
Max tokens	2048	Maximum number of tokens to generate
Stop sequences	["\n", "Q:"]	Used to truncate responses
Prompt format	CoT, CoT-SC, ToT	Prompting strategy used in Section 4.2.2
Tool-calling API	Enabled (Selective)	Used in tool-augmented experiments

Computational Resources. All experiments were conducted on a high-performance computing server equipped with six NVIDIA RTX 6000 Ada Generation GPUs, each with 49 GB of dedicated VRAM. The system utilized CUDA version 12.8 and NVIDIA driver version 570.124.06. These GPUs supported parallel execution of model querying, evaluation, and tool-augmented tasks across our benchmark datasets. The hardware configuration ensured sufficient memory bandwidth and processing capability to accommodate large-scale inference, particularly for multimodal tasks and multi-sample prompting strategies such as CoT-SC and ToT. No resource-related constraints were encountered during experimentation.

B.2 ADDITIONAL RESULT AND DISCUSSION

This part presents additional experimental results that complement the main evaluation in the body of the paper. These supplementary findings, together with what has been presented in previous sections, offer comprehensive insights into LLMs capabilities across different decompositional tasks.

1566

1567 Table 8: LLMs’ performance (F1 standard deviations, scaled to 100%) in causal variable identification
1568 (reordered; adjusted variability).

Dataset	GPT-5		GPT-o4		Qwen3		Llama4-S		Llama4-M		Gemini2.5		DeepSeek	
	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2	v_1	v_2
$v_1 = X$ (Exposure), $v_2 = Z$ (Covariate)														
CRASS	4.6	4.9	9.1	9.9	5.6	6.2	8.3	7.9	3.5	4.4	4.5	5.3	4.8	5.0
CLOMO	4.8	5.1	9.3	10.0	6.8	7.2	9.6	8.9	5.0	5.5	5.9	6.5	5.3	5.7
RNN-Topo	5.0	5.3	9.5	10.2	7.3	7.8	10.2	9.5	5.4	5.9	6.4	6.9	5.8	6.2
CVQA-Bool	5.4	5.7	9.8	10.4	9.7	10.2	12.4	13.1	8.9	9.4	10.8	11.5	9.4	10.1
CVQA-Count	5.6	5.9	9.9	10.5	10.6	11.3	13.7	14.2	9.8	10.5	11.5	12.3	10.3	11.2
COCO	5.8	6.1	10.0	10.6	11.9	12.6	15.2	15.8	11.1	11.8	12.9	13.7	11.6	12.5
Arithmetic	5.2	5.5	9.4	10.1	8.9	9.5	11.3	12.0	7.5	8.1	8.4	9.1	7.9	8.6
MalAlgoQA	5.3	5.6	9.6	10.3	9.5	10.1	12.2	12.9	8.2	8.7	9.0	9.7	8.5	9.2
HumanEval-Exe	6.0	6.3	10.2	10.8	12.8	13.6	16.3	17.1	11.7	12.4	13.9	14.7	12.5	13.4
Open-Critic	6.1	6.4	10.3	10.9	13.5	14.2	17.1	17.8	12.6	13.3	14.6	15.3	13.7	14.5
Code-Preference	6.0	6.2	10.1	10.7	13.1	13.9	16.7	17.4	12.2	12.9	14.3	15.0	13.1	13.9
$v_1 = M$ (Mediator), $v_2 = Y$ (Outcome)														
CRASS	4.9	5.2	9.2	10.0	9.6	7.4	12.1	9.7	8.5	6.1	9.1	6.8	8.7	6.2
CLOMO	5.0	5.3	9.4	10.2	10.5	8.1	12.9	10.3	9.2	6.6	9.8	7.5	9.3	6.8
RNN-Topo	5.1	5.4	9.6	10.3	11.1	8.5	13.6	10.8	9.6	7.0	10.5	7.9	9.8	7.2
CVQA-Bool	5.7	6.0	10.0	10.6	13.8	11.3	16.9	14.2	12.4	10.0	14.1	12.5	13.2	10.7
CVQA-Count	5.9	6.2	10.1	10.7	14.6	12.1	17.8	15.1	13.3	10.8	15.0	13.3	14.1	11.6
COCO	6.0	6.3	10.2	10.8	15.3	12.8	18.5	15.9	14.0	11.5	15.7	14.1	14.8	12.4
Arithmetic	5.4	5.7	9.5	10.2	12.5	9.8	15.4	12.5	11.0	8.6	13.1	9.2	11.3	8.7
MalAlgoQA	5.6	5.9	9.7	10.4	13.1	10.4	16.1	13.2	11.6	9.2	13.8	10.0	12.0	9.3
HumanEval-Exe	6.2	6.5	10.4	11.0	16.2	13.6	19.3	16.7	14.8	12.3	16.5	14.8	15.5	13.1
Open-Critic	6.3	6.6	10.5	11.1	16.9	14.4	18.1	17.5	15.5	13.0	17.2	15.5	16.1	13.9
Code-Preference	6.1	6.4	10.3	10.9	16.5	14.0	19.7	17.0	15.2	12.6	16.9	15.1	15.8	13.5

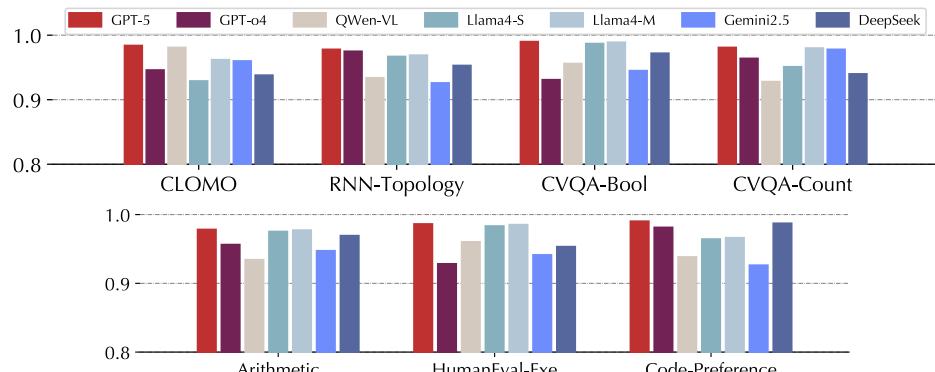


Figure 6: Additional evaluation of LLMs’ intervention identification, complementing to Figure 4

To deepen our understanding of how LLMs handle counterfactual reasoning, we analyze representative datasets that cover text, multimodal, symbolic, and code-based modalities. We aim to uncover impediments at each decompositional stage. Below, we highlight dataset-level characteristics that either enable or hinder performance.

Textual Question-Answering and Logic Parsing. Datasets built on natural language (e.g., textual QA and logical modification tasks) generally facilitate the recognition of explicit causal variables such as exposures and outcomes. Models can easily link a stated intervention (“if tutoring was not provided”) to its corresponding variable. However, mediators are often described abstractly (through latent constructs like “trust,” “motivation,” or “belief states”) rather than explicit entities. This makes Task I disproportionately difficult for mediators compared to exposures or outcomes. Errors at this stage then propagate to Task IV, where the model must simulate how such mediators would affect outcomes under intervention. Even when causal graph construction (Task II) is near-perfect due to the

1620

1621 Table 9: LLM performance (F1 standard deviation) in reasoning the counterfactual mediator (M')
1622 and outcome (Y').

Dataset	GPT-5		GPT-o4		Qwen3		Llama4-S		Llama4-M		Gemini2.5		DeepSeek	
	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'	M'	Y'
CRASS	4.2	5.0	4.8	5.6	6.7	8.5	10.0	12.5	5.0	7.1	6.3	8.1	5.5	7.4
CLOMO	4.4	5.2	5.0	5.9	7.2	9.0	10.6	13.0	5.4	7.5	6.7	8.6	6.0	7.9
RNN-Topo	4.6	5.4	5.1	6.0	7.8	9.5	11.0	13.5	5.6	7.7	7.2	9.1	6.5	8.4
CVQA-Bool	4.8	5.6	5.2	6.1	11.2	13.3	14.5	16.8	9.0	11.1	12.1	14.0	10.4	12.5
CVQA-Count	5.0	5.8	5.3	6.3	12.0	14.1	15.2	17.6	9.8	11.9	12.9	14.9	11.2	13.3
COCO	5.1	6.0	5.4	6.4	12.7	14.8	15.9	18.3	10.5	12.7	13.6	15.6	11.9	14.0
Arithmetic	4.5	5.2	5.1	6.0	9.5	11.6	12.8	15.3	7.4	9.5	9.0	11.1	8.3	10.4
MaIAlgQA	4.7	5.3	5.2	6.1	10.1	12.2	13.4	15.9	7.9	10.0	9.7	11.8	8.9	11.0
HumanEval-Exe	5.2	6.2	5.5	6.5	13.3	15.4	16.5	18.9	11.0	13.2	14.2	16.2	12.5	14.7
Open-Critic	5.3	6.3	5.6	6.6	13.9	16.0	17.0	19.6	11.6	13.8	14.8	16.9	13.1	15.3
Code-Preference	5.1	6.1	5.5	6.5	13.6	15.7	16.7	19.2	11.3	13.5	14.5	16.5	12.8	15.0

1634

1635

1636

1637 structured nature of logical relations, the absence of explicit mediators limits downstream reasoning
1638 fidelity.1639 **Vision-Language Counterfactuals.** Multimodal datasets combining images with text pose unique
1640 challenges. When asked to identify causal variables, LLMs must ground textual descriptions in
1641 visual objects. For example, distinguishing “presence of a ball” (exposure) from “action of kicking”
1642 (mediator) requires fine-grained alignment of object attributes with causal semantics. This grounding
1643 step introduces errors in Task I, especially when visual scenes are cluttered or ambiguous. Even
1644 when interventions (Task III) are identified correctly, outcome reasoning (Task IV) suffers because
1645 models struggle to propagate visual changes into numerical or behavioral predictions. For instance,
1646 recognizing that “removing a ball” should reduce the count of possible goals involves chaining
1647 visual detection, counting logic, and causal propagation—steps that current LLMs rarely integrate
1648 coherently.1649 **Symbolic and Mathematical Reasoning.** Datasets built on arithmetic or algorithmic transformations
1650 highlight another bottleneck: reliance on memorized patterns instead of causal mechanisms. In
1651 variable identification (Task I), explicit quantities are correctly recognized. In causal graph con-
1652 struction (Task II), rules linking operations (e.g., “base conversion influences the final number”) are
1653 applied consistently. However, in outcome reasoning (Task IV), models frequently fail to simulate
1654 the correct causal pathway, often defaulting to template-based responses rather than computing the
1655 actual counterfactual result. This suggests that while symbolic data supports high precision in explicit
1656 structure, it exposes the weakness of LLMs in mechanistic simulation of causal processes.1657 **Code-Based Reasoning.** Programming-oriented datasets such as code execution, code generation,
1658 or preference tasks are particularly difficult across all stages. In Task I, identifying exposures and
1659 outcomes is hindered by the abstractness of programming constructs (e.g., “function signature” as
1660 exposure, “program output” as outcome). Mediators (such as intermediate execution states) are even
1661 harder to capture, as they are not explicitly represented in the code text but must be inferred from
1662 semantics. In Task II, while models can generate plausible causal graphs describing dependencies
1663 among variables or functions, these graphs often overgeneralize or miss critical execution details.
1664 Task IV is especially challenging: even when interventions like “changing a loop to recursion” are
1665 recognized, LLMs often fail to simulate downstream program behavior, producing outcomes that
1666 are logically plausible but incorrect. This reflects a persistent gap between syntactic recognition and
1667 semantic reasoning.1668 **Cross-Cutting Observations.** Across modalities, two key impediments recur:1669 (1) *Complex modalities impede variable identification.* Images and code introduce higher error rates
1670 in Task I, since grounding or semantic parsing must precede causal reasoning.1671 (2) *Implicit mediators bottleneck outcome reasoning.* Regardless of modality, when mediators are
1672 abstract or not explicitly present, performance in Task IV drops substantially. LLMs can identify
1673 interventions reliably, but they fail to propagate their effects along causal chains to yield consistent
counterfactual outcomes.

1674 These findings suggest that LLMs are “eligible” for decompositional reasoning in structured settings
 1675 with explicit causal variables (e.g., clean text or arithmetic). However, when confronted with modality
 1676 complexity or implicit causal pathways, their reasoning capacity is significantly impeded.
 1677

1678 B.3 WORKING MEMORY PERSPECTIVE TO INTERPRET COUNTERFACTUAL REASONING 1679

1680 Prior research (Zhang et al., 2024a) has demonstrated that language models exhibit notable difficulty
 1681 in temporally storing and manipulating information even in n-back tasks that are cognitively simpler
 1682 than explicit reasoning. This underlying limitation in working memory capacity poses constraints
 1683 on long-term and multi-step reasoning. To explore the connection between memory bottlenecks and
 1684 mediator identification challenges, we conducted additional experiments in more depth.

1685 **Experiments on Working Memory.** To examine how working memory affects mediator reasoning,
 1686 we designed a controlled n-back Mediator Recall task. While most benchmarks involve only
 1687 single-step mediation, the Open-Critic dataset (code modality) includes examples of multi-step causal
 1688 mediation. For instance, adjusting code inputs requires reasoning over prior inputs and transformations.
 1689 In this task, the mediator must be inferred from causal variables presented n steps earlier in the
 1690 input. We vary n from 1 to 3 and report F1 scores consistent with Table 10.

1692 1693 Table 10: LLM performance (F1) in n-back mediator recall

Model	1-hop	2-hop	3-hop
GPT-o4	72.2%	63.5%	9.7%
Qwen	58.3%	39.6%	12.1%
Gemini	66.4%	26.1%	7.5%
LLaMA4-Scout	45.5%	47.2%	3.6%

1700 **Findings and insights.** These results reveal a sharp performance drop as the number of intermediate
 1701 steps increases. From a working memory perspective, this suggests that current LLMs struggle to
 1702 retain or reconstruct causal paths to mediators when they are separated by multiple reasoning hops.
 1703 This degradation highlights a key constraint in long-horizon causal reasoning.

1704 These findings align with our earlier observation that mediator reasoning is a consistent bottleneck
 1705 in decompositional analysis. By framing this in terms of working memory capacity, we offer a
 1706 mechanistic explanation for why LLMs falter on such tasks and why enhanced memory mechanisms
 1707 (e.g., intermediate supervision or tool-assisted retrieval) may be necessary for progress.

1709 1710 B.4 INFLUENCE OF MODEL SCALE

1712 1713 Table 11: LLM performance in reasoning variables (X, Z, M, Y, M', Y').

Dataset	Qwen-VL-2B						Qwen-VL-4B						Qwen-VL-8B					
	X	Z	M	Y	M'	Y'	X	Z	M	Y	M'	Y'	X	Z	M	Y	M'	Y'
CRASS	44.2	27.1	20.3	37.2	18.2	16.5	47.0	29.5	22.1	39.4	20.3	18.7	52.8	34.4	26.7	44.5	25.8	23.6
CLOMO	23.8	18.4	11.6	15.3	8.3	5.1	26.0	20.2	13.0	17.0	9.6	6.4	30.5	24.0	16.5	20.8	12.5	9.3
CVQA-Count	23.1	16.2	15.6	18.4	11.5	7.4	25.4	18.0	17.1	20.3	13.1	8.9	29.7	21.3	20.4	23.8	16.4	11.8
MalAlgoQA	17.2	14.0	10.5	14.6	8.5	6.2	19.1	15.6	12.0	16.3	10.2	7.6	23.4	19.3	15.7	19.7	13.6	10.5

1718 **Besides comparing GPT and Llama families, we further conduct a controlled scaling study over the**
 1719 **Qwen-VL series to examine how model size influences both causal variable identification and down-**
 1720 **stream counterfactual reasoning. Table 11 summarizes the performance of Qwen-VL-2B, 4B, and**
 1721 **8B across factual variables (X, Z, M, Y) and counterfactual targets (M', Y'). Overall, we observe**
 1722 **a clear but non-uniform scaling trend. Moving from 2B to 4B yields modest and consistent gains,**
 1723 **particularly on explicit variables (X, Z, Y), suggesting that moderate scaling primarily improves**
 1724 **surface-level grounding and extraction. In contrast, the 8B model shows more noticeable improve-**
 1725 **ments across both explicit and implicit variables, including the more challenging mediators (M, M')**
 1726 **and counterfactual outcomes (Y'). These gains indicate that larger Qwen models are better able to**
 1727 **propagate interventions through the underlying causal structure rather than solely memorizing lexical**

1728 associations. However, even the 8B model retains substantial gaps on tasks requiring multi-hop causal
1729 reasoning, especially when mediators are implicit or visually grounded. Taken together, the scaling
1730 analysis suggests that increased model capacity helps alleviate some of the bottlenecks identified in
1731 smaller models, but does not by itself resolve the fundamental challenges of counterfactual mediation
1732 and outcome inference.

1733 C LARGE LANGUAGE MODEL (LLM) USAGE DISCLOSURE

1734 Large language models were used only for minor grammar revision and sentence-level polishing
1735 during manuscript preparation. They were not employed in ideation, methodological design, ex-
1736 perimental execution, or result analysis. The scientific contributions, benchmarks, and evaluations
1737 presented in this work were entirely conceived and developed by the authors. LLM involvement was
1738 minimal in the research process.

1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781