

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 HOW KNOWLEDGE POPULARITY INFLUENCES AND ENHANCES LLM KNOWLEDGE BOUNDARY PERCEP- TION

006
007
008
009
010
011
012
Anonymous authors
007
Paper under double-blind review

ABSTRACT

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
Large language models (LLMs) often fail to recognize their knowledge boundaries, producing confident yet incorrect answers. In this paper, we investigate how knowledge popularity affects LLMs’ ability to perceive their knowledge boundaries. Focusing on entity-centric factual question answering (QA), we quantify knowledge popularity from three perspectives: the popularity of entities in the question, the popularity of entities in the answer, and relation popularity, defined as their co-occurrence frequency. Experiments on three representative datasets containing knowledge with varying popularity show that LLMs exhibit better QA performance, higher confidence, and more accurate perception on more popular knowledge, with relation popularity having the strongest correlation. Cause knowledge popularity shows strong correlation with LLMs’ QA performance, we propose to leverage these signals for confidence calibration. This improves the accuracy of answer correctness prediction by an average of 5.24% across all models and datasets. Furthermore, we explore prompting LLMs to estimate popularity without external corpora, which yields a viable alternative.

1 INTRODUCTION

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
Large language models (LLMs) (Achiam et al., 2023; Yang et al., 2024; Dubey et al., 2024) often hallucinate—producing fluent but factually incorrect answers which is unacceptable in safety-critic domains such as healthcare. Accurately identifying when LLMs produce correct answers not only helps determine when to trust their outputs, but also enables adaptive retrieval-augmented generation (RAG)—performing retrieval only when they do not know the answer—thereby enhancing both the effectiveness and efficiency of RAG (Ni et al., 2024a). A reliable model should have a clear perception of its knowledge boundaries—knowing what it knows and what it does not. This requires its confidence in an answer, reflected in the generation probability, to align with the actual likelihood of the answer being correct (Jiang et al., 2021). While many studies have examined LLMs’ perception level of their knowledge boundaries and found that they tend to be overconfident (Lin et al., 2022; Tian et al., 2023), the underlying factors influencing the perception remain poorly understood.

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
A natural hypothesis is that a model’s perception level can be influenced by the popularity of the knowledge—i.e., how frequently the model has encountered it during training. When asked about popular knowledge, the model may be more likely to respond both correctly and confidently. Prior work (Mallen et al., 2023) has shown that LLMs achieve better QA performance on more popular questions. This raises a key question: how do the model’s confidence and its alignment with QA performance vary with knowledge popularity?

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
To investigate this, we focus on entity-centric factual QA (Mallen et al., 2023; Yuksekgonul et al., 2023) where both the question and the answer contain an entity because this enables us to quantify the popularity of knowledge based on entities. Specifically, we assess knowledge popularity from the following three perspectives: 1) *Question popularity* Pop_Q : popularity of the entity in the question. 2) *Ground-truth answer popularity* Pop_{GT} : popularity of the entity in the ground-truth answer. 3) *Ground-truth relation popularity* $RPop_{GT}$: the co-occurrence frequency of the question and ground-truth entities. Higher entity popularity suggests more accurate entity representations. Rela-

054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107
tion popularity directly influences the model’s ability to comprehend associations between entities, but is costly to collect.

Since ground-truth answers are unavailable in real-world scenarios, we also examine model-generated answers. Specifically, we investigate whether the popularity of the generated answer (Pop_{Ge}) and the relation popularity between the generated and question entities (RPop_{Ge}) reflect the model’s QA performance, confidence, and perception level. We focus in particular on their correlation with QA performance, as a strong correlation could allow these signals to be used for calibrating the model’s confidence.

We conduct experiments on three entity-centric factual QA datasets—Movies, Songs, and Basketball—constructed from Wikidata knowledge triplets by Yuksekgonul et al. (2023). Some question examples can be seen in Figure 1. We quantify entity popularity by the number of Wikidata language editions in which an entity appears. Relation popularity is measured by the number of Wikipedia documents where both entities are mentioned together. We use two representative open-source models—LLaMA3-8B-Instruct (Dubey et al., 2024) and Qwen2-7B-Instruct (Yang et al., 2024)—as well as the black-box model ChatGPT (Achiam et al., 2023).

Results on Pop_{Q} , Pop_{GT} , and RPop_{GT} show that *LLMs demonstrate better QA performance, higher confidence, and more accurate perception of their knowledge boundaries on more popular knowledge*. Although LLMs are generally overconfident, the extent of overconfidence diminishes as knowledge popularity increases, since QA performance improves more rapidly than confidence. Among the three popularity measures, RPop_{GT} shows the strongest correlation with QA accuracy, confidence, and perception level most of the cases. Interestingly, question popularity correlates more strongly with confidence than with QA performance, implying that LLMs may become overconfident simply due to familiarity with the question.

Regarding generated answers, RPop_{Ge} shows a strong positive correlation with QA performance, confidence, and perception level, while Pop_{Ge} exhibits a weaker correlation. Notably, RPop_{Ge} shows even stronger correlation with QA performance than RPop_{GT} , while Pop_{Ge} correlates more weakly than Pop_{GT} . We further analyze the reason and reveal that *when LLMs make errors, they tend to generate more popular entities that co-occur less frequently with the question entity compared to ground-truth answers*, indicating a tendency toward over-generalization. This is consistent with the findings of Zhang et al. (2024b).

Based on these findings, we propose to leverage popularity features (i.e., Pop_{Q} , Pop_{Ge} , and RPop_{Ge}) to calibrate confidence which aims to improve the effectiveness of confidence in predicting answer correctness. Given that computing knowledge popularity requires access to external corpora and incurs additional collection costs, we also investigate prompting the model to estimate popularity on its own. Results show that Pop_{Q} and Pop_{Ge} provide modest gains in calibration. In contrast, RPop_{Ge} provides substantial gains. *Combining all these three types of popularity yields the best calibration performance, boosting answer correctness prediction by an average of 5.24% across all models and datasets*. Moreover, leveraging model-estimated popularity also performs well for confidence calibration. The choice between external corpora and self-estimation ultimately hinges on the trade-off between performance and efficiency.

2 RELATED WORK

Existing research on model knowledge boundary perception focuses on assessing model confidence and can be mainly classified into four categories.

Probabilistic Confidence. This line of research treats the generation probability of the answer as the confidence of the model (Guo et al., 2017; Desai & Durrett, 2020; Jiang et al., 2021; Kadavath et al., 2022; Si et al., 2022; Kuhn et al., 2023). Guo et al. (2017) examined early neural networks (e.g., ResNet (He et al., 2016)) and found them to be overconfident, proposing temperature scaling as a remedy. Later, Desai & Durrett (2020) showed that BERT-style models tend to be relatively well-calibrated, while Jiang et al. (2021) found that pre-trained language models such as T5 (Raffel et al., 2020) remained overconfident. More recent work has turned to LLMs, with studies showing that they, too, exhibit overconfidence (Si et al., 2022; Lin et al., 2022; Tian et al., 2023).

108 **Verbalized Confidence.** LLMs have been shown to express their confidence verbally (Lin et al.,
 109 2022; Yin et al., 2023; Tian et al., 2023; Xiong et al., 2023; Yang et al., 2023; Ni et al., 2024a).
 110 Some studies (Yin et al., 2023; Ni et al., 2024a) found that LLMs often fail to recognize their
 111 knowledge limitations verbally and tend to be overconfident. Xiong et al. (2023) systematically
 112 studied black-box approaches for estimating LLM confidence. Beyond prompting-based methods,
 113 some studies aim to train LLMs to verbalize more accurate confidence (Lin et al., 2022; Yang et al.,
 114 2023; Zhang et al., 2024a).

115

116

117 **Consistency-based Confidence.** If the model is confident in its answer, it should maintain consistency
 118 across multiple generations. Recent studies have used self-consistency across generations as
 119 a proxy for LLM confidence (Manakul et al., 2023; Kuhn et al., 2023). Zhang et al. (2023) extended
 120 this by evaluating the consistency of answers across multiple semantically equivalent inputs and
 121 across different models. Ding et al. (2024) further adapted this approach to the multilingual setting.

122

123

124 **Confidence Estimation via LLM Internal States.** LLMs' internal states have shown to be effective
 125 in evaluating the factuality of their self-generated content (Su et al., 2024; Chen et al., 2024;
 126 Wang et al., 2024; Ni et al., 2025). Specifically, Su et al. (2024) and Chen et al. (2024) focused
 127 on internal states after generation, Wang et al. (2024) examined those before generation, and Ni
 128 et al. (2025) explored leveraging LLMs' internal states to enhance their perception of knowledge
 129 boundaries from efficiency and risk perspectives.

130

131

132

133

134

135

136

137

138

139 We focus on probabilistic confidence for the following reasons: 1) Both the model's generation
 140 probabilities and its knowledge acquisition arise from the same training objective, and are expected
 141 to align with each other. 2) Models without specialized training often struggle to verbalize confidence
 142 accurately (Ni et al., 2024b); consistency-based methods require multiple generations and incur high inference costs;
 143 and internal-state-based approaches require access to hidden representations and additional training. In contrast, probabilistic confidence is readily accessible and has been
 144 shown to perform well, especially when answers are short (Ding et al., 2024).

145

146

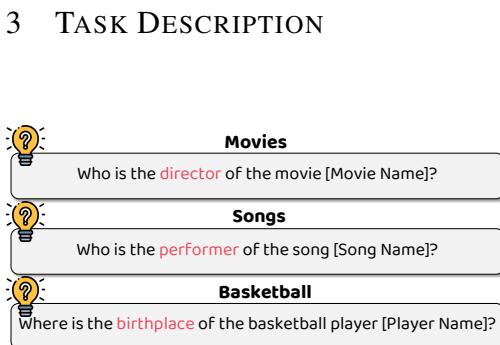
147

148

149

150

151



152 Figure 1: Question examples for each dataset.
 153

154

155

156

157

158

159

160

161

159 **Entity-Centric QA.** We focus on entity-centric knowledge because it allows us to measure
 160 knowledge popularity through entities. In entity-centric QA, questions and answers are derived
 161 from knowledge triples in the form of (subject, relation, object), where the question queries the
 162 relation of a given subject, and the model is expected to generate the corresponding object.
 163 Examples of knowledge triples are provided in Table 1, with their transformed question forms
 164 shown in Figure 1.

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

162 **4 HOW DOES KNOWLEDGE POPULARITY AFFECT LLMs’ PERCEPTION**
 163 **LEVEL?**
 164

165 In this section, we investigate how Pop_Q , Pop_{GT} , and RPop_{GT} influence LLMs’ QA performance,
 166 confidence, and perception level.
 167

168 Table 1: Data size for each dataset, along with the corresponding subject, relation, and object types.
 169

Datasets	Count	Subject	Relation	Object
Movies	10,964	Movie	Directed by	Director
Songs	2,157	Song	Performed by	Performer
Basketball	13,309	Player	Birthplace	City

175 Table 2: Definitions of notations about knowledge popularity where pop. means popularity.
 176

Notation	Definition
Pop_Q	Popularity of entities in the question
Pop_{GT}	Popularity of entities in the ground-truth answer
RPop_{GT}	Relation pop. between question and ground-truth entities
Pop_{Ge}	Popularity of entities in the generated answer
RPop_{Ge}	Relation pop. between question and generated entities

184 **4.1 EXPERIMENTAL SETUP**
 185

186 **Datasets.** Yuksekgonul et al. (2023) constructed entity-centric QA datasets based on Wikidata¹,
 187 using the number of sitelinks on a page as a proxy for entity popularity. They showed that this
 188 measure strongly correlates with an entity’s frequency in the training data. Building on this, we
 189 conduct experiments on their datasets to ensure reliable entity popularity measurement. We select
 190 three representative datasets—Movies, Song, and Basketball—because they exhibit clear differences
 191 in knowledge popularity. Specifically, question popularity ranks as Movies \downarrow Songs \downarrow Basketball,
 192 while ground-truth answer popularity follows Movies \downarrow Songs \downarrow Basketball. Table 1 lists the knowl-
 193 edge triplets and data counts for each dataset, and Figure 1 presents example questions. We apply
 194 data filtering to ensure reliable results, as detailed in Section §A.

195 **Entity Popularity.** Following Mallen et al. (2023); Yuksekgonul et al. (2023), we define the popu-
 196 larity of an entity by the number of sitelinks it has—i.e., the number of Wikipedia pages in different
 197 languages that link to it.

199 **Relation Popularity.** As Wikipedia is the primary high-quality source for Wikidata, we estimate
 200 relation popularity based on Wikipedia content. Specifically, for each entity pair, we measure rela-
 201 tion popularity by counting the number of documents in the Wikipedia dump² in which both entities
 202 co-occur. This reflects relation popularity in the model’s training data, as it shows a strong corre-
 203 lation with QA performance (see Table 3).

204 **LLMs.** We conduct experiments on three representative LLMs: two open-source models, Llama3-
 205 8B-Instruct (Dubey et al., 2024) and Qwen2-7B-Instruct (Yang et al., 2024), as well as a black-box
 206 model, ChatGPT (i.e., GPT-3.5-Turbo-1106) (Achiam et al., 2023).

208 **Answer Generation.** For all the models, we use greedy search, selecting the token with the highest
 209 probability at each generation step. An example can be seen in Figure 20.

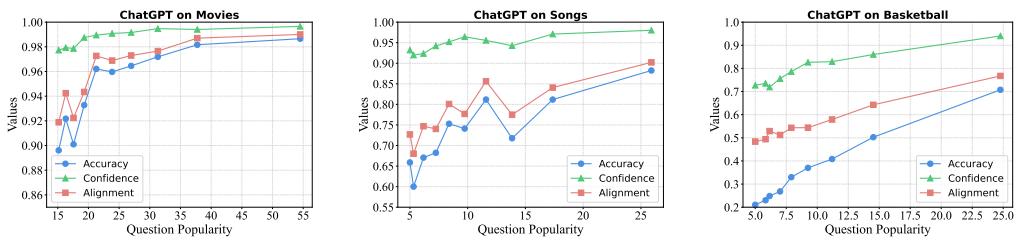
211 **Metrics.** For each question q_i , we measure answer correctness using accuracy acc_i , where the gen-
 212 erated answer is considered correct if it contains the ground-truth answer. The model’s confidence
 213 c_i is defined as the generation probability of the answer, as described in Section §3. Alignment is

215 ¹<https://query.wikidata.org/sparql>

²<https://huggingface.co/datasets/wikimedia/wikipedia>

216
217
218
Table 3: LLMs’ QA performance, confidence, alignment and the correlations between knowledge
popularity and accuracy, confidence, and alignment across different datasets.

Datasets	Models	Accuracy			Confidence			Alignment				
		Acc.	Pop _Q	Pop _{GT}	Conf.	Pop _Q	Pop _{GT}	RPop _{GT}	Align.	Pop _Q	Pop _{GT}	RPop _{GT}
Movies	Llama3	72.65	0.317	0.220	0.357	90.68	0.404	0.367	0.509	75.50	0.404	0.347
	Qwen2	42.85	0.433	0.299	0.494	82.32	0.413	0.371	0.507	53.63	0.386	0.279
	ChatGPT	94.78	0.134	0.069	0.130	98.80	0.210	0.230	0.280	94.85	0.211	0.228
Songs	Llama3	38.97	0.277	0.164	0.517	79.74	0.369	0.210	0.502	53.04	0.182	0.093
	Qwen2	25.82	0.362	0.255	0.541	78.00	0.300	0.200	0.345	42.97	0.230	0.180
	ChatGPT	73.36	0.171	0.266	0.399	94.84	0.249	0.295	0.381	75.28	0.232	0.340
Basketball	Llama3	13.37	0.118	0.293	0.231	60.09	0.173	0.063	0.055	46.21	-0.052	0.104
	Qwen2	9.90	0.014	0.348	0.151	74.76	0.151	0.076	0.009	32.35	0.126	0.189
	ChatGPT	34.89	0.288	0.215	0.353	79.06	0.351	0.054	0.270	50.43	0.201	0.164

228
229
230
231
232
233
234
235
Figure 2: QA performance, confidence, and alignment of ChatGPT across question popularity.

236
237
238
239
240 then computed as $1 - |acc_i - c_i|$. To examine the relationship between knowledge popularity and
241 QA performance, confidence, and alignment, we use Spearman correlation coefficients (Hauke &
242 Kossowski, 2011), which range from -1 to 1. The sign indicates the direction of the correlation,
243 while the absolute value reflects its strength.

244
4.2 RESULTS AND ANALYSIS
245

246 LLMs’ QA performance, confidence, and perception levels across different datasets, along with
247 the Spearman correlation coefficients between knowledge popularity and accuracy, confidence, and
248 alignment are shown in Table 3. We observe that:

249
250 **1) LLMs achieve better QA performance and higher confidence on more popular knowledge.**
251 All three types of popularity are positively correlated with both QA performance and confidence.
252 For QA performance and confidence, we conduct the following analyses respectively.

253 For QA performance, relation popularity generally shows the strongest correlation, as expected,
254 while question popularity exhibits a stronger correlation than answer popularity in most cases. This
255 suggests that learning through co-occurrence is especially effective for acquiring knowledge, and
256 that familiarity with the question contributes more to answering correctly than familiarity with the
257 answer. However, on the Basketball dataset, answer popularity shows the highest correlation for
258 both LLaMA3 and Qwen2. This dataset is challenging because both the question and relation have
259 low popularity. The models are often unfamiliar with the question entity and generate a popular
260 city name instead—a behavior consistent with knowledge overshadowing (Zhang et al., 2024b).
261 ChatGPT does not exhibit this pattern, likely due to a stronger mastery of the relevant knowledge.

262 For confidence, question and relation popularity are strongly correlated in most cases, while answer
263 popularity has a weaker impact. Notably, question popularity consistently correlates strongly with
264 confidence and, in 7 of 9 cases, more than with QA performance. This suggests that LLMs may
265 become more confident simply because familiarity with the question, even if they do not know the
266 answer. On the Basketball dataset, confidence shows little correlation with answer popularity across
267 all three models. We hypothesize that the models are generally familiar with city names, and thus
268 do not exhibit higher confidence for samples with more common answers.

269 **2) LLMs better perceive their knowledge boundaries on more popular knowledge.** To better
270 understand this, we analyze how the gap between confidence and QA performance changes with

270 Table 4: Correlations between LLMs’ QA performance and Pop_Q , Pop_{Ge} , and RPop_{Ge} .
271

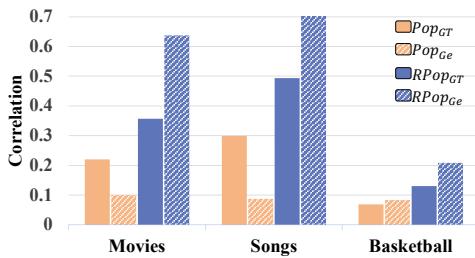
	Datasets	Models	Pop_Q	Pop_{Ge}	RPop_{Ge}
274 Movies		Llama3	0.317	0.100	0.637
		Qwen2	0.433	0.087	0.756
		ChatGPT	0.134	0.083	0.208
277 Songs		Llama3	0.277	0.257	0.621
		Qwen2	0.362	0.188	0.666
		ChatGPT	0.171	0.218	0.351
280 Basketball		Llama3	0.118	0.116	0.245
		Qwen2	0.014	0.116	0.106
		ChatGPT	0.288	-0.164	0.293

284 increasing knowledge popularity. Due to space constraints, we just present this gap for ChatGPT as
285 question popularity increases, shown in Figure 2. We observe that although LLMs are consistently
286 overconfident, their QA performance improves more rapidly than confidence as question popularity
287 increases, thereby narrowing the gap. Results for other models, as well as analyses based on other
288 popularity, are included in the Appendix and exhibit similar trends. As shown in Table 3, among
289 the three types of popularity, relation popularity typically shows the strongest correlation.
290

291 5 ANALYSIS OF MODEL-GENERATED ANSWERS

294 In real-world scenarios, ground-truth entities are often unavailable. This motivates us to investigate
295 whether the popularity of model-generated entities—along with their relational popularity with the
296 question entity—correlates with the model’s QA performance, confidence, and perception level. We
297 focus particularly on the relationship between popularity and QA performance, as a strong correlation
298 could enable us to leverage these signals for confidence calibration. The experimental settings
299 are the same as those in Section § 4.
300

301 5.1 RESULTS AND ANALYSIS



314 Figure 3: Comparison of the correlation be-
315 tween ChatGPT’s QA performance and ground-
316 truth vs. generated answers: Pop_{GT} vs. Pop_{Ge} ,
317 and RPop_{GT} vs. RPop_{Ge} .
318 based on ground-truth entities, as discussed in Section § 4.

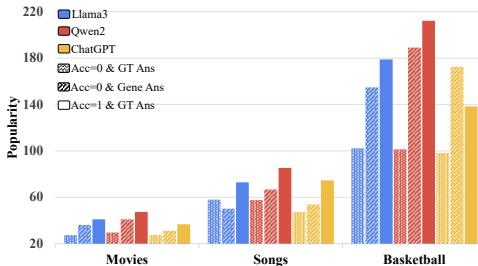
319 Pop_{Ge} shows a weaker correlation with QA performance compared to Pop_{GT} while RPop_{Ge} exhibits
320 a comparable or even stronger correlation than RPop_{GT} . We present the comparison for ChatGPT
321 in Figure 3, while results for other models can be obtained by comparing Table 3 and Table 4. To
322 better understand this, we perform a more detailed comparison between model-generated answers
323 and ground-truth answers. We only focus on cases where the model makes mistakes since the
generated answer matches the ground-truth answer otherwise and analyze in Section § 5.2.

303 Table 4 shows the Spearman correlation coefficients between LLMs’ QA performance and
304 knowledge popularity based on model-generated
305 entities. Due to space constraints, results on
306 LLMs’ confidence and perception levels are pro-
307 vided in Table 7 in the Appendix. We observe the
308 following.

309 The popularity of generated entities (Pop_{Ge})
310 and their co-occurrence with question entities
311 (RPop_{Ge}) positively correlate with LLMs’ QA
312 performance, confidence, and perception level in
313 most cases. RPop_{Ge} typically shows the strongest
314 correlation, outperforming both Pop_{Q} and Pop_{Ge} .
315 In contrast, Pop_{Ge} often exhibits the weakest cor-
316 relation. These findings are similar to the results
317

324 5.2 WHAT DO LLMs GENERATE WHEN THEY HALLUCINATE?
325

326 We focus on the differences in popularity between model-generated answers and ground-truth an-
327 swers when the model makes errors (See Figure 4), as well as the differences in their co-occurrence
328 frequency with the question entity (See Figure 5).
329



340 Figure 4: Pop_{GT} , Pop_{Ge} in incorrectly answered
341 samples and Pop_{GT} (also Pop_{Ge}) in correctly an-
342 swered samples.
343

344 In incorrectly answered samples, model-generated answers typically co-occur less frequently with
345 question entities compared to ground-truth answers, as shown in Figure 5. As a result, the correlation
346 between RPop_{Ge} and QA performance is stronger than that of RPop_{GT} , because the difference in
347 RPop_{Ge} between correct and incorrect samples is greater than that of RPop_{GT} .
348

349 6 CONFIDENCE CALIBRATION WITH KNOWLEDGE POPULARITY

350 Given that Pop_{Q} , Pop_{Ge} , and RPop_{Ge} —especially
351 RPop_{Ge} —are strongly correlated with QA per-
352 formance, we propose to use these signals for
353 confidence calibration, i.e., improving the effec-
354 tiveness of confidence in predicting answer cor-
355 rectness. Since obtaining these signals typically
356 requires external corpora and incurs additional
357 costs, we also explore whether LLMs can assess
358 knowledge familiarity on their own.
359

360 6.1 KNOWLEDGE
361 POPULARITY ACQUISITION

362 **Corpora-based Popularity.** As outlined in
363 Section § 3, we get popularity from external corpora.
364

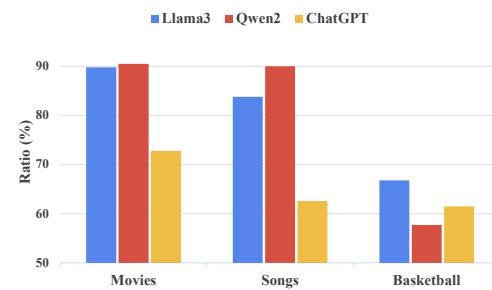
365 **Model-generated Popularity.** To eliminate reliance on external corpora and reduce the overhead
366 of collecting popularity, we investigate whether LLMs can self-assess their familiarity with a given
367 the entity or the relation. Familiarity is measured on a 10-point scale, where 1 denotes the lowest and
368 10 the highest level. The model is asked to provide its familiarity score accordingly. We provide the
369 model with varying numbers of corpora-based popularity examples to examine whether supplying
370 such examples helps the model produce more accurate familiarity. We present examples under both
371 zero-shot and few-shot settings and all these prompts can be found in Section § F in the Appendix.
372

373 6.2 BASELINES

374 We use representative confidence estimation methods that do not require access to model parameters
375 as our baselines.
376

- 377 • **Verbalized Confidence (Verb)** (Yin et al., 2023) instructs the model to verbally assess whether it
can answer the question correctly. The prompt can be seen in Figure 21 in the appendix.

378 **LLMs tend to generate entities that are more
379 common and less frequently co-occur with the
380 question entities than ground-truth answers
381 when they hallucinate.** As shown in Figure 4,
382 for incorrectly answered samples, the generated
383 entities are often more popular than the ground-
384 truth entities. So the popularity gap between cor-
385 rect and incorrect answers is smaller for gener-
386 ated entities than for ground-truth entities, lead-
387 ing to a weaker correlation between Pop_{Ge} and
388 QA performance. This may be because the
389 model tends to overgeneralize—once it learns
390 high-frequency answers, it tends to use them in
391 many similar contexts.
392



393 Figure 5: Proportion of incorrectly answered
394 samples where RPop_{Ge} is less than RPop_{GT} .
395

- **Self-consistency (Cosis)** (Manakul et al., 2023) estimates the model’s confidence by measuring the semantic consistency of multiple sampled answers. The core idea is that if the model knows the correct answer, multiple sampled answers should be semantically consistent. For each question, we sample 10 additional responses with a temperature of 1.0. For ChatGPT, due to cost constraints, we limit sampling to 3 additional responses. Semantic consistency is assessed using Qwen2.5-32B-Instruct (Yang et al., 2024).
- **Probabilistic Confidence (PC)** (Kumar et al., 2024) takes generation probabilities for the tokens in the answer as the model’s confidence. This is the confidence we have been using throughout the paper, and the one we aim to calibrate. Details can be seen in Section § 3.

6.3 DATA CONSTRUCTION

For each sample in the datasets, we construct a data pair $\{x, y\}$, where y is a binary correctness label: $y = 1$ if the generated answer contains the ground-truth answer, and $y = 0$ otherwise. To study the effect of each type of popularity on confidence calibration, we construct x using the following features: 1) **PC+Pop_Q**, 2) **PC + Pop_{Ge}**, 3) **PC + RPop_{Ge}**, and 4) **PC + ALL** — PC and all these three types of popularity. We also use each type of popularity itself as x to study its effectiveness in answer correctness prediction.

6.4 ANSWER CORRECTNESS PREDICTION

Based on the constructed x , we predict a binary judgment on correctness, as shown below.

$$\hat{y} = \mathcal{E}(x), \quad (2)$$

where \mathcal{E} represents the binary classification function and \hat{y} means the predicted correctness (i.e., 1 for correct and 0 for incorrect).

Single-feature Prediction. For the setting where x contains only a single feature, we select a threshold λ that maximizes prediction accuracy on the training set, and apply this threshold to perform binary classification on the test data. This can be formulated as:

$$\hat{y} = \begin{cases} 1 & \text{if } x > \lambda, \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

Multi-feature Prediction. For the setting where x contains multiple features, we perform binary classification using a lightweight MLP network, as defined below:

$$P(\hat{y} = 1) = \sigma(\text{MLP}(x)), \quad (4)$$

where σ refers to the softmax function, $x \in \mathcal{R}^{d \times h}$ represents the input features, d is the count of input features (e.g., $d=2$ for PC+Pop_Q) and h means the model’s hidden dimension. We use a 3-layer MLP with 64, 32, and 2 neurons in each layer, respectively. The activation function in MLP is ReLU. We employ cross-entropy loss as the training objective:

$$\mathcal{L}_{\text{CE}} = - \sum_{i=1}^N y_i \log(P_i) + (1 - y_i) \log(1 - P_i), \quad (5)$$

where y_i is the ground-truth correctness for the i -th training sample, N is the count of training samples, and P_i denotes $P(\hat{y}_i = 1)$. Detailed training parameters can be found in Section § D.

Metrics. We use answer correctness prediction accuracy as the metric. To reduce the impact of randomness, all our reported results are the averages obtained from three random seeds: (0, 42, 100).

Datasets and LLMs. We use the same data and LLMs as in Section § 4.1. We randomly split each dataset into two equal parts for training and testing and select the checkpoint with the highest prediction accuracy on the training set. Detailed settings can be found in Section § D in the Appendix.

432 Table 5: Accuracy of answer correctness prediction. Bold denotes the highest score in either
 433 corpora-based or self-generated knowledge popularity. Self-generated knowledge popularity is ob-
 434 tained under the zero-shot setting.

Features	Movies			Songs			Basketball			Avg.
	Llama3	Qwen2	ChatGPT	Llama3	Qwen2	ChatGPT	Llama3	Qwen2	ChatGPT	
<i>Baselines</i>										
Verb	65.58	45.93	83.41	40.22	29.58	69.25	51.58	50.49	48.89	53.88
Consis	82.21	74.61	96.00	77.62	86.31	83.72	53.76	52.10	77.77	76.01
PC	83.20	79.77	95.95	75.20	83.02	79.11	65.49	66.36	77.87	78.44
<i>Corpora-based Knowledge Popularity</i>										
Pop _Q	71.68	70.62	88.24	66.35	76.84	68.00	56.25	50.90	69.32	68.69
Pop _{Ge}	73.09	58.86	94.22	63.38	74.57	74.41	60.54	60.33	64.86	69.36
RPop _{Ge}	89.66	87.92	96.03	82.71	89.59	81.46	67.03	59.64	64.74	79.86
PC+Pop _Q	83.57	81.36	95.97	76.60	84.58	79.11	65.93	66.95	78.39	79.16
PC+Pop _{Ge}	84.46	80.49	95.58	76.68	83.57	80.12	69.04	68.64	78.62	79.69
PC+RPop _{Ge}	90.93	88.58	96.13	80.21	90.46	84.04	71.93	66.33	78.10	82.97
PC+ALL	93.32	92.47	96.37	81.46	88.11	82.71	71.93	68.18	78.59	83.68
<i>Self-generated Knowledge Popularity</i>										
PC+Pop _Q	83.91	80.60	95.87	77.85	84.82	79.50	65.30	67.31	78.43	79.29
PC+Pop _{Ge}	84.02	80.24	95.59	75.20	83.02	78.56	68.40	67.49	78.21	78.97
PC+RPop _{Ge}	85.30	80.20	95.80	79.65	84.04	79.81	66.17	67.90	77.59	79.61
PC+ALL	85.95	81.40	95.84	78.87	86.07	80.05	67.69	68.08	78.70	80.29

452 453 454 6.5 RESULTS AND ANALYSIS

455 **Results on corpora-based knowledge popularity.** Results based on knowledge popularity from
 456 external corpora is shown in the upper half of Table 5. We observe that: 1) Compared to the model’s
 457 confidence, RPop_{Ge} more accurately reflects answer correctness, outperforming all baselines in 6
 458 out of 9 cases. In contrast, Pop_Q and Pop_{Ge} individually show limited effectiveness in predicting
 459 correctness. 2) **All three types of popularity contribute to calibrating the model’s confidence,**
 460 **with their combination yielding the most effective results.** In most cases, augmenting PC with
 461 each type of popularity improves upon PC, with PC+RPop_{Ge} achieving the highest average accuracy
 462 among them. Notably, combining all three types leads to the most effective calibration, consistently
 463 outperforming PC and yielding an average accuracy improvement of 5.24% across diverse datasets
 464 and models. Further analysis and case studies are provided in Section § E.

465 **Results on model-generated knowledge popularity.** The prediction accuracy based on model
 466 self-generated knowledge popularity under the zero-shot setting can be found in the lower half of
 467 Table 5. It show that: 1) All three types of self-generated popularity contribute to confidence cali-
 468 bration. On average, all three signals can calibrate PC, and their combination achieves the best cali-
 469 bration effect, obtaining the optimal value in 6 out of 9 cases. However, the model’s self-generated
 470 signals yield weaker calibration effects compared to corpus-based knowledge popularity. The choice
 471 between corpus-based popularity and self-generated popularity depends on the trade-off between
 472 effectiveness and efficiency. 2) LLMs can not estimate popularity better with few-shot learning
 473 compared to zero-shot. Detailed analysis can be found Section § C in the Appendix.

474 475 476 7 CONCLUSION

477 In this paper, we investigate how knowledge popularity—measured through entity and relation
 478 popularity—affects LLMs’ QA performance, confidence, and perception of their knowledge
 479 boundaries, and explore its utility for confidence calibration. We find that LLMs perform better,
 480 express higher confidence, and demonstrate more accurate perception on more popular knowledge,
 481 with relation popularity having the strongest influence. We further show that the popularity and
 482 co-occurrence of model-generated answers also positively correlate with QA accuracy. Leveraging
 483 these popularity signals for confidence calibration yields an average 5.24% improvement in
 484 predicting answer correctness. To reduce reliance on external corpora, we also demonstrate that
 485 model-estimated popularity can serve as a viable alternative, offering a practical trade-off between
 performance and efficiency.

486
487
ETHICS STATEMENT488
489
490
491
492
493
494
495
We approach ethics with great care. In this paper, all the datasets and models we use are open-source.
490
491
492
493
494
495
496
497
498
499
500
Our analysis of knowledge popularity does not introduce any harmful information. Moreover, our
proposed method can help accurately determine whether the model’s answer is trustworthy, prevent-
ing users from being misled by incorrect responses.501
502
REPRODUCIBILITY STATEMENT503
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
All datasets used in this paper are publicly available, and the popularity signals were constructed
based on these datasets. The detailed procedure is described in Section 4.1. The three LLMs
employed in this study are widely adopted models. Furthermore, for training, we only used a
lightweight MLP network, which requires minimal computational resources. Furthermore, all
prompts used in this paper are provided in Section ??.501
502
REFERENCES503
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
Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Ale-
man, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical
report. *arXiv preprint arXiv:2303.08774*, 2023.
Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping
Ye. Inside: Llms’ internal states retain the power of hallucination detection. *arXiv preprint*
arXiv:2402.03744, 2024.
Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. *arXiv preprint*
arXiv:2003.07892, 2020.
Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Retrieve only when it
needs: Adaptive retrieval augmentation for hallucination mitigation in large language models.
arXiv preprint arXiv:2402.10612, 2024.
Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.
arXiv preprint arXiv:2407.21783, 2024.
Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural
networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.
Jan Hauke and Tomasz Kossowski. Comparison of values of pearson’s and spearman’s correlation
coefficients on the same sets of data. *Quaestiones geographicae*, 30(2):87–93, 2011.
Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recog-
nition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.
770–778, 2016.
Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language
models know? on the calibration of language models for question answering. *Transactions of the*
Association for Computational Linguistics, 9:962–977, 2021.
Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez,
Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language mod-
els (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.
Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for
uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.
Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. Confidence under
the hood: An investigation into the confidence-probability alignment in large language models.
arXiv preprint arXiv:2405.16282, 2024.

540 Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in
 541 words. *arXiv preprint arXiv:2205.14334*, 2022.

542

543 Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi.
 544 When not to trust language models: Investigating effectiveness of parametric and non-parametric
 545 memories. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of
 546 the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long
 547 Papers)*, pp. 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguis-
 548 tics. doi: 10.18653/v1/2023.acl-long.546. URL <https://aclanthology.org/2023.acl-long.546/>.

549

550 Potsawee Manakul, Adian Liusie, and Mark JF Gales. Selfcheckgpt: Zero-resource black-box hallu-
 551 cination detection for generative large language models. *arXiv preprint arXiv:2303.08896*, 2023.

552

553 Shiyu Ni, Keping Bi, Jiafeng Guo, and Xueqi Cheng. When do llms need retrieval augmentation?
 554 mitigating llms’ overconfidence helps retrieval augmentation. *arXiv preprint arXiv:2402.11457*,
 555 2024a.

556

557 Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. Are large language models more honest in their
 558 probabilistic or verbalized confidence? *arXiv preprint arXiv:2408.09773*, 2024b.

559

560 Shiyu Ni, Keping Bi, Jiafeng Guo, Lulu Yu, Baolong Bi, and Xueqi Cheng. Towards fully exploiting
 561 llm internal states to enhance knowledge boundary perception. *arXiv preprint arXiv:2502.11677*,
 562 2025.

563

564 Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi
 565 Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text
 566 transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

567

568 Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Boyd-Graber, and
 569 Lijuan Wang. Prompting gpt-3 to be reliable. *arXiv preprint arXiv:2210.09150*, 2022.

570

571 Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. Un-
 572 supervised real-time hallucination detection based on the internal states of large language models.
 573 *arXiv preprint arXiv:2403.06448*, 2024.

574

575 Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea
 576 Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated
 577 confidence scores from language models fine-tuned with human feedback. *arXiv preprint
 578 arXiv:2305.14975*, 2023.

579

580 Yanling Wang, Haoyang Li, Hao Zou, Jing Zhang, Xinlei He, Qi Li, and Ke Xu. Hidden question
 581 representations tell non-factualty within and across large language models. *arXiv e-prints*, pp.
 582 arXiv–2406, 2024.

583

584 Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms
 585 express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint
 586 arXiv:2306.13063*, 2023.

587

588 An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li,
 589 Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint
 590 arXiv:2407.10671*, 2024.

591

592 Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. Alignment for honesty.
 593 *arXiv preprint arXiv:2312.07000*, 2023.

594

595 Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. Do large
 596 language models know what they don’t know? *arXiv preprint arXiv:2305.18153*, 2023.

597

598 Mert Yuksekgonul, Varun Chandrasekaran, Erik Jones, Suriya Gunasekar, Ranjita Naik, Hamid
 599 Palangi, Ece Kamar, and Besmira Nushi. Attention satisfies: A constraint-satisfaction lens on
 600 factual errors of language models. *arXiv preprint arXiv:2309.15098*, 2023.

594 Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji,
595 and Tong Zhang. R-tuning: Instructing large language models to say ‘i don’t know’. In *Proceed-
596 ings of the 2024 Conference of the North American Chapter of the Association for Computation-
597 al Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7106–7132, 2024a.
598

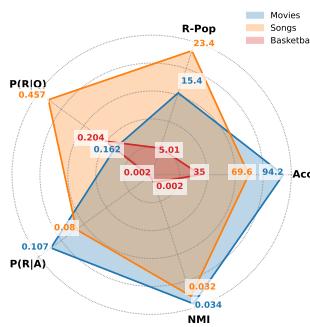
599 Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley A Malin, and Srirangan Kumar. Sac3: Reliable
600 hallucination detection in black-box language models via semantic-aware cross-check consis-
601 tency. *arXiv preprint arXiv:2311.01740*, 2023.

602 Yuji Zhang, Sha Li, Jiateng Liu, Pengfei Yu, Yi R Fung, Jing Li, Manling Li, and Heng Ji. Knowl-
603 edge overshadowing causes amalgamated hallucination in large language models. *arXiv preprint
604 arXiv:2407.08039*, 2024b.

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 **A DATA FILTERING**
649

650 The model’s generated response may be empty or fail to find a corresponding entity in Wikidata. To
 651 ensure comparability of results across different models on the same dataset, we filter out data where
 652 any model’s generation is empty or the generated entity cannot be found in Wikidata. Additionally,
 653 for the Movies and Songs datasets, we filter out cases where the question entity, ground truth entity,
 654 or model-generated entity appears in more than 6,000 documents. This is because entities in these
 655 two datasets typically do not appear in more than 6,000 documents, and those that exceed this
 656 threshold often introduce noise. For example, ”Queen” appears more than 6,000 times but is not
 657 exclusively used as a band name. We filter these cases to obtain an accurate co-occurrence counts.
 658 After filtering, the remaining data sizes for the Movies, Songs, and Basketball datasets are 8,184,
 659 852, and 13,136, respectively.
 660



672 Figure 6: QA performance and NMI calculated based on ChatGPT. R-Pop means relation popularity,
 673 where $P(R|Q)$ and $P(R|A)$ denote the co-occurrence proportion of question and answer entities
 674 relative to their individual occurrences in documents.
 675

676 **B ANALYSIS ON RELATIONSHIP STRENGTH**
677

678 We hypothesize that the strength of the relationship between entities may also influence the model’s
 679 learning. Specifically, when the subject and object frequently co-occur but are also commonly as-
 680 sociated with other entities, the model may struggle to learn their specific relationship. We use
 681 normalized mutual information to quantify relationship strength and find that stronger relationship
 682 contributes to better QA performance. Normalized mutual information is computed as:
 683

$$684 \text{NMI}(X, Y) = \frac{I(X; Y)}{\sqrt{H(X)H(Y)}}, \quad (6)$$

686 where $I(X, Y)$ is defined as:
 687

$$688 I(X, Y) = \sum_{i=1}^n \sum_{j=1}^m P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)}, \quad (7)$$

691 and $H(X)$ and $H(Y)$ serve as regularization terms to mitigate the influence of the sizes of n and m ,
 692 as well as the magnitude of probability values. Their formulations are:
 693

$$694 H(X) = - \sum_{i=1}^n P(x_i) \log P(x_i), \quad (8)$$

$$697 H(Y) = - \sum_{j=1}^m P(y_j) \log P(y_j). \quad (9)$$

700 Specifically, for a dataset converted from knowledge triplets $D = \{s_i, r, o_i\}_{i=1}^n$, we define $X =$
 701 $\{s_1, \dots, s_n\}$ and $Y = \{o_1, \dots, o_n\}$. We estimate $P(s_i)$, $P(o_j)$, and $P(s_i, o_j)$ using a Wikipedia
 702 dump of d documents, where $P(s_i)$ and $P(o_j)$ are the proportions of documents containing s_i and

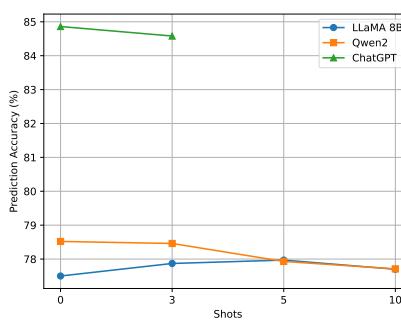


Figure 7: The prediction accuracy obtained by performing confidence calibration using knowledge popularity generated from different numbers of examples. Each point represents the average prediction accuracy of the model across three datasets.

o_j , respectively, and $P(s_i, o_j)$ is the proportion containing both. If $i \neq j$, we set $P(s_i, o_j) = 0$, as we focus only on the relationship between s_i and o_i .

The results based on ChatGPT are shown in Figure 6. We observe that, compared to the movies dataset, the question entity co-occurs more frequently with the ground-truth entity in the songs dataset, yet the model’s QA performance is lower. This can be attributed to the lower NMI in the songs dataset, driven by a low $P(R|A)$. This indicates that, besides the question entity, the answer entity also co-occurs with many other entities through various relations. This may interfere with the model’s memory of the relationship between the question entity and the answer entity.

C EFFECTS OF FEW-SHOT LEARNING ON POPULARITY GENERATION

Examples selection. For a given dataset, we sort all samples by popularity in ascending order, remove duplicates, and divide the popularity values into 10 equal intervals, assigning values from 1 to 10 in ascending order. Each sample is then assigned to its corresponding interval, updating its popularity accordingly. For 3-shot, we randomly select one sample from the intervals with popularity values of 2, 5, and 8. For 5-shot, we randomly select one sample from the intervals with popularity values of 1, 3, 5, 7, and 9. For 10-shot, we randomly select one sample from each of the 10 intervals.

Results. Figure 7 shows the average prediction accuracy of Conf-QG-R across three datasets using model-generated popularity under different shot settings. As the number of examples increases, prediction accuracy does not improve, while inference cost rises. Therefore, we recommend prompting LLMs to assess their familiarity with entities and their relationships in a zero-shot setting. Due to API costs, we first conduct experiments on LLaMA3-8B-Instruct and Qwen2-7B-Instruct and find that increasing the number of samples in the prompt does not yield more effective knowledge popularity. Therefore, we only perform 0-shot and 3-shot experiments on ChatGPT.

D DETAILED PARAMETER SETTINGS

Inference. For all the models, we use greedy search, selecting the token with the highest probability at each generation step. For open-source models, our experiments are conducted on a single 80GB A800 GPU.

MLP Training. For both corpora-based and model-generated popularity, we train the model using the Adam optimizer with a learning rate of 2e-3 and a batch size of 8. The intermediate layer has a dropout rate of 0.4, and training runs for 100 epochs. All experiments are conducted on two 16GB V100 GPUs. We select the checkpoint with the highest prediction accuracy on the training set for evaluation on the test set.

756 **Class Balancing for The Basketball Dataset.** Since the MLP fails to learn meaningful patterns on
 757 the basketball dataset for Llama3 and Qwen2—consistently classifying all samples as incorrect due
 758 to the overwhelming imbalance—we extract all correctly answered samples and randomly sample
 759 an equal number of incorrect ones (seed = 0) to ensure balanced learning across both classes. The
 760 training set and the test set are evenly split from the sampled dataset.

763 E CASE STUDIES

765 We compare PC and PC+ALL on LLaMA3 for answer correctness prediction to illustrate how
 766 knowledge popularity works in confidence calibration. The imperfect alignment between the
 767 model’s confidence and its actual performance arises from two main factors:

- 769 • **Overconfidence:** The model generates incorrect answers with high confidence. When classifica-
 770 tion relies on generation probabilities, such answers are incorrectly labeled as correct.
- 771 • **Conservativeness:** The model generates correct answers with low confidence. When classifica-
 772 tion relies on generation probabilities, such answers are incorrectly labeled as incorrect.

773 We collect the samples misclassified by PC but successfully calibrated by PC+ALL. These fall into
 774 two categories:

- 775 • **Overconfidence Group:** Samples where the model generates an incorrect answer, PC incorrectly
 776 classifies them as correct, while PC+ALL correctly identifies them as incorrect.
- 777 • **Conservativeness Group:** Samples where the model generates a correct answer, PC incorrectly
 778 classifies them as incorrect, while PC+ALL correctly identifies them as correct.

779 We compute the knowledge popularity for each group, and the results appear in Table 6. The results
 780 show that in the overconfidence group, PC+ALL achieves calibration by leveraging low knowl-
 781 edge popularity despite the model’s high confidence. In contrast, in the conservativeness group, it
 782 achieves calibration through high knowledge popularity.

783 Although PC+ALL achieves strong calibration performance, it also introduces some over-calibration
 784 issues by misclassifying samples that were correctly predicted by PC, as shown in Figure 8. How-
 785 ever, the number of correctly calibrated samples significantly exceeds the over-calibrated ones.
 786 Moreover, we show some cases on the Movies dataset for Llama3. Figures 9 and Figure 10 illustrate
 787 cases where knowledge popularity effectively calibrated the model’s confidence, while Figure 11
 788 shows a failure case. All the results in this section are obtained with seed=0.

Datasets	Group	PC	Pop_Q	Pop_{Ge}	$RPop_{Ge}$
Movies	Overc.	0.91	47.05	20.08	1.03
	Conse.	0.78	24.34	23.22	12.91
Songs	Overc.	0.91	38.91	13.04	18.89
	Conse.	0.78	131.00	15.50	103.00
Basketball	Overc.	0.78	102.68	10.69	0.97
	Conse.	0.62	234.83	10.59	11.47

797 Table 6: Knowledge popularity of samples that are misclassified by PC but correctly classified by
 798 PC+ALL. Overc. refers to the Overconfidence group, in which the model generates an incorrect
 799 answer but PC classifies it as correct. Conse. refers to the Conservativeness group, in which the
 800 model generates a correct answer but PC classifies it as incorrect.

802 From Figure 10, we can see that the model generated an incorrect answer with a probabilistic confi-
 803 dence of 0.95, which is significantly higher than the classification threshold for confidence ($\zeta 0.85$),
 804 leading to it being classified as correct. However, knowledge popularity reveals that the question
 805 pop, generated answer pop, and relation pop are 16, 20, and 1, respectively, all below the dataset’s
 806 average levels. This indicates that both the question and the generated entity are relatively uncom-
 807 mon and rarely co-occur. As a result, the classification outcome was corrected to incorrect. Simi-
 808 larly, in Figure 10, the model exhibits low probabilistic confidence for a correctly generated answer,
 809 leading to a misclassification as incorrect. However, its knowledge popularity was relatively high,
 resulting in a correction to the correct classification.

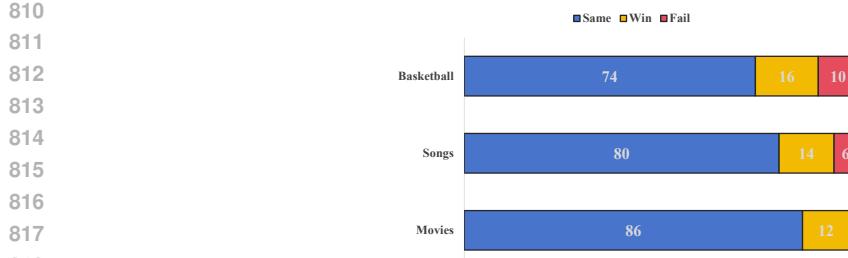


Figure 8: The difference in answer correctness prediction on LLaMA3 between using PC+ALL and using PC. Blue indicates that both methods make the same prediction, yellow indicates cases where only PC+ALL predicts correctly, and red indicates cases where only PC predicts correctly.

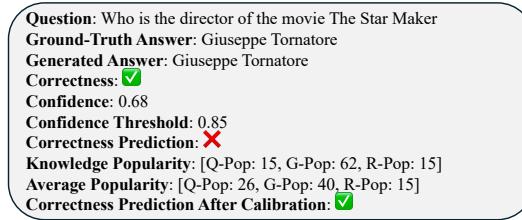


Figure 9: The case where Llama3-8B generates incorrect answers with high confidence on the Movieis dataset and is corrected by low knowledge popularity.

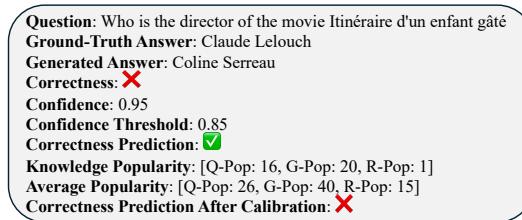


Figure 10: The case where Llama3-8B generates correct answers with low confidence on the Movieis dataset and is corrected by high knowledge popularity.

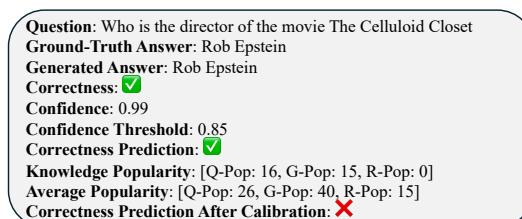


Figure 11: The case where Llama3-8B generates correct answers with high confidence on the Movieis dataset and is misled by low knowledge popularity.

Figure 11 presents a case of error correction. While similar misclassifications may occur, the proportion of correctly corrected samples (6.0%) is significantly higher than that of miscalibrated ones (1.2%), demonstrating the reliability of knowledge popularity in confidence calibration.

F PROMPTS

We display all the prompts used in this paper here and show some examples.

QA prompt. We just ask the model to give a short answer without any other words. The example is shown in Figure 20.

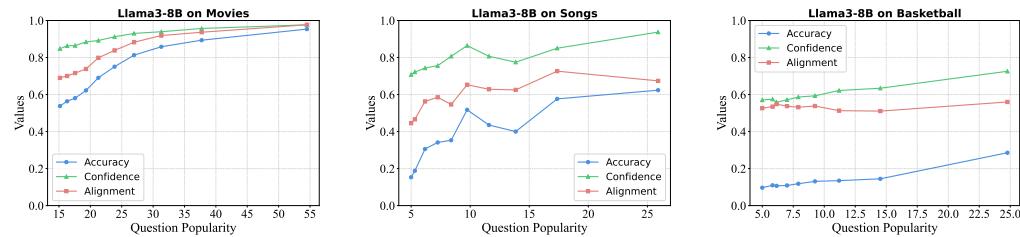
864
 865 **Prompts for knowledge popularity generation.** Examples for instructing LLMs to provide ques-
 866 tion entity popularity, generated answer popularity, and the popularity of their relationship can be
 867 found in Figure 22 23 24 25 26 27.

868 G THE USE OF LARGE LANGUAGE MODELS 869

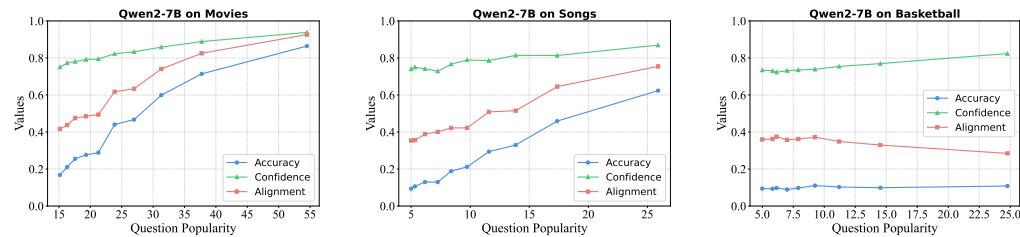
870 LLMs were used solely for grammar correction and sentence polishing. All content and experi-
 871 ments in this paper were conducted entirely by humans, and any model-polished text was manually
 872 reviewed.
 873

874 Datasets	875 Models	876 Accuracy			877 Confidence			878 Alignment		
		879 Q-Pop	880 G-Pop	881 Co-Occ	882 Q-Pop	883 G-Pop	884 Co-Occ	885 Q-Pop	886 G-Pop	887 Co-Occ
888 Movies	Llama3-8B	0.317	0.100	0.637	0.404	0.324	0.653	0.404	0.231	0.667
	Qwen2-7B	0.433	0.087	0.756	0.413	0.345	0.679	0.386	0.021	0.607
	ChatGPT	0.134	0.083	0.208	0.210	0.233	0.304	0.211	0.231	0.304
889 Songs	Llama3-8B	0.277	0.257	0.621	0.369	0.188	0.680	0.182	0.207	0.358
	Qwen2-7B	0.362	0.188	0.666	0.300	0.246	0.511	0.230	0.058	0.405
	ChatGPT	0.171	0.218	0.351	0.249	0.305	0.445	0.232	0.297	0.326
890 Basketball	Llama3-8B	0.118	0.116	0.245	0.173	-0.034	0.010	-0.052	0.083	0.163
	Qwen2-7B	0.014	0.116	0.106	0.151	0.114	0.068	-0.126	-0.015	0.018
	ChatGPT	0.288	-0.164	0.293	0.351	-0.210	0.257	0.201	-0.107	0.241

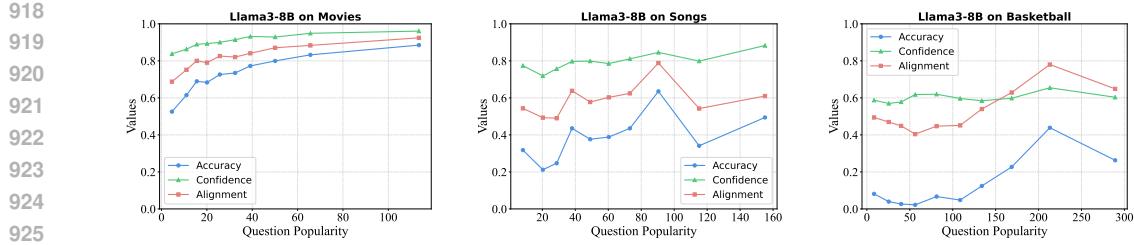
888 Table 7: Spearman correlation coefficients for Accuracy, Confidence, and Alignment scores with
 889 the popularity of question entities, generated entities, and their co-occurrence.



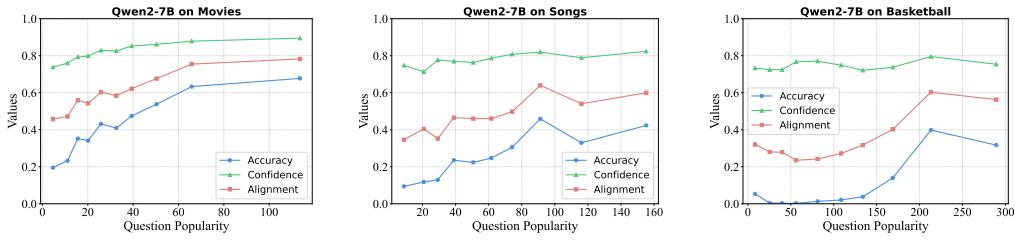
890 Figure 12: The QA performance, confidence, and alignment of Llama3 under different question
 891 popularity.



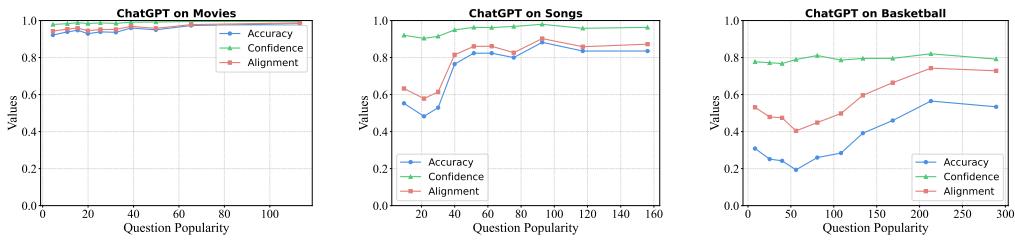
892 Figure 13: The QA performance, confidence, and alignment of Qwen2 under different question
 893 popularity.



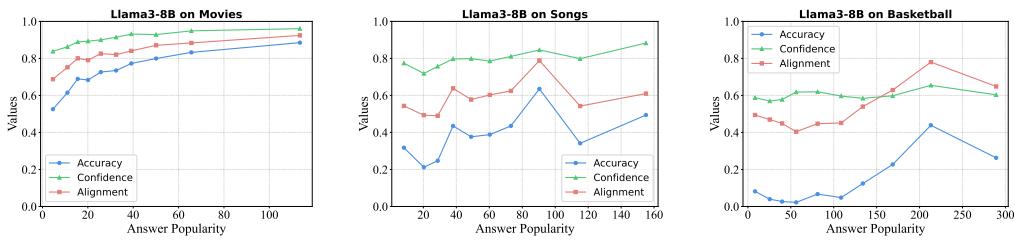
926
927
928
Figure 14: The QA performance, confidence, and alignment of Llama3 under different answer popularity.



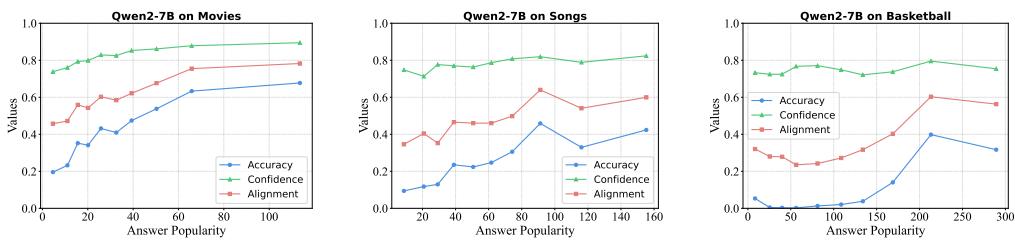
937
938
939
Figure 15: The QA performance, confidence, and alignment of Qwen2 under different answer popularity.



948
949
950
Figure 16: The QA performance, confidence, and alignment of ChatGPT under different answer popularity.



959
960
961
Figure 17: The QA performance, confidence, and alignment of Llama3 under different relation popularity.



970
971
Figure 18: The QA performance, confidence, and alignment of Qwen2 under different relation popularity.

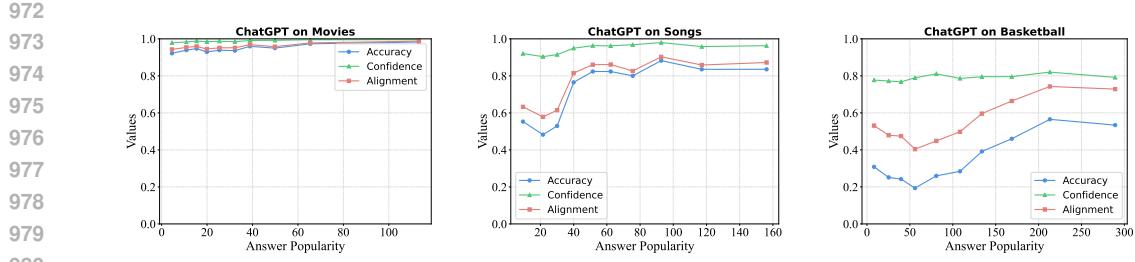


Figure 19: The QA performance, confidence, and alignment of ChatGPT under different relation popularity.

Input:
 Answer the following question with one or few words.
 Question: Who is the director of the movie The Intouchables
 Answer:
Response: Eric Toledano

Figure 20: A question-answering example for Llama3.

Input:
 Judge whether the following answer (this is your self-generated answer) about the question is correct. If you are sure the answer is correct, say certain. If not, please say uncertain. Just give your judgement without any other words.
 Question: Where is the birthplace of the basketball player Jiang Xingquan?
 Answer: Beijing.
Response: Uncertain.

Figure 21: An example for verbalized confidence.

Question: Rate how familiar you are with the **movie** 'The Intouchables'. The familiarity is rated on a scale from 1 to 10, where 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it. Your answer needs to be a precise integer. Provide only the number, without any additional explanation.
 Number:

Response: 8

Figure 22: An example of obtaining question popularity on the movies dataset using LLaMA3 in a zero-shot setting.

Question: Rate how familiar you are with the **director** 'Eric Toledano'. The familiarity is rated on a scale from 1 to 10, where 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it. Your answer needs to be a precise integer. Provide only the number, without any additional explanation.
 Number:

Response: 4

Figure 23: An example of obtaining generated answer popularity on the movies dataset using LLaMA3 in a zero-shot setting.

1026

1027

1028 **Question:** Rate how familiar you are with the **relationship** between the **movie** 'The Intouchables' and the
 1029 **director** 'Eric Toledano'. The familiarity is rated on a scale from 1 to 10, where 10 means you are highly familiar
 1030 with their relationship, and 1 means you know little to nothing about it. Your answer needs to be a precise
 1031 integer. Provide only the number, without any additional explanation.

1032 Number:

1033 **Response:** 8

1034

1035 Figure 24: An example of obtaining relation popularity on the movies dataset using LLaMA3 in a
 1036 zero-shot setting.

1037

1038

1039

1040

1041 **Question:** Rate how familiar you are with the **movie** 'Swept Away'. The familiarity is rated on a scale from 1
 1042 to 10, where 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it.
 1043 Your answer needs to be a precise integer. Provide only the number, without any additional explanation.

1044 **Here are some examples:**

1045 The movie: Matchstick Men

1046 Number: 2

1047 The movie: Kick-Ass

1048 Number: 5

1049 The movie: Skyfall

1050 Number: 8

1051 Rate how familiar you are with the **movie** 'Swept Away'. The familiarity is rated on a scale from 1 to 10, where
 1052 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it. Your answer
 1053 needs to be a precise integer. Provide only the number, without any additional explanation.

1054 Number:

1055 **Response:** 3

1056

1057 Figure 25: An example of obtaining question popularity on the movies dataset using ChatGPT in a
 1058 3-shot setting.

1059

1060

1061

1062 **Question:** Rate how familiar you are with the **director** 'Guy Ritchie'. The familiarity is rated on a scale from 1
 1063 to 10, where 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it.
 1064 Your answer needs to be a precise integer. Provide only the number, without any additional explanation.

1065 **Here are some examples:**

1066 The director: James McTeigue

1067 Number: 2

1068 The director: Guy Ritchie

1069 Number: 5

1070 The director: Jodie Foster

1071 Number: 8

1072 Rate how familiar you are with the **director** 'Guy Ritchie'. The familiarity is rated on a scale from 1 to 10,
 1073 where 10 means you are highly familiar with it, and 1 means you have little to no knowledge about it. Your
 1074 answer needs to be a precise integer. Provide only the number, without any additional explanation.

1075 Number:

1076 **Response:** 7

1077

1078 Figure 26: An example of obtaining answer popularity on the movies dataset using ChatGPT in a
 1079 3-shot setting.

1080
 1081
 1082
 1083
 1084
 1085
 1086
 1087
 1088
 1089
 1090
 1091
 1092
 1093
 1094
 1095
 1096
 1097
 1098 **Question:** Rate how familiar you are with the relationship between the **movie** 'Swept Away' and the **director** 'Guy Ritchie'. The familiarity is rated on a scale from 1 to 10, where 10 means you are highly familiar with their relationship, and 1 means you know little to nothing about it. Your answer needs to be a precise integer.
 1099 Provide only the number, without any additional explanation.
 1100
 1101 **Here are some examples:**
 1102 The movie: Kick-Ass; The director: Matthew Vaughn
 1103 Number: 2
 1104 The movie: Eraserhead; The director: David Lynch
 1105 Number: 5
 1106 The movie: Heat; The director: Michael Mann
 1107 Number: 8
 1108 Rate how familiar you are with the relationship between the **movie** 'Swept Away' and the **director** 'Guy
 1109 Ritchie'. The familiarity is rated on a scale from 1 to 10, where 10 means you are highly familiar with their
 1110 relationship, and 1 means you know little to nothing about it. Your answer needs to be a precise integer.
 1111 Provide only the number, without any additional explanation.
 1112 Number:
 1113 **Response:** 7
 1114
 1115 Figure 27: An example of obtaining relation popularity on the movies dataset using ChatGPT in a
 1116 3-shot setting.
 1117
 1118
 1119
 1120
 1121
 1122
 1123
 1124
 1125
 1126
 1127
 1128
 1129
 1130
 1131
 1132
 1133