Who Said What: Formalization and Benchmarks for the Task of Quote Attribution

Wenjie Zhong[1,3], **Jason Naradowsky**[1], **Hiroya Takamura**[3],
Ichiro Kobayashi[2,3], **Yusuke Miyao**[1,3]
[1]The University of Tokyo, [2]Ochanomizu University,
[3]National Institute of Advanced Industrial Science and Technology
{zvengin, narad, yusuke}@is.s.u-tokyo.ac.jp
takamura.hiroya@aist.go.jp, koba@is.ocha.ac.jp

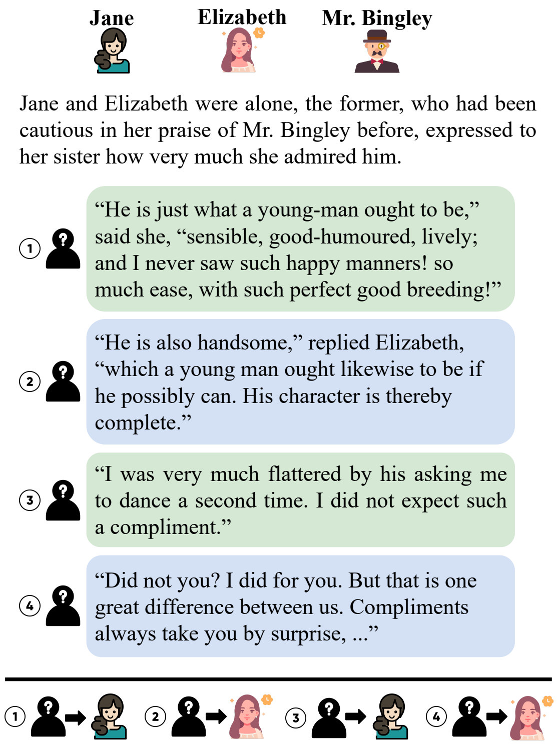
Abstract

The task of quote attribution seeks to pair textual utterances with the name of their speakers. Despite continuing research efforts on the task, models are rarely evaluated systematically against previous models in comparable settings on the same datasets. This has resulted in a poor understanding of the relative strengths and weaknesses of various approaches. In this work we formalize the task of quote attribution, and in doing so, establish a basis of comparison across existing models. We present an exhaustive benchmark of known models, including natural extensions to larger LLM base models, on all available datasets in both English and Chinese. Our benchmarking results reveal that the CEQA model attains state-of-the-art performance among all supervised methods, and ChatGPT, operating in a four-shot setting, demonstrates performance on par with or surpassing that of supervised methods on some datasets. Detailed error analysis identify several key factors contributing to prediction errors.

Keywords: Quote Attribution, Speaker Identification, Benchmark

1. Introduction

Dialogues play a central role in literary texts, contributing to character development and plot advancement (Cuesta-Lázaro et al., 2022). Accurate and automated analysis of dialogues can be valuable for downstream tasks, such as character relation analysis (Jayannavar et al., 2015; Labatut and Bost, 2019). However, before large-scale automated analysis of dialogues is possible, each quote must be labeled with its speaker, a task known as **quote attribution**. Previous approaches to quote attribution have utilized hand-crafted features to identify the speaker names (He et al., 2013; Muzny et al., 2017). With recent advances in leveraging representations from pre-trained language models, accuracy has significantly improved (Chen et al., 2021; Yu et al., 2022; Zhou et al., 2022).



Jane and Elizabeth were alone, the former, who had been cautious in her praise of Mr. Bingley before, expressed to her sister how very much she admired him.

“He is just what a young-man ought to be,” said she, “sensible, good-humoured, lively; and I never saw such happy manners! so much ease, with such perfect good breeding!”

“He is also handsome,” replied Elizabeth, “which a young man ought likewise to be if he possibly can. His character is thereby complete.”

“I was very much flattered by his asking me to dance a second time. I did not expect such a compliment.”

“Did not you? I did for you. But that is one great difference between us. Compliments always take you by surprise, ...”

Figure 1: An example of quote attribution task. The task involves assigning a speaker name to each quote in the dialogue.

Despite significant advancements in prior research, the task remains ill-defined, with no consensus on task settings. Task considerations include what constitutes a quote-speaker pair, the context around the quote, and whether character names are known in advanced. Furthermore, previous studies (Muzny et al., 2017; Chen et al., 2021; Yao et al., 2022) compare their methods with limited baselines, often on a single dataset, leading to a poor understanding of model limitations. The absence of a standardized benchmark for comparison poses a challenge in determining the state-of-the-art method, and what relative strengths or weaknesses pertain to each approach. Additionally, the inadequacy of comprehensive analysis of existing models may impede future research from making meaningful contributions.

In order to support future research endeavors, we propose a formalization of the task of quote

LREC-COLING 2024, pages 17588–17602
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

attribution. Under this definition, we rigorously assess all available models using publicly available datasets in two languages, Chinese and English, and establish an inclusive benchmark for future work. We identify and quantify several factors that contribute to prediction errors. These factors exist at both the quote level and book level. At the quote level, we explore the effect of speaker names, non-speaker character names, speech tags occurring in the near vicinity of quotes. At the book level, we explored the robustness of models across various literary genres.

Our contributions in this work are threefold:

1. We establish a formal definition of the quote attribution task and convert all available datasets to a standard format.

2. We adapt prior available models and benchmark their performance on multiple datasets. We release the benchmark code for future work[1]. For models using pre-trained language models, we also assess their performance across larger and more recent base models than were available in the original work. We also develop a system for quote attribution using ChatGPT, in both zero and few-shot settings. Results show that an existing model (CEQA) attains state-of-the-art performance, and a ChatGPT-based approach operating in a four-shot setting performs comparably, surpassing supervised methods on some datasets.

3. We offer a comprehensive examination of the factors impacting performance and illuminate potential avenues for future research. Among these factors are the manner in which the quote is phrased (*quote mention type*), the presence of confounding speakers in the vicinity of the quote mention, and the extent to which predictions adhere to typical discourse structure.

2. Related Work

There are a number of existing works related to the task of quote attribution beyond the scope of this study. In addition to quote attribution in the literary domain, quote attribution has also been studied in newswire (O'Keefe et al., 2012; Almeida et al., 2014; Salway et al., 2017; Zhang and Liu, 2022).

Additionally, other methods have been proposed for the task. These include other models which rely on the hand-crafted features (Zhang et al., 2003; Mamede and Chaleira, 2004; Glass and Bangay, 2007), and train machine learning models such as logistic regression (Elson and McKeown, 2010; O'Keefe et al., 2012), SVM ranking (He et al., 2013), linear model (Almeida et al., 2014), conditional random fields (Yeung and Lee, 2017), and averaged structured perceptron (Ek et al., 2018). As more recent LLM-based models have shown superior performance in text classification tasks, we refrain from including these methods in this survey.

While some works (Almeida et al., 2014; Ek et al., 2018) have touched upon error analysis in quote attribution, their examinations were limited in scope and tied to their respective methods. In contrast, our study offers a comprehensive and thorough investigation into the factors contributing to errors in publicly accessible datasets across various models.

3. Task Definition

We define the task of quote attribution as follows. A book consists of a sequence of paragraphs $\{p_j\}_{j=0}^{N}$[2]. Note that we adopt the established task definition of a paragraph as a distinct block of text, separated from the next using a newline or indent. This differs from a common colloquial definition where a paragraph may contain multiple alternating quotes, each separated from the next. We treat a dialogue as a longest possible subsequence of paragraphs $\{p_j, ..., p_k\}$, where each paragraph contains quotes, $\{q_j^m, ..., q_j^n\}$. Prior studies (Chen et al., 2021; Muzny et al., 2017) concatenate multiple paragraphs, but we argue this is an unreasonable for quote attribution since the paragraph boundaries serve as a marker of speaking turn alternation in literature text and such information is beneficial for speaker inference in a multi-turn dialogue (He et al., 2013).

The task of quote attribution then is to map each quote q_j^m to a label $l \in L$, where L is the set of character names and where l is the target speaker. There is not a consensus in previous work whether character name labels should be provided. However, because it is possible to automatically extract a list of character names from a dialogue using regular expressions (Cuesta-Lázaro et al., 2022), we assume L is provided. In practice there may be noise in the process, and we use l_{null} as the gold label when the speaker is not in the label set.

A second point of consideration is what additional context around each quote the model can observe when making predictions. There is some contention on whether the context should be defined in various windows of text around the quote (Chen et al., 2021; Muzny et al., 2017), or to regard the

[1] https://github.com/ZVengin/Speaker-Identification-Benchmark-COLING2024.git

[2] For the purpose of this work which focuses on narrative texts with clearly defined paragraphs, and where paragraphs may carry important narrative purpose, we adopt paragraphs as a unit of chunking sentences. Alternatively, a window of w sentences could be substituted.

	Chinese			English			
	CSI	WP	JY	P&P	QuoteLi	RIQUA	PDNC
Books	18	1	3	1	3	11	22
Quotes(K)	67	2	29	2	3	6	34
Distance	26	40	19	64	63	22	60
Per Dialogue							
Turns	2.0	1.8	1.3	3.6	3.5	1.6	4.0
Characters	19.8	4.0	3.8	9.1	9.2	10.5	6.6
Speakers	1.6	1.4	1.2	1.7	1.7	1.3	1.5

Table 1: The statistics for the datasets. The **Distance** is the number of tokens between the quote and the closest mention of speaker.

whole dialogue as the context. (He et al., 2013; Cuesta-Lázaro et al., 2022; Ek et al., 2018). To make the context more general, we define the context (in terms of paragraphs) as $\{p_{j-w}, \ldots, p_{k+w}\}$, for some window size w, covering the text inside and around the dialogues. This choice allows us to apply the task definition both to entire books and to datasets which provide only isolated quotes with a limited surrounding context.

4. Benchmark Models

In this work, we only consider publicly available models including the **T**wo-stage **S**ieve based approach(**TSQA**) (Muzny et al., 2017), the **N**eural-**N**etwork based approach(**NNQA**) (Chen et al., 2021), the **E**nd-to-**E**nd approach (**EEQA**) (Yu et al., 2022), the **C**haracter-**E**mbedding based approach (**CEQA**) (Yao et al., 2022), and **ChatGPT** (Ouyang et al., 2022). We summarize the major distinctions between models below:

TSQA is a quote attribution model that relies on a set of manually-crafted features to identify the speaker for each quote.

NNQA utilizes BERT (Devlin et al., 2019) to encode context, quote, the candidate, which are scored by a MLP layer to generate a score for each candidate speaker name, and returns the highest scoring candidate as the speaker label.

EEQA formulates the quote attribution task as a question-answering problem, and the context is treated as the document and the quote is as the question. BERT (Cui et al., 2020) is used to build representations of a context and quote, and the model answers the question by selecting a text span (corresponding to the speaker name) from the document.

CEQA is another method which employs a pre-trained LLM, a BART encoder (Shao et al., 2021), to build contextual representations of the quotes. However, it differs in that it also builds an embedding for each character from other character mentions. Determining the speaker name for a quote involves computing the similarity between each character's embeddings and the quote.

ChatGPT has demonstrated a remarkable degree of general language understanding and is a state-of-the-art method for a diverse set of NLP tasks. However, the evaluation of ChatGPT in relation to the quote attribution task remains unexplored. We conduct preliminary tests using a variety of prompt formats, namely plain style, cloze style (Petroni et al., 2019), QA style (Brown et al., 2020), and code style (Zhang et al., 2023), while varying the set of examples ranging from zero-shot to sixteen-shot ranking, and permuting the order in which examples are provided. This choice is motivated by previous research highlighting the substantial influence of prompt format (Shin et al., 2020), example number (Brown et al., 2020), and example order (Lu et al., 2022) on model performance. The results of these experiments is presented in Appendix. We adopt the highest performing configuration (4-shot plain style prompt) in remaining experiments.

5. Benchmark Datasets

We gather a collection of 7 datasets (4 English, and 3 Chinese) annotated for the quote attribution task. An overview of these datasets is in Table 1.

5.1. English Datasets

P&P (He et al., 2013) is based on Jane Austen's 1813 novel, "*Pride and Prejudice*." Each quote is annotated with a character name selected from the provided character list by a single annotator.

QuoteLi (Muzny et al., 2017) contains three English novels: Jane Austen's "*Pride and Prejudice*", "*Emma*", and Anton Chekhov's "*The Steppe*". Although QuoteLi contains the entirety of the P&P dataset, the annotations are different. Two annotators assign a character name to each quote from a manually constructed character list (inter-annotator agreement Cohen's kappa $\kappa = 0.97$).

RIQUA (Papay and Padó, 2020) has eleven 19th-century fiction works by various authors, including four contemporary translations from French and Russian. Each quote in these books is attributed by three native English speakers with a pronoun (e.g., he/she), noun phrase (e.g.,the police/man), or name (e.g.,Jane/Elizabeth) that refers to a specific character entity (inter-annotator agreement 0.84 F1.

PDNC (Vishnubhotla et al., 2022) is the largest English quote attribution dataset, covering 22 fiction books written by different authors across multiple genres. Each quote in these books is annotated with a character name selected from the provided character list by two annotators.

5.2. Chinese Datasets

CSI (Yu et al., 2022) is based on 18 web novels spanning 10 genres, from 2010 to 2020. Paragraphs with quotes are regarded as utterances. Native Chinese speakers annotate the speaker's name span for each utterance paragraph within its context (Cohen's kappa $\kappa = 0.76$). The context is the text span bounded by the nearest narrative paragraphs before and after the utterance paragraph.

JY (Jia et al., 2020) includes three Chinese martial arts fiction books by Jin Yong. Two Chinese grad students choose a character from a character list for the central quote in an excerpt (annotation consistency rate of 0.94). The excerpt consists of the central quote, the five preceding sentences, and the five following sentences.

WP2021 (Chen et al., 2019) is derived from the Chinese novel "*World of Plainness.*" Excerpts with a central quote and 10 sentences before and after it are created for annotation. Chinese professional annotators select a character name from a provided list for annotating the central quote of each excerpt (annotation consistency rate of 0.94).

5.3. Dataset Unification

Variation in dataset annotation schemes and differing assumptions regarding model input pose challenges for benchmarking each model across the full collection of datasets. To reconcile this disparity and ensure compatibility with various models, we construct uniform data instances from the original datasets following our task definition in Section 3. This process involves constructing character lists, dialogues, and contexts, as well as partitioning the data into train, test, and validation splits.

Certain datasets like RIQUA and CSI lack character lists, which poses a problem for classification-based models which require such lists as input. For datasets without character lists (CSI and RIQUA) we automatically generate character lists using a previously established method (Cuesta-Lázaro et al., 2022).

For English datasets we extract dialogues from books by identifying consecutive paragraphs containing quotes. We set the context window $w = 10$ (ten paragraphs before and ten after).

Chinese datasets adopt a different annotation process, selecting book excerpts and annotating the central quote within each. Thus only one quote is annotated per excerpt, aligning with models attributing each quote individually to a speaker. We group excerpts into clusters, where each cluster consists of excerpts whose quotes consecutive in the original book. Consecutive quotes are those without intervening quotes and no more than one narrative sentence. Excerpts within the same cluster are merged to form the context. Each sentence or quote in the context is treated as a paragraph.

	Chinese			English			
	CSI	WP	JY	P&P	QuoteLi	RIQUA	PDNC
TSQA	23.4	36.8	32.4	48.6	38.5	23.6	38.6
ChatGPT	74.5	**88.4**	95.6	**93.9**	**89.0**	59.0	86.9
EEQA							
–base	85.2	75.0	97.5	63.6	56.0	67.0	80.6
–large	87.3	80.5	98.4	70.2	60.8	68.9	86.9
NNQA							
–base	85.6	76.8	97.8	78.3	67.7	68.4	84.3
–large	87.2	80.0	97.6	82.9	78.0	71.7	87.7
CEQA							
–base	86.7	80.4	98.5	68.6	68.6	71.6	87.7
–large	**89.7**	82.0	**99.2**	76.3	76.3	**74.6**	**92.9**
–gpt2	76.9	69.1	97.7	62.8	62.4	63.8	87.6
–bert	87.2	76.3	98.2	61.6	63.8	65.0	85.8
–roberta	89.6	87.1	98.8	81.3	74.2	73.6	92.6

Table 2: The accuracy of models across datasets.

Where possible, we partition the standardized datasets into train, validation, and test sets to maintain the same quote-partition assignments as in the original work. For the CSI, which lack validation sets, we allocate 10% of the training set to instead be used for validation. In the case of the PDNC and the RIQUA dataset, we divide the dataset according to the standard 0.8/0.1/0.1 split, as the original dataset did not come with predefined divisions.

6. Benchmark Results

We train each model using the training set of each dataset, following the settings of the original paper. The results of these experiments is presented in Table 2. Supervised models leveraging LLMs (EEQA, NNQA, and CEQA) consistently exhibit significantly higher accuracy compared to models which rely on manually crafted rules (TSQA) across all datasets. This observation underscores the limitations of rule-based approaches when confronted with a diverse set of scenarios as found within any of these datasets, or even within a single book.

Among the supervised models, the CEQA model achieves higher accuracy than other methods on all datasets apart from P&P. When considering that QuoteLi is very similar to P&P (and contains it), it is possible that the dataset size may be a contributing factor. The other differentiating factor is the type of LLM used in each, and respective training data used for each, where CEQA utilizes BART, in comparison to other models built on BERT.

Additionally, we evaluate LLM-based methods using larger variants of their original foundation model. For the CEQA model, we also test the performance with other foundation model architectures. Despite the original models already being state-of-the-art

Explicit “*He is also handsome,*” replied Elizabeth
Anaphoric-Pronoun “*Then what was it?*” She repeated.
Anaphoric-Other “*That is absurd,*” said the irritated man, sharply.
Implicit “Would he, you think?” asked Mark. “*Not in the least*” “Not surprising, I suppose.”

Table 3: Examples of quote types. The target quote is marked in italics, with the speaker in blue.

at time of publication, simply increasing the foundation model size leads to substantial increases in performance in all settings. On average, replacing the base with models 330 % larger resulted in performance improvements of 4.0 on average. While alternatives such as RoBERTa achieve higher performance than in the original work, a larger version of the BART encoder achieves state-of-the-art performance and outperforms all other configurations in all but three datasets (P&P, QuoteLi, and WP).

We conduct the first examination of ChatGPT on the task of quote attribution, measuring its performance on the zero-shot and few-shot setting. In the 4-shot setting (as shown) ChatGPT consistently delivers competitive results across most datasets. Particularly interesting is ChatGPT’s superior performance in smaller datasets like WP, P&P, and QuoteLi, demonstrating its effectiveness in low-resource settings when compared to other models.

7. Analysis

7.1. Quote Level

It is common in narrative text to label quotes with the speaker name, but when context is sufficient to determine who the speaker is, the name is sometimes dropped or replaced by a pronoun. We consider four types of quote mention, as describe in previous work (Muzny et al., 2017): Explicit, Anaphoric (pronoun), Anaphoric (other), and Implicit. The Explicit quotes mention a speaker name or alias in their surrounding text. Anaphoric (pronoun) quotes replace the speaker name with a pronoun, and Anaphoric (other) replace it with other text. This can be another identifier for the character, such as an occupation or a reference to their appearance, which is neither pronominal or a proper noun. Implicit quotes drop any reference to the speaker entirely. Examples of each type are shown in Table 3. As the degree of speaker information can vary drastically depending on the style of the quote mention, we explore the extent to which this affects performance.

			TSQA	EEQA	NNQA	CEQA	ChatGPT
WP	Exp.	0.14K	47	90	88	89	88
	Imp.	0.08K	19	55	63	67	86
JY	Exp.	5.12K	33	99	99	99	97
	Imp.	0.60K	28	88	89	92	89
CSI	Exp.	11.90K	25	92	92	93	80
	Imp.	2.60K	12	50	53	55	48
	Ana.(P)	0.35K	22	71	68	74	69
	Ana.(O)	0.18K	19	65	62	64	53
P&P	Exp.	0.07K	66	84	94	81	100
	Imp.	0.05K	45	43	65	55	82
	Ana.(P)	0.03K	14	54	71	68	100
	Ana.(O)	0.00K	-	-	-	-	-
RIQUA	Exp.	0.49K	28	89	92	91	61
	Imp.	0.29K	15	33	33	38	56
	Ana.(P)	0.01K	9	45	54	54	45
	Ana.(O)	0.00K	-	-	-	-	-
QuoteLi	Exp.	0.60K	57	75	88	82	96
	Imp.	0.33K	22	35	51	53	79
	Ana.(P)	0.17K	9	36	55	71	93
	Ana.(O)	0.03K	3	10	32	42	42
PDNC	Exp.	1.17K	63	94	92	96	91
	Imp.	0.95K	23	62	73	74	80
	Ana.(P)	0.53K	14	83	80	91	90
	Ana.(O)	0.08K	12	54	64	90	88

Table 4: The accuracy of each model on each of the four quote type categories:**Expl**icit, **Ana**phoric (**P**ronoun), **Ana**phoric (**O**ther), and **Impl**icit.

As datasets are not annotated with the types of quote mention, we preprocess each dataset using a set of heuristics. Following the established literary convention that quotes within the same paragraph are typically attributed to the same speaker (He et al., 2013), we classify quotes as Explicit when the speaker name is found in the proximity outside of the quotation but within the paragraph. In the absence of the speaker name, Anaphoric (pronoun) when they are accompanied by pronouns followed by speech tags (“*he said*”), and or Anaphoric (other) when speech tags are observed with no pronouns like “the man said.”. Remaining examples are classified as Implicit. Note that in datasets where paragraph boundaries are not clearly defined, quotes which are not classified as Explicit are considered Implicit.

7.1.1. The Effect of Quote Type

Table 4 presents the accuracy rates within each quote type category across all models and datasets. Performance is consistently highest on Explicit mentions, where the speaker names are mentioned around the quotes. In the Explicit category, accuracy is on average 81%. When explicit speaker names are absent, performance drops to 57% in Anaphoric-Pronoun, and 54% in Implicit categories on average. The substantial disparity in accuracy highlights the models' heavy reliance on the presence of speaker names and their limited inference capabilities when such names are absent. In the case of English datasets, models achieve 5.7 points higher accuracy in the Anaphoric category compared to the Implicit category. Pronouns and other textual descriptions referencing the speakers near the quotations, though not explicitly tied to speaker names, appear to still support model prediction. It is worth noting that Implicit quotes account for less than 30% of all quotes, compared to more than 50% in English datasets, suggesting that English datasets are more challenging.

7.1.2. Errors in Explicit Mentions

As discussed in Section 7.1.1, Explicit quotes contain strong indications of speaker identity, and it is therefore unsurprising that models perform better in this category than in others. However, Explicit quotes are also the most prevalent quote type, and the error rates for this category are still substantial. Consequently, errors of this type contribute considerably to declines in overall performance.

We hypothesize that mistakes in this relatively simple scenario may be due to (1) the interference caused by character names other than speakers mentioned around the quotes; or correlated with (2) the absence of speech tags to indicate which name mentioned around the quotes is the speaker name.

Confounding Speaker Names To test the role that confounding character names may play, we further divide Explicit quotes into two categories based on whether additional character names appear in the text surrounding the quote, and calculate accuracy within each category. Results are presented in Table 5, we observe that the accuracy within the category without additional names consistently approaches 100% across all Chinese datasets, significantly surpassing the performance of the other category. This outcome substantiates our hypothesis that the presence of additional character names around the quotes indeed introduces interference in predictive accuracy. The relatively prolific number of quotes where confounding speaker names are present, coupled with significantly lower model performance in this category, indicates that focused attempts to better identify confounding speakers may result in significant improvements in overall accuracy.

Speech Tags Dialogues between characters in narrative fiction often involves the presence of both speaker names and accompanying speech tags, such as "*Katharine asked*" or "*Mary said*," among others. However, for brevity or narrative purposes, authors sometimes omit these key words. We hypothesize that the absence of clear speech tags may increase the difficulty of the attribution task, especially in situations where confounding speaker names are also present. We further divide the set of quotes containing additional character names, splitting it into two distinct categories: one with speech tags and the other without. We then calculate accuracy separately for each category. Results are presented in Table 6. We find that accuracy is significantly higher in the category with speech tags, by an average of 7.4 points over those without. Examining the effect on the highest performing models, we can observe that on a diverse set of contemporary books (PDNC), CEQA performs 4 points worse when speech tags are not present. While we cannot rule out other confounding factors that appear in similar ratios in these two categories, our analysis indicates that improving performance on this scenario may have the ability to significantly improve overall accuracy.

			TSQA	EEQA	NNQA	CEQA	ChatGPT
CSI	w/	11.52K	26	92	92	93	80
	w/o	0.35K	19	96	97	97	88
WP	w/	0.09K	43	85	82	86	83
	w/o	0.05K	53	97	100	95	97
JY	w/	4.08K	31	99	99	99	96
	w/o	1.04K	37	98	98	99	97
P&P	w/	0.05K	65	78	92	76	100
	w/o	0.02K	64	100	100	94	100
QuoteLi	w/	0.39K	52	70	85	77	94
	w/o	0.21K	66	82	93	90	98
RIQUA	w/	0.46K	29	89	91	90	61
	w/o	0.03K	22	92	96	100	51
PDNC	w/	0.56K	53	93	89	94	91
	w/o	0.61K	71	95	95	97	90

Table 5: The accuracy of each model in the categories with/without the confounding speaker names mentioned around the quotes in Explicit category.

In conclusion, our analysis reveals that confounding speaker names and the absence of speech tags both appear to have a detrimental effect on model performance on Explicit quote attribution. Moreover, it is noteworthy that an average of 36% of all quotes fall into this category, underscoring the need for future research to develop models capable of mitigating the challenges posed by both the presence of additional character names and the absence of speech tags.

7.1.3. Anaphoric and Implicit Errors

The overall trend in the results shows a significant drop in performance when the speaker is not explic-

			TSQA	EEQA	NNQA	CEQA	ChatGPT
CSI	w/	1.85K	39	96	95	97	86
	w/o	9.68K	23	92	91	92	79
WP	w/	0.02K	52	82	90	81	91
	w/o	0.07K	40	86	79	87	80
JY	w/	0.94K	39	99	99	99	96
	w/o	3.14K	29	98	98	99	96
P&P	w/	0.02K	81	81	100	90	100
	w/o	0.04K	61	78	90	73	100
QuoteLi	w/	0.13K	64	82	88	84	95
	w/o	0.27K	47	65	84	74	94
RIQUA	w/	0.22K	32	97	98	97	61
	w/o	0.24K	26	81	86	83	62
PDNC	w/	0.26K	62	98	95	96	92
	w/o	0.31K	46	89	83	92	90

Table 6: The accuracy of each model in categories where the speakers are with/without speech tags when the speakers are around the quotes.

			TSQA	EEQA	NNQA	CEQA	ChatGPT
CSI	w/	2.91K	14	53	56	57	50
	w/o	0.17K	9	49	53	70	56
WP	w/	0.03K	22	61	77	62	87
	w/o	0.05K	17	50	53	69	84
JY	w/	0.45K	26	88	87	92	89
	w/o	0.14K	36	89	92	90	89
P&P	w/	0.06K	28	45	68	63	91
	w/o	0.02K	52	52	63	47	78
QuoteLi	w/	0.37K	15	32	52	59	81
	w/o	0.17K	19	37	48	56	80
RIQUA	w/	0.28K	14	33	32	38	55
	w/o	0.01K	27	33	66	50	54
PDNC	w/	0.75K	18	66	72	78	84
	w/o	0.81K	21	71	77	82	83

Table 7: The accuracy in the categories with/without additional character names around the quotes of Anaphoric and Implicit categories.

itly mentioned in the quote context, indicating that the models lack ability to precisely infer the speaker names from the context when the speaker names are not explicitly mentioned around the quotes. But what contextual cues do the models fail to identify? We again examine two possible causal factors for this performance drop: (1) the interference of additional character names mentioned near the quotes, as examined previously; and (2) a limited capacity to model dialogue structure for speaker inference.

Confounding Speaker Names We divide the set of Anaphoric quotes, again into two categories: one where confounding character names are mentioned in the quote context, and the other without. Table 7 presents the accuracy of the models on each category. The accuracy of quotes without confounding speakers does not exhibit a trend of superiority over those with confounding speakers. This observation differs from the patterns of model accuracy observed in the Explicit category. On average, performance is 1.6 points higher in cases without confounding speakers, compared to 6.0 points in the Explicit category.

Dialogue Structure In Anaphoric quotes, as opposed to Explicit quotes, a greater degree of contextual awareness is necessary to resolve the reasoning chain which connects a quote to other preceding quotes by the same speaker, and ultimately to the target speaker name. We hypothesize that failures to accurately understand dialogue structure may have an influence on quote attribution in this category. We focus on two dialogue characteristics for which we can automate analysis: (1) Fixed participants: In dialogues, many character names could be mentioned, but as only a subset of speakers are typically present in any given scene, many speakers can be removed from consideration, (2) Alternative Speaking Order: Dialogue participants take turns speaking (this is especially prevalent in 2-speaker conversations).

	Chinese			English			
	CSI	JY	WP	P&P	QuoteLi	RIQUA	PDNC
TSQA	71.3	65.4	61.0	49.4	57.1	58.9	56.8
EEQA	32.8	10.4	14.3	59.5	53.3	44.1	26.8
NNQA	30.2	9.5	11.4	12.7	20.9	35.9	8.0
CEQA	29.0	6.9	13.3	6.3	17.2	29.5	6.0
ChatGPT	47.6	9.8	7.8	1.3	14.1	34.7	10.9

Table 8: The percentage of out-of-scope errors in the Anaphoric and Implicit categories.

Out-of-scope prediction Models with limited understanding of broader narrative structure may lack awareness of the participants engaged in each dialogue, and may erroneously attribute quotes to characters who are not present. We seek to measure the extent to which this occurs. We calculate the number of out-of-scope errors within the Anaphoric and Implicit categories, and present the results in Table 8. All models exhibit varying degrees of making out-of-scope predictions across different datasets, which appears to scale proportionately to overall error rate, with no clear advantage to any specific model type.

Repeated prediction When inferring the speakers for implicit quotes, the absence of awareness dialogue structure may bias the models to over-predict the same speaker, even for consecutive utterances. As the role of the paragraph is to signal the end of the speaker's turn, this is an exceedingly uncommon pattern, existing in only 4.0% of utterances.

	Chinese			English			
	CSI	JY	WP	P&P	QuoteLi	RIQUA	PDNC
TSQA	17.7	1.7	28.6	15.2	21.2	36.0	22.9
EEQA	5.1	0.3	18.2	24.1	18.0	25.4	13.9
NNQA	5.7	0.4	11.4	10.1	14.9	26.1	7.9
CEQA	5.4	0.2	9.3	12.7	12.6	16.4	7.6
ChatGPT	0.3	0.0	1.3	5.1	1.9	6.1	3.5

Table 9: The percentage of repetition errors in the Anaphoric and Implicit categories.

To assess the models' awareness of this characteristic, we measure repetition error rates within the Anaphoric and the Implicit categories. The results, as depicted in Table 9, reveal the prevalence of repetition errors across all models to varying degrees. Within this category, ChatGPT exhibits the lowest error rates among all the models, an improvement of 6.5 points on average compared to CEQA. ChatGPT has shown to be capable of performing complex reasoning of latent structure (as demonstrated in code completion and refactoring tasks), and may benefit from being optimized to perform such tasks when compared to other models which make simpler classification decisions.

In summary, in analyzing the role of quote type in overall performance, we find complementary strengths: supervised methods like CEQA achieve the highest performance on explicit mentions and in resolving anaphora where character embeddings can be exploited, but ChatGPT excels at adhering to common discourse structure constraints. Models which leverage the strengths of both models may result in overall improvements in accuracy.

7.2. Book Level

Each genre of fiction has its own unique writing style, and may have different high-level characteristics with important implications for the quote attribution task. For instance, the Romance genre is often considered dialogue-driven, and may contain long discourses between two characters. In this section we explore the extent to which genre differences between train and test sets impact attribution accuracy.

We focus on the PDNC dataset, which contains a diverse set of book genres. We manually label each book with a genre tag based on their ISBN entry in WorldCat. We categorize books into five genres: *Children/Adventure*, *Classic*, *Detective/Mystery*, *Period/Romance*, and *Science(Fiction)/Fantasy*. We employ a rotating selection process wherein we alternated the choice of genre to serve as the test set, with the remaining groups comprising the training and validation sets. Each training set contained 15,000 instances, while the validation and test sets each included 1,000 instances. These partition sizes were chosen to ensure enough data was available even when evaluating the smaller genre partitions.

	Children Adventure	Classic	Detective Mystery	Period Romance	Science Fantasy
TSQA	62	37	32	34	43
EEQA	88	62	56	66	74
NNQA	88	75	76	79	87
CEQA	94	76	82	81	90
ChatGPT	91	83	89	90	84
Avg.	85	67	67	70	76

Table 10: The accuracy of each model on different book genres.

	Children Adventure	Classic	Detective Mystery	Period Romance	Science Fantasy
Exp.	76(89)	30(89)	28(89)	42(87)	56(84)
Imp.	12(68)	50(56)	46(56)	33(56)	18(59)
Ana.(P)	12(75)	18(63)	24(62)	23(63)	25(69)
Ana.(O)	0(27)	2(49)	2(46)	2(47)	2(30)

Table 11: The distribution of the quote types in different genres. The average accuracy of all models within each quote type is also presented in parentheses.

Results are shown in Table 10. We observe clear genre-based performance trends, with the Children/Adventure genre being considerably easier than others. There are obvious explanations for this performance difference, for instance, Children's books may contain simpler language / more Explicit quotes since the target readers are younger. To examine the extent to which performance differences across genre are correlated with the genre quote type distribution, we calculate the distribution of quote type within each genre (Table 11). For the Children/Adventure drama, we do find a parallel between the high proportion of Explicit quotes and the high model performance. But there are yet more exceptions to this trend, and genres like Classic contain an identically high proportion of Explicit quote types while being the most difficult genre for both CEQA and ChatGPT. We therefore conclude that varying genres pose difficulties which are not fully capture simply by the distribution of quote type.

8. Conclusion

In this work, we formalize the quote attribution task and present a benchmark using a diverse set of publicly available models and datasets. Our benchmark shows that the CEQA model is the state-of-the-art supervised model, but also finds that ChatGPT can perform comparably on most datasets while being more robust to genre effects. Despite overall strong performance in many scenarios, we also observe clear areas for improving accuracy. Existing models are still susceptible to confounding speaker names in the quote context, and show an apparent lack of tracking discourse participants or modeling of dialogue turn taking. We release all code and model predictions in the hopes of facilitating further research in quote attribution and promoting more systematic comparisons with previous work.

9. Limitations

This study utilizes a set of publicly available datasets for the task of quote attribution, and many limitations stem from the relative dearth of large-scale resources for the evaluation of quote attribution. This is reflected in our study covering only two languages, and there are likely many different practices for narrative writing of dialogues in other languages which we cannot analyze. It is also possible that our findings regarding the rankings of models would be different if tested on other languages. Similarly, many of the books found in the datasets used in this work are hundreds of years old, as popular contemporary books are protected by copyright. Additionally, datasets for quote attribution are small when compared to many other NLP tasks, and this increases the impact of biases. As it is an expensive proposition to construct a significantly larger annotated corpus for quote attribution, across languages, genre, author, and date, we hope that the exhaustive nature of our evaluations across all existing datasets helps provide a clearer picture of the current state of the field.

Acknowledgements

This study is based on the results obtained from a project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). For experiments, the computational resource of the AI Bridging Cloud Infrastructure (ABCI) provided by the National Institute of Advanced Industrial Science and Technology (AIST) were used.

The icons in Fig 1 are made by tulpahn, aDara, Victoruler, Slidicon, and riajulislam from www.flaticon.com.

10. Bibliographical References

Mariana S. C. Almeida, Miguel B. Almeida, and André F. T. Martins. 2014. A joint model for quotation attribution and coreference resolution. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 39–48. The Association for Computer Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Yue Chen, Zhen-Hua Ling, and Qing-Feng Liu. 2021. A neural-network-based approach to identifying speakers in novels. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, pages 4114–4118. ISCA.

Carolina Cuesta-Lázaro, Animesh Prasad, and Trevor Wood. 2022. What does the sea say to the shore? A BERT based DST style approach for speaker to dialogue attribution in novels. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5820–5829. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 657–668. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Adam Ek, Mats Wirén, Robert Östling, Kristina Nilsson Björkenstam, Gintare Grigonyte, and Sofia Gustafson-Capková. 2018. Identifying speakers and addressees in dialogues extracted from literary fiction. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA).

David K. Elson and Kathleen R. McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*. AAAI Press.

Kevin R. Glass and Shaun Bangay. 2007. A naïve, salience-based method for speaker identification in fiction books.

Hua He, Denilson Barbosa, and Grzegorz Kondrak. 2013. Identification of speakers in novels. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1312–1320. The Association for Computer Linguistics.

Prashant Jayannavar, Apoorv Agarwal, Melody Ju, and Owen Rambow. 2015. Validating literary theories using automatic social network extraction. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature, CLfL@NAACL-HLT 2015, June 4, 2015, Denver, Colorado, USA*, pages 32–41. The Association for Computer Linguistics.

Vincent Labatut and Xavier Bost. 2019. Extraction and analysis of fictional character networks: A survey. *ACM Comput. Surv.*, 52(5):89:1–89:40.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8086–8098. Association for Computational Linguistics.

Nuno J. Mamede and Pedro Chaleira. 2004. Character identification in children stories. In *Advances in Natural Language Processing, 4th International Conference, EsTAL 2004, Alicante, Spain, October 20-22, 2004, Proceedings*, volume 3230 of *Lecture Notes in Computer Science*, pages 82–90. Springer.

Grace Muzny, Michael Fang, Angel X. Chang, and Dan Jurafsky. 2017. A two-stage sieve approach for quote attribution. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 460–470. Association for Computational Linguistics.

Timothy O'Keefe, Silvia Pareti, James R. Curran, Irena Koprinska, and Matthew Honnibal. 2012. A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 790–799. ACL.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.

Andrew Salway, Paul Meurer, Knut Hofland, and Øystein Reigem. 2017. Quote extraction and attribution from norwegian newspapers. In *Proceedings of the 21st Nordic Conference on Computational Linguistics, NODALIDA 2017, Gothenburg, Sweden, May 22-24, 2017*, volume 131 of *Linköping Electronic Conference Proceedings*, pages 293–297. Linköping University Electronic Press / Association for Computational Linguistics.

Yunfan Shao, Zhichao Geng, Yitao Liu, Junqi Dai, Fei Yang, Li Zhe, Hujun Bao, and Xipeng Qiu. 2021. CPT: A pre-trained unbalanced transformer for both chinese language understanding and generation. *CoRR*, abs/2109.05729.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20,* *2020*, pages 4222–4235. Association for Computational Linguistics.

Jianzhu Yao, Ziqi Liu, Jian Guan, and Minlie Huang. 2022. A benchmark for understanding and generating dialogue between characters in stories. *CoRR*, abs/2209.08524.

Chak Yan Yeung and John Lee. 2017. Identifying speakers and listeners of quoted speech in literary works. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017, Volume 2: Short Papers*, pages 325–329. Asian Federation of Natural Language Processing.

Dian Yu, Ben Zhou, and Dong Yu. 2022. End-to-end chinese speaker identification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2274–2285. Association for Computational Linguistics.

Jason Y. Zhang, Alan W. Black, and Richard Sproat. 2003. Identifying speakers in children's stories for speech synthesis. In *8th European Conference on Speech Communication and Technology, EUROSPEECH 2003 - INTERSPEECH 2003, Geneva, Switzerland, September 1-4, 2003*, pages 2041–2044. ISCA.

Li Zhang, Hainiu Xu, Yue Yang, Shuyan Zhou, Weiqiu You, Manni Arora, and Chris Callison-Burch. 2023. Causal reasoning of entities and events in procedural texts. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 415–431. Association for Computational Linguistics.

Yuanchi Zhang and Yang Liu. 2022. Directquote: A dataset for direct quotation extraction and attribution in news articles. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pages 6959–6966. European Language Resources Association.

Ben Zhou, Dian Yu, Dong Yu, and Dan Roth. 2022. Cross-lingual speaker identification using distant supervision. *CoRR*, abs/2210.05780.

11. Language Resource References

Jia-Xiang Chen, Zhen-Hua Ling, and Li-Rong Dai. 2019. A chinese dataset for identifying speakers in novels. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 1561–1565. ISCA.

Hua He, Denilson Barbosa, and Grzegorz Kondrak. 2013. Identification of speakers in novels. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1312–1320. The Association for Computer Linguistics.

Yuxiang Jia, Huayi Dou, Shuai Cao, and Hongying Zan. 2020. Speaker identification and its application to social network construction for chinese novels. In *International Conference on Asian Language Processing, IALP 2020, Kuala Lumpur, Malaysia, December 4-6, 2020*, pages 13–18. IEEE.

Grace Muzny, Michael Fang, Angel X. Chang, and Dan Jurafsky. 2017. A two-stage sieve approach for quote attribution. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 460–470. Association for Computational Linguistics.

Sean Papay and Sebastian Padó. 2020. Riqua: A corpus of rich quotation annotation for english literary text. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 835–841. European Language Resources Association.

Krishnapriya Vishnubhotla, Adam Hammond, and Graeme Hirst. 2022. The project dialogism novel corpus: A dataset for quotation attribution in literary texts. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pages 5838–5848. European Language Resources Association.

Dian Yu, Ben Zhou, and Dong Yu. 2022. End-to-end chinese speaker identification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2274–2285. Association for Computational Linguistics.

A. Prompt Setting

To determine the optimal prompt settings for ChatGPT (gpt-3.5-turbo-16k-0613) on the P&P dataset, we first assess the impact of prompt format, including Plain, QA, Code, and Cloze styles (see Table 16 for examples). We evaluate ChatGPT with these prompt styles in a zero-shot setting, and the results in Table 12 indicate that the Plain style performs the best.

We investigated the optimal number of examples in prompts, ranging from zero to sixteen, encompassing Explicit, Anaphoric(pronoun), Anaphoric(other), and Implicit quotes with 20 randomly selected examples for each type. Prompts were constructed by filling them with N randomly ordered examples, and ChatGPT was evaluated ten times with different orderings. This process was repeated five times for the N-example setting, with each iteration using a different group of N examples. The average accuracy across all groups and orders indicates the performance for that N-example setting. Table 13 displays these results, revealing improved performance up to four-shot prompts, making four-shot examples the recommended choice.

Additionally, we explore the impact of example order within the four-shot setting. We define ten different ranking orders and repeatedly evaluate ChatGPT five times for each order, using different example groups each time. The average score for each order is taken as its performance. Table 14 presents these results, illustrating that example order does indeed influence performance.

Style	Plain	Cloze	QA	Code
Acc.	91	76	85	18

Table 12: The evaluation of ChatGPT with different prompt formats across different datasets.

#Shot	0	1	2	3	4	8	12	16
Acc.	91.0	91.1	92.0	92.8	93.2	92.9	92.7	92.1

Table 13: The accuracy of ChatGPT under different shot-number settings.

B. Training details

In this section, we will provide the training details for each model.

NNQA The NNQA model is based on the BERT architecture. For training on Chinese datasets, we utilized the Chinese BERT-base model, which was released by Google Research. For English datasets, we downloaded the BERT-base and BERT-large models from the Huggingface website.

To enhance the model's performance, we conducted a meticulous hyperparameter search focused on the learning rate, spanning from 5e-6 to 1e-4 across various datasets. For the BERT-base model, we set the learning rate to 8e-6 for CSI, JY, RIQUA, P&P, and PDNC datasets, 5e-6 for the WP2021 dataset, and 3e-6 for QuoteLi datasets, maintaining a consistent batch size of 16. Conversely, for the BERT-large model, we employed a learning rate of 5e-6 for CSI, WP2021, P&P, QuoteLi, and PDNC datasets, 1e-5 for the JY dataset, and 8e-6 for the RIQUA dataset. The batch size for the BERT-large model was set to 24.

Training specifics include utilizing a BART-large model on a single NVIDIA A100-SXM4-40GB, while a BART-base model was trained on a single Tesla V100-SXM2-16GB for 25 epochs.

EEQA EEQA is another BERT-based model. For Chinese datasets, we selected the BERT-wwm-ext-base (Cui et al., 2020) and RoBERTa-wwm-large, while for English datasets, we opted for the BERT-base and BERT-large models provided by Huggingface due to their codes are incompatible with the RoBERTa models.

To optimize the model performance, we meticulously adjusted the learning rates for each dataset. Specifically, for the BERT-base model, we set the learning rate to 3e-5 for the CSI, P&P, and RIQUA datasets, 8e-6 for the WP2021 and JY datasets, and 6e-5 for the QuoteLi and PDNC datasets. As for the BERT-large model, we set the learning rates to 8e-6 for the CSI and JY datasets, and to 3e-5 for the WP2021, P&P, QuoteLi, RIQUA, and PDNC datasets.

The batch size for the base model is 24, and for the larger model, it is 12. All these models were trained with the aid of 4 Tesla V100-SXM2-16GB GPUs.

CEQA CEQA is based on a BART. The models BART-base, BART-large, BERT-base, RoBERTa-base, and GPT2-base were sourced from the Huggingface website.

To optimize model performance, specific learning rates were tailored for each dataset. The BART-base model employed a learning rate of 3e-5 for the CSI, JY, and PDNC datasets, 6e-5 for the WP2021 and RIQUA datasets, and 1e-4 for the P&P and QuoteLi datasets. Conversely, the BART-large model utilized a learning rate of 3e-5 for the CSI, P&P, and QuoteLi datasets, 6e-5 for the WP2021 dataset, 1e-5 for the JY and RIQUA datasets, and 8e-6 for the PDNC dataset.

Order	1	2	3	4	5	6	7	8	9	10
Acc.	93.0	92.2	92.2	92.8	93.6	93.6	93.0	93.0	92.2	93.4

Table 14: The impact of example order on the accuracy of ChatGPT under 4-shot setting.

The RoBERTa-base model was trained with a learning rate of 1e-5 for the CSI and JY datasets, 3e-5 for the WP2021, RIQUA, and PDNC datasets, and 6e-5 for the other datasets. For the GPT2-base model, learning rates varied with a setting of 1e-5 for the CSI and JY datasets, 3e-5 for PDNC, 6e-5 for WP2021, 6e-4 for RIQUA, and 1e-3 for the P&P and QuoteLi datasets. Similarly, the BERT-base model employed different rates: 1e-5 for CSI, 3e-5 for JY, RIQUA, and PDNC datasets, and 6e-5 for WP2021, P&P, and QuoteLi datasets.

All models were trained for 25 epochs with a batch size of 8. The base models were trained on a single Tesla V100-SXM2-16GB, and the larger models on a single NVIDIA A100-SXM4-40GB.

	Original	Standard
Chinese	Sun Shaoan was so anxious · · · . His wife · · · , saying, “Don’t be in a hurry, we’ll figure something out. · · · ” “We haven’t even paid back the money we borrowed last time!” (Speaker: Sun Shaoan) Shaoan hung his head in despair,· · · . “How about you go to the county office again and find County Chief Zhou?” Xiulian suggested another idea. Sun Shaoan thought his wife’s idea had some merit. · · ·	Sun Shaoan was so anxious · · · . His wife · · · , saying, “Don’t be in a hurry, we’ll figure something out. · · · ” “We haven’t even paid back the money we borrowed last time!” (Speaker: Sun Shaoan) Shaoan hung his head in despair,· · · . “How about you go to the county office again and find County Chief Zhou?”(Speaker: Xiulian) Xiulian suggested another idea. Sun Shaoan thought his wife’s idea had some merit. · · ·
	Sun Shaoan was so anxious · · · . His wife · · · , saying, “Don’t be in a hurry, we’ll figure something out. · · · " “We haven’t even paid back the money we borrowed last time!” Shaoan hung his head in despair,· · · . “How about you go to the county office again and find County Chief Zhou?” (Speaker: Xiulian) Xiulian suggested another idea. Sun Shaoan thought his wife’s idea had some merit. · · ·	
English	The man was Julius Beaufort. “Ah !” Archer cried, · · · . (Speaker: Archer) Madame Olenska had sprung up and · · · . “So that was it?” Archer said derisively. (Speaker: Archer) “I didn’t know he was here,” Madame Olenska murmured.· · · (Speaker: Madame Olenska) “Hallo, Beaufort this way! Madame Olenska was expecting you,” he said. (Speaker: Archer) During his journey back to New York · · ·	The man was Julius Beaufort. “Ah !” Archer cried, · · · . (Speaker: Archer) Madame Olenska had sprung up and · · · . “So that was it?” Archer said derisively. “I didn’t know he was here,” Madame Olenska murmured.· · · “Hallo, Beaufort this way! Madame Olenska was expecting you,” he said. During his journey back to New York · · ·
		The man was Julius Beaufort. “Ah !” Archer cried, · · · . Madame Olenska had sprung up and · · · . “So that was it?” Archer said derisively. (Speaker: Archer) “I didn’t know he was here,” Madame Olenska murmured.· · · (Speaker: Madame Olenska) “Hallo, Beaufort this way! Madame Olenska was expecting you,” he said. (Speaker: Archer) During his journey back to New York · · ·

Table 15: The examples of unifying the datasets. The left column **Original** is the fragments of annotation taken from the datasets. The right column **Standard** is the dialogues constructed from the fragments. The context of the dialogues is highlighted with blue color and the dialogue contents are highlighted with purple color.

Plain	Given a snippet of text containing a conversation from a fiction book, identify the speaker name for the specified quote. The speaker name should be selected from the character list.
	{Example 1}
	· · ·
	{Example N}
	Conversation:{}
	Character list:{}
	Specified quote:{}

Cloze	Given a snippet of text containing conversations from a fiction book, you are required to fill the blank with a correct character name. The character name should be selected from the character list.
	{Example 1}
	· · ·
	{Example N}
	Conversation: {}
	Character list:{}
	In the conversation, character __ is the speaker of the quote "{}"

QA	Give a snippet of text containing conversations from a fiction book, I want to know the speaker name of the specified quote in the conversation. the speaker name should be selected from the character list.
	{Example 1}
	· · ·
	{Example N}
	Conversation:{}
	Character list:{}
	Question: Who is the speaker for the quote '{}'?
	Answer:

Code	#Identify the speaker name for the specified quote in the conversation by executing the following codes and return the speaker name. The speaker name should be selected from the character list.
	{Example 1}
	· · ·
	{Example N}
	def identify_speaker_name():
	    conversation = "{}"
	    character_list="{}"
	    pecified_quote = "{}"
	    return get_speaker_name_for_quote(conversation,character_list,specified_quote)

Table 16: The format of different prompts. When constructing the prompts, the placeholder brackets are replaced with corresponding contents.