From Narratives to Portraits: Automating Elderly Portraits Construction through Named Entity Recognition with GPT-3.5

Anonymous ACL submission

Abstract

Elderly portraits, descriptive summaries of older individuals, aid caregivers in offering personalized care. However, the manual construction of these portraits is time-consuming and challenging. This paper introduces an automated framework for constructing elderly portraits with event elements. The primary objective is to efficiently extract relevant features from elderly narratives. Traditional named entity recognition (NER) methods often falter due to data limitations and the inherent complexity of the stories. To address this, we present EPNER (Elderly Portrait Named Entity Recognition), a NER approach leveraging in-context learning with large language models. Our experimental results confirm that EPNER surpasses existing techniques.

1 Introduction

Portraits refers to a descriptive summary of that individual, by collecting, organizing, and analyzing an individual's pertinent information and characteristics (Maes, 2015). This information and characteristics span multiple dimensions, encompassing, but not limited to, an individual's basic background (e.g., age, gender, occupation), hobbies and interests, behavioral habits, social relations, consumer behaviors, preferences, and values (Spiliotopoulos et al., 2020).

Every elderly individual possesses unique life experiences, personal preferences, distinctive values, and individual needs. These factors culminate in their unique personalities. By delving into the narratives of the elderly, we can gain insights about their past life experiences, family backgrounds, career trajectories, hobbies, and pivotal interpersonal relationships. These characteristic tags can be employed to craft a more comprehensive and meticulous elderly individual portraits. By amalgamating these characteristic insights with associated event information, one can intertwine the elderly's personality traits with their life experiences, making

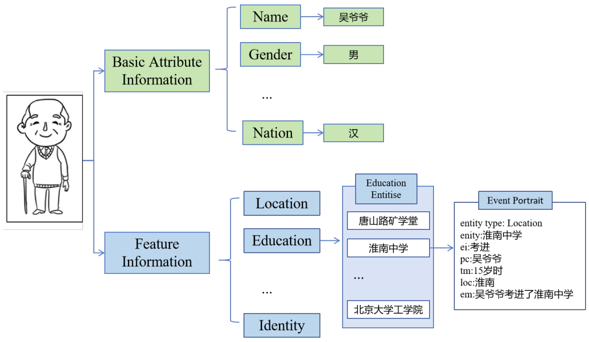

Figure 1: Elderly portraits combined with event elements

the portraits more pragmatically valuable. Figure 1 shows an example of a portrait of an elderly person combined with event elements.

A wealth of information pertinent to elderly portraits can be extracted from narratives about the elderly. These narratives, often filled with intricate life details. A pressing challenge in this domain, however, is how to automate the process of named entity recognition (NER) for elderly portrait entities from these narratives, as well as how to extract relevant events efficiently. Given the unique nature of elderly narratives, which often come with their own set of complexities such as redundancy, limited data, and intricate entity types, conventional NER methods often fall short in delivering accurate results.

Large Language Models (LLMs) like GPT-3 (Brown et al., 2020), LLaMA (Touvron et al., 2023) and GPT-4 (OpenAI, 2023). Renowned for their exceptional In-Context Learning (ICL) capabilities (Liu et al., 2021), these models can perform a myriad of tasks, ranging from text generation to complex problem-solving, often with minimal instruction. By tapping into the few-shot learning prowess of such models, many of the aforementioned challenges in elderly portrait construction can be effectively addressed (Wang et al., 2023; Gao et al., 2023). In light of this, our paper introduces a framework for the automated construction of elderly portraits leveraging these large language models. The contributions of this paper are delineated as follows:

We introduce an elderly portrait generation framework rooted in the narratives of the elderly. Utilizing generative large language models, we extract elderly portrait entities from the narrative texts and, in conjunction with event elements, construct a portrait encapsulating the life experiences of the elderly.

We devise an ICL based named entity recognition method EPNER (Elderly Portrait Named Entity Recognition), specifically tailored for the extraction of portrait entities from elderly narratives. This method utilizes the ICL capability of LLMs and adds protagonist feature as context to the prompt, effectively improving the efficiency of entity recognition.

We conduct experiments on Older Adults' Life Stories (OALS) Dataset (An et al., 2023). We compare our method swith existing few-shot named entity recognition methodologies, empirical evaluations corroborate that our proposed approach surpasses baseline methods in terms of accuracy. Furthermore, through ablation studies, we authenticate the efficacy of our prompt construction strategy.

We devise an ICL based event extraction method EPEE (Elderly Portrait Event eExtraction) by adding portrait entities as context to the prompt. To assess the efficacy of EPEE, manual evaluations were conducted from six distinct perspectives. Experimental outputs suggest that the event elements extracted by the EPEE do indeed exhibit a high correlation with elderly portrait entities.

2 Framework for Automatic Generation of Elderly Portraits

As depicted in Figure 2, the Framework for automatic generation of elderly portraits is presented. Drawing from OALS Dataset as our data source, we initially partition the elderly narratives according to the main protagonist. For each older individual, an elderly narrative set is constructed, denoted as $S = \{s_1, s_2, ..., s_n\}$, where s_i represents a segment of the elderly narrative. Given a specific input s_i, our objective is to discern the elderly portrait entities contained within s_i. These elderly portrait entities span six categories, as delineated in Table 1. Upon the completion of named entity recognition for the elderly portrait elements, specific events corresponding to the identified entity are subsequently extracted from s_i. These elderly story events are represented as $event = \{ei, pc, tm, loc, em\}$, where ei characterizes the action of the event, pc specifies the involved characters, tm indicates the temporal occurrence of the event, loc pinpoints the event's location, and em is the summary of the event. By consolidating the portrait entities from each s_i in S that contain corresponding story events, a comprehensive elderly portrait for an individual is formulated (as showed in Figure 1).

Table 1: Elderly Portraits Entity illustrate

Entity name	Entity illustrate
Location	This entity refers to the geographical information mentioned within the life stories of elderly individuals.
Health	This entity pertains to aspects such as diseases, physiological indicators, injuries, medication intake, mental health, daily living abilities, and passing away due to illnesses.
Interest	This entity encompasses activities such as reading, singing, physical exercise, arts, writing, board and card games, and gardening.
Identity	This entity refers to an individual's status, title, or position within an organization or society, with examples including manager, director, president, or student council president.
Occupation	This entity pertains to the regular, salaried work or duties that an individual engages in for livelihood, such as doctors, teachers, engineers, or waitstaff.
Education	This entity denotes the level of education an elderly individual has received, including primary school, middle school, high school, and university.

3 Methodology

3.1 Elderly Portrait Named Entity Recognition

We introduce a Named Entity Recognition (NER) method, EPNER (Elderly Portrait Named Entity Recognition), which leverages the in-context learning capabilities of Large Language Models (LLMs). The workflow of this method is illustrated in Figure 3. For a given input s_i, we construct a prompt $prompt(s_i)$ as the input to GPT-3.5, resulting in a generated output sequence $text_{output}$. Then, we transform the output sequence $text_{output}$ into BIO format for entity recognition through a parser. $label = Parser(text_{output})$

3.1.1 Prompt Construction

Distinct from existing prompt construction methods for NER tasks (Wang et al., 2023), the prompt construction method proposed in this paper incorporates elements of the protagonist about elderly narratives as knowledge into the prompt.This can guide the LLMs to more easily find portrait entities related to the protagonist from redundant information.

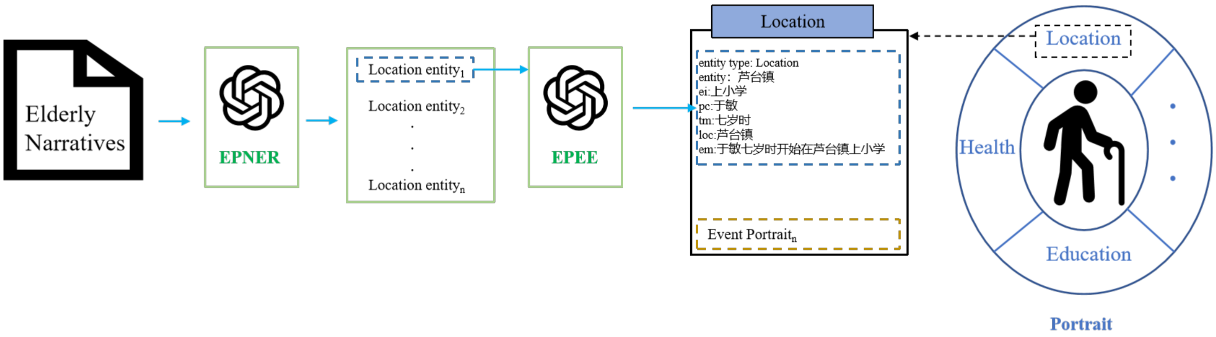

Figure 2: Framework for Automatic Generation of Elderly Portraits

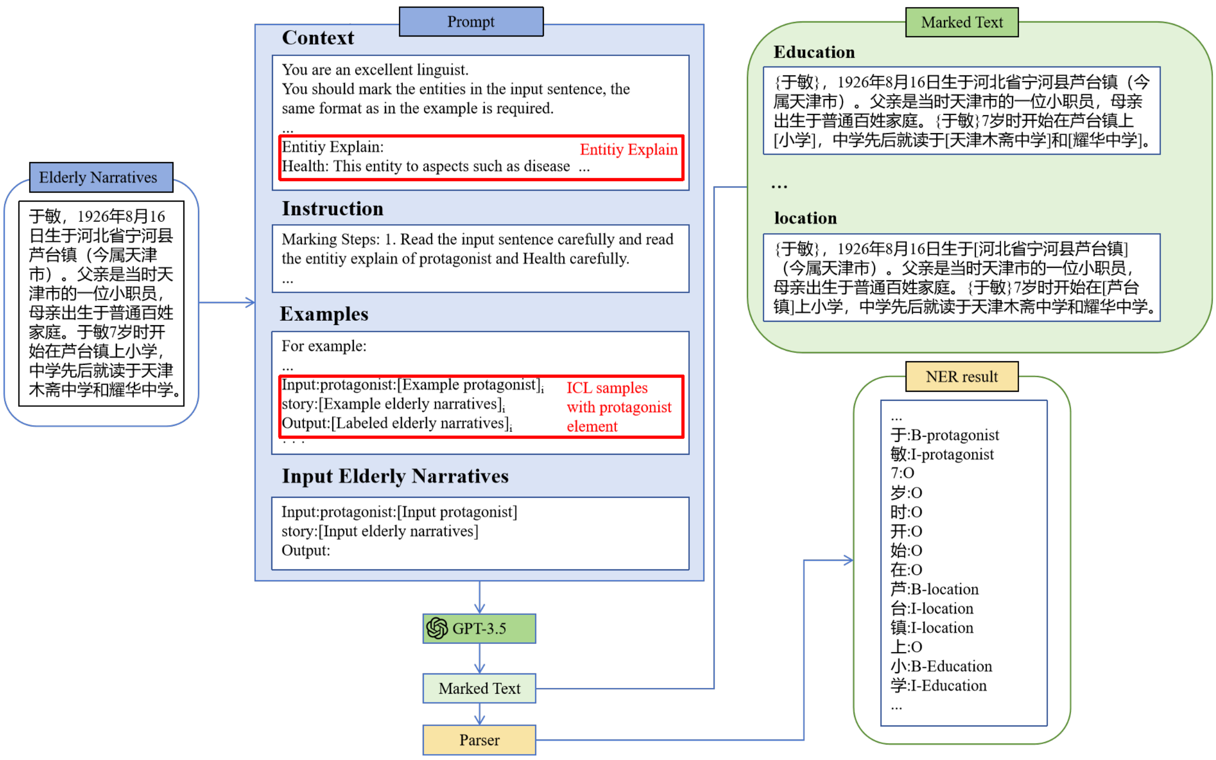

Figure 3: Elderly Portrait Named Entity Recognition

You are an excellent linguist.

You should mark the entities in the input sentence, the same format as in the example is required.

Read and understand these instructions carefully.

You can not omit any original information or add or modify the original text during annotation.

Entity Explain:

Protagonist: This entity refers to the central character or leading figure in the narratives.

Health:This entity to aspects such as diseases, physiological indicators, injuries, medication intake, mental health, daily living abilities, and passing away due to illnesses

Figure 4: Context within Prompt

In addition, to realize the automated identification of entities in elderly portraits, it is essential to ensure that the output results of LLMs conform to the format required for parser input.

The construction of prompts is comprised of four parts: 1) Context, 2) Instruction, 3) ICL Examples, and 4) Input Elderly Narratives.

3.1.2 Context within Prompt

The Context includes the roles assigned to the LLM in the prompts, the tasks, and the background knowledge. In this segment, we incorporated detailed explanations of entity types as foundational knowledge, enhancing the model's comprehension of diverse entity classifications, thereby augmenting the recognition accuracy. Figure 4 shows an example of background in prompt.

3.1.3 Instruction within Prompt

To ensure precision in entity recognition, the method iterates over all entity labels for each input sentence. This approach effectively transforms an N-way NER task into N individual 1-way NER tasks. The rationale behind this transformation is that LLMs, such as GPT-3.5, tend to produce outputs that diverge from the desired format when dealing with descriptions for all entity types simultaneously. This phenomenon will be further detailed in the results section. Therefore, for each input sentence, we generate N distinct prompts, each corresponding to a specific entity type.

We employs an intermediate representation sequence termed auto Chain-of-Thoughts (CoT) (Wei et al., 2022). CoT provides background, guidance, and clarity to the text generated by LLMs. Additionally, it offers insight into the evaluation process and results. An example of CoT's utilization in the instruction prompt can be observed in Figure 4.

Marking Steps:

1. Read the input sentence carefully and read the entity explain of protagonist and Health carefully.

2. Identify the Protagonist and Health entities in the input sentence.

3. If there is no desired entity in the input sentence, just output the input sentence.

4. If there is any Health entities, mark these entities using [] like Example Marked sentence.

5. If there is any Protagonist entities, mark these entities using { } like Example Marked sentence.

Figure 5: Instruction within Prompt

3.1.4 Examples within Prompt

In this segment, we elucidate the examples incorporated within the prompts presented to the LLM. The OALS dataset is partitioned into 15% for training and 85% for validation. Each time a prompt is constructed, $k (k = 4$ in this paper) elderly narratives containing the corresponding entity types and their associated labels are randomly selected from the training set to serve as demonstrations. We embed the protagonist elements as context within the examples presented in the prompt.

3.1.5 Input Elderly Narratives within Prompt

In this segment, the input elderly narratives is appended to the end of the prompt and then fed into the LLM. We anticipate that the LLM will produce an output sequence based on the format defined in Instruction and ICL Examples. As illustrated in the "marked text" segment of Figure 3. For each entity category (e.g., "education"), the LLM will produce a marked text where the protagonist is denoted with '{}' and 'education' is enclosed within '[]'.

3.1.6 Parse marked output

Upon obtaining the text sequences generated by the GPT-3.5, we construct a parser to scan the marked sections for each entity category. These NER results are then consolidated into BIO-format annotations.

3.2 Elderly Portrait Event Extract

Upon completing named entity recognition for elderly narratives, we extract events corresponding to these entities for the construction of elderly portraits. We propose an event extraction

...
Input:
entity:[Example entity]$_k$
entity type:[Example entity type]$_k$
story:[Example elderly narratives]$_k$
Output:
ei:[Example ei]$_k$
pc:[Example pc]$_k$
tm:[Example tm]$_k$
loc:[Example loc]$_k$
em:[Example em]$_k$
Input:
entity:[Input entity]
entity type:[Input entity type]
story:[Input elderly narratives]
Output:

Figure 6: Prompt for Event Extract

method based on ICL, Elderly Portrait Event Extract (EPEE). Utilizing both the entity type and entity name as context, we construct prompts to be input into LLMs, ultimately extracting five types of event elements. Figure 6 shows the input and output formats part defined in the prompt, and the complete prompt can be found in the appendix A.

4 Experiments

We chose GPT-3.5 as the Large Language Model (LLM), accessed via the API (gpt-3.5-turbo). Regarding parameter configurations, to ensure the stability of the experimental results as much as possible, the temperature is set to 0, with all other parameters left at their default settings.

4.1 Baseline

4.1.1 Few shot NER

SpanProto (Wang et al., 2022a): The SpanProto approach employs a two-phase method to address the low-sample entity extraction challenge. Initially, it involves span extraction, followed by mention categorization to better adapt to new entity categories. Additionally, it enhances model performance by introducing a boundary-based loss, specifically addressing false positives generated by the span extractor.

ESD (Wang et al., 2022b): An Enhanced Span-Decomposition (ESD) technique tailored for Few-Shot Sequence Labeling (FSSL). ESD formalizes the low-sample sequence tagging as a span-level matching problem between test queries and support instances. This approach decomposes the span matching challenge into a series of span-level processes, primarily encompassing enhanced span representation, category prototype aggregation, and span conflict resolution.

CONTaiNER (Das et al., 2021): A contrastive learning method based on low samples. This approach optimizes a generalized objective, which distinguishes the intrinsic representation distribution of entities based on their Gaussian distribution embeddings, effectively mitigating overfitting issues arising from the training domain.

ProtoNet (Snell et al., 2017): This method leverages prototype networks to address the challenge of low-sample classification. Prototype networks learn a metric space wherein classification can be executed by computing distances to prototype representations of each category.

NNShot (Yang and Katiyar, 2020): The NNShot approach utilizes a supervised NER model trained on the source domain as a feature extractor. By employing a nearest-neighbor classifier, it achieves more efficient performance across multiple test domains. This method can capture label dependencies between entity labels without the necessity for Conditional Random Field (CRF) training .

4.1.2 Other Prompt Construction

In Section 3.1, we enhanced the entity recognition efficacy by incorporating specific illustrate of elderly portrait entities and protagonist elements as context within the prompt. In the subsequent experiments, we omitted these components to quantitatively assess the tangible impact of these elements in the prompt on recognition performance.

EPNER without protagonist and illustrate : Exclude the specific illustrate related to elderly portrait entities and the protagonist elements within the prompt. Consequently, the LLM no longer annotates the protagonist entity.

EPNER without protagonist : Exclude the protagonist elements from the prompt. Again, the LLM does not annotate the protagonist entity in this case.

MUti EPNER : We devised a prompt instructing the LLM to annotate all elderly portrait entity types simultaneously, using a tagging strategy like [Entity#EntityType].

When comparing various Prompt Construction, we also consider an additional metric—Marking

Failure Rate. This consideration arises from our observation during experiments that, upon feeding the constructed prompt to instruct the LLM for a text labeling task, the LLM does not consistently produce outputs that adhere to the required format. Such deviations encompass the emergence of fictitious marking formats, alterations to the original content, and the generation of irrelevant content, among others. We categorize these non-compliant output texts as $T_f = \{tf_1, tf_2, ..., tf_m\}$. Assuming the entire set of output texts is represented by $Text_{output} = \{text_{output_1}, text_{output_2}, ..., text_{output_n}\}$, the Marking Failure Rate is computed using the formula $Score_f = m/n$, where m denotes the number of texts in T_f and n signifies the number of texts in $Text_{output}$.

4.2 Result for NER

Table 2 presents a compare of our method with other Few-Shot NER methods. The results indicate that our approach (EPNER) achieved the highest performance in terms of the F1 score(35.8%). Although the SpanProto and CONTaiNER methods respectively secured optimal results in Precision(44.9%) and Recall(54.0%), their corresponding low scores in Recall(21.3%) and Precision(0.1%) respectively rendered their overall performance inferior to EPNER.

Table 2: Result of different few shot NER

Method	Precision(%)	Recall(%)	F1(%)
SpanProto	44.9	21.3	28.9
ESD	2.7	35.0	5.1
CONTaiNER	0.1	54.0	0.2
Protobert	17.9	37.4	24.3
NNShot	5.8	31.3	9.8
EPNER	38.9	34.7	35.8

Table 3 illustrates the impact on our method after omitting the specific explanations of elderly portrait elements and protagonist features from the event elements within the prompt. The table reveals that upon entirely removing both the protagonist and explanation from the prompt, the Marking Failure Rate reaches its lowest at 1.0%, but correspondingly, the F1 value also plunges to its lowest at 14.6%. The table also presents the results of experiments recognizing all entity types simultaneously. The F1 score for this approach, at 27.4%, is inferior compared to recognizing different entity types separately, and notably, the Marking Failure Rate escalates significantly to 32.1%.

Table 4 delineates the recognition performance of our method across diverse entity types. As discerned from the table, with the exception of the ’Occupation’ entity, the F1 score for all entity categories exceed 26.3%. The recognition efficacy for the ’Location’ entity is the most commendable, achieving a score of 46.0%. Conversely, the performance for the ’Occupation’ entity is the least impressive, registering a mere 9.4%.

Question 1: Does the event type correspond with the feature entity? A match is scored as 1, and a mismatch as 0. In cases of mismatch, subsequent questions are bypassed.

Question 2: Does the event trigger word align with the entity feature? A match is scored as 1, and a mismatch as 0.

Question 3: Does the temporal occurrence of the event correspond with the entity feature? A match is scored as 1, and a mismatch as 0.

Question 4: Does the event location align with the entity feature? A match is scored as 1, and a mismatch as 0.

Question 5: Does the protagonist of the event correspond with the entity feature? A match is scored as 1, and a mismatch as 0.

Question 6: Is the event description associated with physical features? Set the 7-level correlation level of 1-7, where 1 is completely unrelated and 7 is completely related.

Figure 7: Question for evaluating EPEE

4.3 Manual Evaluation for EPEE

To assess the efficacy of EPEE, manual evaluation is conducted on feature-associated events extracted from the OALS dataset. The primary aspects evaluated in Figure 7.

Table 5 shows the evaluation results of event correlation after event extraction for different types of elderly portrait entities. The events extracted using EPEE method demonstrate high relevance to the corresponding portrait entities across various elements (the average scores for Q1 to Q5 are all above 0.8, and for Q6, they are consistently above 4.5). Notably, the events related to the "Identity" entity exhibit the highest degree of extraction relevance.

5 Discussion

5.1 Overall Recognition Performance

As shown in the Table 2, the F1 scores of traditional few-shot entity recognition methods are somewhat underwhelming when applied to elderly narrative texts. This subpar performance can be attributed to a confluence of factors inherent to these texts:

Table 3: Result of different prompt

Method	Precision(%)	Recall(%)	F1(%)	Marking failure rate(%)
EPNER without protagonist and explanation	8.5	29.0	14.6	1.0
EPNER without protagonist	14.8	31.8	19.6	1.3
Muti-EPNER	38.5	24.1	27.4	32.1
EPNER	38.9	34.7	35.8	2.8

Table 4: Result of different entities

Entity type	Precision(%)	Recall(%)	F1(%)
Location	51.6	41.5	46.0
Health	26.8	38.1	31.5
Interest	28.5	31.0	29.7
Occupation	6.1	20.0	9.4
Identity	31.4	22.6	26.3
Education	21.4	41.4	28.3

Table 5: Result of manual evaluation for elderly portrait

Entity type	Q1	Q2	Q3	Q4	Q5	Q6
Location	0.957	0.898	0.893	0.954	0.941	4.575
Health	0.886	0.910	0.919	0.897	0.902	4.691
Interest	0.902	0.855	0.945	0.905	0.910	4.667
Occupation	0.867	0.952	0.952	0.889	0.900	4.767
Identity	0.931	0.947	0.857	0.941	0.958	4.806
Education	0.859	0.862	0.882	0.923	0.885	5.721

a paucity of samples, an imbalanced distribution of entity types, and the prevalence of noise in the text. These combined challenges exacerbate the difficulties for traditional methods.

Incorporating context into entity recognition prompts led to significant accuracy improvements. This highlights the efficacy of LLMs, especially with appropriate prompts, in tasks with limited samples and complex contexts.

5.2 Deliberations on the Context within the Prompt

We observed a critical relationship between the amount of context included and entity recognition efficiency. As shown in Table 3, increased knowledge can improve accuracy. However, overloading with knowledge can adversely affect the large language model's output, leading to deviations from the expected format.

5.3 Recognition Strategies for Different Labels

Generally, our approach demonstrated strong performance for most entity categories, with F1 scores exceeding 28%, as shown in the Table 2. However, accurately identifying "Occupation" and "Identity" labels proved challenging.

To address this, we considered implementing a two-stage or multi-stage entity recognition workflow, building on the LLM's initial results. This iterative process aims to continuously refine recognition. We also considered adding contrasting examples for ambiguous entity categories in the prompts, to aid the model in differentiating these entities without significantly increasing annotation errors.

6 Related Work

6.1 Few-shot NER

Named Entity Recognition (NER) is a task that identifies key information within text and categorizes it into a set of predefined classes. A common approach to address NER is to treat it as a sequence tagging task (Hammerton, 2003). Few-shot NER requires recognizing entities with the support of only very few labeled instances (Hofer et al., 2018; Fritzler et al., 2019). Due to limited information contained in labeled instances, methods for few-shot NER mainly resort to a rich-resource source domain to help train models, resulting in transfer-learning and meta-learning frameworks.

Contemporary meta-learning techniques predominantly cater to few-shot learning scenarios and can be broadly categorized into three paradigms: Metric-based, Optimization-based, and Memory-based approaches (Li et al., 2020). Metric-based techniques predicate label predictions on similarity measures between samples, such as Euclidean distances or cosine similarities (Vinyals et al., 2016). Optimization-based strategies endeavor to expedite learning through explicitly learned update rules or weight initializations (Ravi and Larochelle, 2016). Memory-based methods, conversely, instantiate memory or storage units, enabling the model to retain and leverage previously observed experiences, thereby fostering rapid learning and generalization (Santoro et al., 2016). Existing Few-Shot NER techniques typically emphasize metric-based learning, deriving entity recognition by discerning representations within semantic spaces. For instance, ProtoNet (Snell et al., 2017) employs prototype networks to discern prototype representations for each entity category, while NNShot (Yang and Katiyar, 2020) directly utilizes word embeddings as representations, subsequently employing nearest neighbor classification for inference.

Lately, prompt learning has witnessed substantial advancements in few-shot tasks by designing bespoke templates to guide models towards pertinent information, emerging as a novel paradigm in natural language processing. As such, a slew of methods integrating prompt learning into Few-Shot NER tasks have been proposed (Cui et al., 2021)(Liu et al., 2022).

6.2 In-Context Learning

Large Language Models (LLMs) (Brown et al., 2020; Rae et al., 2021; Smith et al., 2022; Hoffmann et al., 2022; Chowdhery et al., 2022) have achieved significant performance improvements across various Natural Language Processing tasks (Hegselmann et al., 2023; Vilar et al., 2022; Perez et al., 2021; Swanson et al., 2021; Wei et al., 2021). Strategies to leverage LLMs for downstream tasks can be classified into two categories: fine-tuning and in-context learning. Fine-tuning strategies use a pre-trained model as initialization and run additional epochs on downstream supervised data (Raffel et al., 2020; Gururangan et al., 2018; Roberts et al., 2020; Guu et al., 2020). In contrast to the fine-tuning strategy, In-Context Learning (ICL) prompts LLMs to generate text under few-shot demonstrations. Radford was the first to use prompts containing demonstrations to reformulate downstream tasks (Radford et al., 2019). Many studies showed that better prompts and demonstrations can enhance the performance of in-context learning (Perez et al., 2021; Lu et al., 2021; Rubin et al., 2021; Min et al., 2022; Liu et al., 2021). There has been research applying the in-context learning to applications like entity recognition (Wang et al., 2023), event extraction (Gao et al., 2023), and information extraction (Wei et al., 2023).

7 Limitation

7.1 Limited Dataset Volume

Research pertaining to elderly narratives is still in its nascent stages, and there isn't a substantial dataset dedicated to these narratives available. The efficacy of both our proposed elderly portrait automatic construction framework and the EPNER method must be further validated through practical applications, given the current dataset constraints.

7.2 Inconsistent Recognition Performance

Owing to the inherent probabilistic of general language models, the entity recognition process for elderly portraits doesn't always proceed seamlessly. We've endeavored to mitigate the likelihood of producing erroneously formatted outputs by adjusting the temperature parameter and meticulously optimizing the prompts. However, instances of such discrepancies still manifest, which invariably impacts the reproducibility of our results.

7.3 Optimization for Cross-Domain Usage

Our EPNER method can be transposed to NER tasks in diverse domains. Nevertheless, when applied in practice, there's a requisite for manual prompt adjustments. For LLMs, even minute alterations at the word level can substantially influence the model's output. Crafting the optimal prompt tailored for varied domains remains a pressing challenge that warrants further exploration.

8 Conclusion

In this paper, we propose an event-based elderly portrait and developed a framework that Large Language Models (LLM) to autonomously construct portraits from elderly narratives. Our findings suggest that incorporating specific contextual information within prompts can substantially enhance the recognition performance when deploying LLMs for Named Entity Recognition (NER) tasks. Stemming from this insight, we proposed EPNER, a in-context learning based elderly portrait named entity recognition method, tailored to address the challenge of extracting elderly portrait features from their narratives. Experimental evaluations on the OALS dataset revealed that our approach outperforms baseline methods. Additionally, manual evaluations corroborated the efficacy of the event-based elderly portraits that our framework autonomously generates from elderly narratives.

References

Ning An, Fang Gui, Liuqi Jin, Hong Ming, and Jiaoyun Yang. 2023. Toward better understanding older adults: A biography brief timeline extraction approach. *International Journal of Human–Computer Interaction*, 39(5):1084–1095.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using bart. *arXiv preprint arXiv:2106.01760*.

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J Passonneau, and Rui Zhang. 2021. Container: Few-shot named entity recognition via contrastive learning. *arXiv preprint arXiv:2109.07589*.

Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 993–1000.

Jun Gao, Huan Zhao, Changlong Yu, and Ruifeng Xu. 2023. Exploring the feasibility of chatgpt for event extraction. *arXiv preprint arXiv:2303.03836*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. *arXiv preprint arXiv:1803.02324*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

James Hammerton. 2003. Named entity recognition with long short-term memory. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 172–175.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pages 5549–5581. PMLR.

Maximilian Hofer, Andrey Kormilitzin, Paul Goldberg, and Alejo J. Nevado-Holgado. 2018. Few-shot learning for named entity recognition in medical text. *CoRR*, abs/1811.05468.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Jing Li, Billy Chiu, Shanshan Feng, and Hao Wang. 2020. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(9):4245–4256.

Andy T Liu, Wei Xiao, Henghui Zhu, Dejiao Zhang, Shang-Wen Li, and Andrew Arnold. 2022. Qaner: Prompting question answering models for few-shot named entity recognition. *arXiv preprint arXiv:2203.01543*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Hans Maes. 2015. What is a portrait? *British Journal of Aesthetics*, 55(3):303–322.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

OpenAI. 2023. Gpt-4 technical report.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *Advances in neural information processing systems*, 34:11054–11070.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Sachin Ravi and Hugo Larochelle. 2016. Optimization as a model for few-shot learning. In *International conference on learning representations*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2021. Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. 2016. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850. PMLR.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. 2022. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*.

Jake Snell, Kevin Swersky, and Richard S Zemel. 2017. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*.

Dimitris Spiliotopoulos, Dionisis Margaris, and Costas Vassilakis. 2020. Data-assisted persona construction using social media data. *Big Data and Cognitive Computing*, 4(3):21.

Ben Swanson, Kory Mathewson, Ben Pietrzak, Sherol Chen, and Monica Dinalescu. 2021. Story centaur: Large language model few shot learning as a creative writing tool. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 244–256.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2022. Prompting palm for translation: Assessing strategies and performance. *arXiv preprint arXiv:2211.09102*.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. *Advances in neural information processing systems*, 29.

Jianing Wang, Chengyu Wang, Chuanqi Tan, Minghui Qiu, Songfang Huang, Jun Huang, and Ming Gao. 2022a. Spanproto: A two-stage span-based prototypical network for few-shot named entity recognition. *CoRR*, abs/2210.09049.

Peiyi Wang, Runxin Xu, Tianyu Liu, Qingyu Zhou, Yunbo Cao, Baobao Chang, and Zhifang Sui. 2022b. An enhanced span-based decomposition method for few-shot sequence labeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5012–5024, Seattle, United States. Association for Computational Linguistics.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models. *arXiv preprint arXiv:2304.10428*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, et al. 2023. Zero-shot information extraction via chatting with chatgpt. *arXiv preprint arXiv:2302.10205*.

Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *arXiv preprint arXiv:2010.02405*.

A Prompt for Event Extract

You are an excellent linguist.

You should extract event elements in the input sentence, the same format as in the example is required.

Read and understand these instructions carefully.

You can not omit any original information during annotation.

Event Elements Explain:

pc: The person in the event.

tm: Time of the event.

loc: The location of the event.

ei: The action of the event.

em: Summary of the event

Extract Steps:

1. Read the input sentence carefully and read the event elements explain carefully.

2. Identify event elements in the input sentence.

3. If there is no event element in the input sentence, Just output None after the corresponding element class .

For example:

. . .

Input:

entity:[Example entity]$_k$

entity type:[Example entity type]$_k$

story:[Example elderly narratives]$_k$

Output:

ei:[Example ei]$_k$

pc:[Example pc]$_k$

tm:[Example tm]$_k$

loc:[Example loc]$_k$

em:[Example em]$_k$

Input:

entity:[Input entity]

entity type:[Input entity type]

story:[Input elderly narratives]

Output:

Figure 8: Prompt for Event Extract