Intervention extraction in preclinical animal studies of Alzheimer's Disease: Enhancing regex performance with language model-based filtering

Yiyuan Pu[1], Kaitlyn Hair[3], Daniel Beck[2], Mike Conway[1], Malcolm Macleod[3], Karin Verspoor[1,2]

[1] School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia
[2] School of Computing Technologies, RMIT University, Melbourne, Australia
[3] Centre for Clinical Brain Sciences, The University of Edinburgh, Edinburgh, U.K

Abstract

We explore different information extraction tools for annotation of interventions to support automated systematic reviews of preclinical AD animal studies. We compare two PICO (Population, Intervention, Comparison, and Outcome) extraction tools and two prompting-based learning strategies based on Large Language Models (LLMs). Motivated by the high recall of a dictionary-based approach, we define a two-stage method, removing false positives obtained from regexes with a pre-trained LM. With ChatGPT-based filtering using three-shot prompting, our approach reduces almost two-thirds of False Positives compared to the dictionary approach alone, while outperforming knowledge-free instructional prompting.

1 Introduction and Related Work

Biomedical information extraction is the task of automatically extracting entities, relations, and events from biomedical literature (Hobbs, 2002; Liu et al., 2016). This information is in turn relevant to writing of systematic reviews, which support evidence-based decision making by identifying, integrating, and assimilating relevant articles on a given clinical question (Methley et al., 2014). A standard framework used for defining review questions is PICO, standing for Population (or Patient), Intervention (or Exposure), Comparison, and Outcome (Cooke et al., 2012).

We examine information extraction in Alzheimer's Disease (AD), which has affected more than 55 million people around the world[1]. We focus on detecting the PICO dimension of *Intervention* in the AD literature, where interventions are typically drugs. This task is sometimes referred to as *intervention extraction*. It suffers from having low precision compared to extraction of other PICO elements (Hair et al., 2023a). More precise extraction of interventions will support more effective systematic reviewing, and can help to prioritize drugs for clinical trials in literature-based discovery (Pu et al., 2023).

Standard methods for intervention extraction include dictionary-based approaches (Hair et al., 2023b) and machine learning models (Wang et al., 2021; Wei et al., 2024). The recent advent of generative Large Language Models (LLMs) and the prompting-based paradigm for information extraction (Liu et al., 2023) raise questions of how to better leverage them in this task and whether they outperform previous methods. This is particularly interesting in domain-specific scenarios such as AD, where limited data is available for training models (Wang et al., 2023). We address these questions, with two main contributions: 1) we show that while generative LLMs improve intervention extraction precision, they suffer from low recall compared to dictionary-based methods, and 2) we propose a two-stage architecture combining both dictionaries and LLMs that better balances precision and recall and reaches a new state-of-the-art on the AD dataset.

2 Methods

2.1 Data

We used a manually-curated dataset containing preclinical animal studies in the context of AD (Hair et al., 2023b). This dataset consists of documents comprising title, abstract, and keyword fields for 100 studies. The dataset was created in two steps: 1) a set of regular expression (regex) patterns corresponding to a dictionary of interventions was applied to annotate intervention entities, and 2) a human annotator labeled each tagged entity as "intervention" or "not an intervention". Figure 1 shows an example of an annotated document, the extracted entities, and the human judgment label for each. The AD dataset may not be perfect since

[1]https://www.alzint.org/about/dementia-facts-figures/dementia-statistics/

Proceedings of the 23rd Workshop on Biomedical Language Processing, pages 486–492
August 16, 2024.

	Training set	Test set
Document count	5	95
#Intervention	6	67
#Not an intervention	14	288

Table 1: AD dataset statistics for training and testing.

human annotations were a subset of regex annotations. Intervention entities not being captured by the regex dictionary were out of the scope for the human annotator. However, we used human annotations as a gold standard for this study.

Article id: PMID 31190768
Document: ["... Icariin (ICA) as one of the active ingredients of Chinese herbal medicine has the immunomodulating function. This study aimed to investigate the immunotherapeutic potential of ICA on AD... Then the ethological and biochemical experiments such as Morris water maze assay Aβ ELISA blood T cell flow cytometry and plasma and brain cytokines array were conducted to evaluate the effects of ICA administration. ..."]
Matched spans: [[156,163], [527,532]]
Text spans: ["Icariin", "water"]
Human labels: ["intervention", "not an intervention"]

Figure 1: An example of an annotated document in the dataset from Hair et al. (2023b). The "document" field contains the title, abstract, and keywords for each paper. Only part of the abstract is shown for brevity.

We randomly split the dataset into a training set (5 documents) and a test set (95 documents) (Table 1). The training documents were used for few-shot learning in prompt-based methods. All results are reported on the test set.

2.2 Baselines

We adapted three biomedical entity extraction tools for intervention extraction to use as baselines, detailed in Table 2. The regex-based method of (Hair et al., 2023b) utilizes a customized dictionary based on regular expressions for preclinical AD animal studies. Each publication was tagged with animal models, outcomes, interventions, species, and sexes; here we considered only entities tagged as interventions. The intervention dictionary had a list of 12,447 compounds compiled from DrugBank[2] and Alzforum[3]. Synonyms, alternate spellings, and punctuation differences were captured in regexes (Hair et al., 2023b). This method was used to create the dataset employed in our experiments, resulting in maximum recall by design.

Wang et al. (2021) constructed a PICO extraction workflow based on Bidirectional Encoder Representations (BERT; Devlin et al. (2019)) for general preclinical animal studies (not specific to AD). This method had two entity categories that relate to interventions: Intervention and Comparator. Intervention was defined as interventions that reflect clinical practice, while Comparator was defined as a control group, such as no treatment, vehicle/-placebo, sham treatment, or another intervention. We treated entities tagged as either Intervention or Comparator entity types as interventions.

Finally, we also used the latest version of PubTator 3.0 (Wei et al., 2024) as an additional baseline, due to its widespread usage in biomedical information extraction. This tool extracted proteins, genetic variants, diseases, and chemicals with a recently developed named entity recognition (NER) model called AIONER (Luo et al., 2023). We treated Pubtator-identified Chemical entities as Interventions. For this entity type, training was based on the NLM-Chem corpus (Islamaj et al., 2021), with $\sim$5000 unique drug/chemical name annotations in 150 PubMed full-text chemical literature. The PubTator API[4] was used to conduct raw processing of input texts for entity extraction.

2.3 Prompt-based methods

Since we had limited labeled data, we prioritized prompt-based models over training machine learning or deep learning models. We followed the framework of (Liu et al., 2023) to design our prompt-based models. We considered four aspects of prompt-based learning for intervention extraction: pre-trained language models (PLMs), prompting templates, answer space, and prompting parameters.

Pre-trained LMs We selected ChatGPT[5] and GPT-4 (OpenAI, 2023) as the PLMs.[6]

Prompting templates We adapted prompting templates previously used for zero-shot gene extraction in biomedical literature (Törnkvist, 2024) to our task for both zero-shot and few-shot learning.

[2]https://go.drugbank.com/drugs

[3]https://www.alzforum.org/therapeutics

[4]https://www.ncbi.nlm.nih.gov/research/pubtator/api.html

[5]https://platform.openai.com/docs/models/gpt-3-5-turbo

[6]In an effort to employ open-source LMs, we also considered OLMo (Groeneveld et al., 2024). However, we were not able to obtain meaningful answers from this LM.

Entity extraction tool	Scope of entity types	Entity types used
Regex-based method (Hair et al., 2023b)	Animal model, Outcome measure, Intervention, Species, Sex	Intervention
BERT-based method (Wang et al., 2021)	Intervention, Comparator, Outcome, Species, Strain, Induction	Intervention, Comparator
PubTator 3.0 (Wei et al., 2024)	Gene/Protein, Variant, Disease, Chemical, Species, and Cell Line	Chemical
ChatGPT (OpenAI, 2023)	-	Intervention
GPT-4 (OpenAI, 2023)	-	Intervention

Table 2: Scope and use of entity types for this study from entity extraction tools

Our templates[7] are described in Appendix B.

Answer space We used text spans from the documents that were recognized as interventions.

Prompting parameters For both models, we set temperature to be 0.7, max_tokens as 50, and top_p as 1. The "temperature" parameter controls randomness, which ranges between 0 to 2.

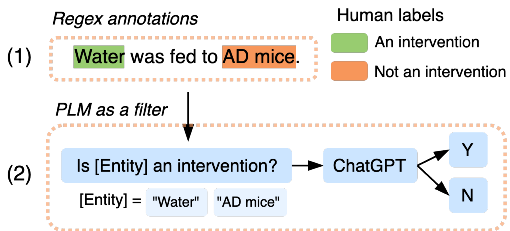

Figure 2: An example of the two-stage filtering method that we proposed. (1) A regex-based method annotated potential interventions in each document. Each potential intervention had a corresponding human label, indicating whether it was an intervention in context. (2) Each potential intervention and its context were inputs for a PLM. Figure is simplified due to space limitations.

2.4 Two-stage filtering method

The approach we proposed (Figure 2) was motivated by the maximum recall provided by the regex patterns used to create the dataset. Instead of having a PLM doing the full work of extracting interventions, we proposed using it to *filter the false positives obtained from regexes*. Precision errors arising from regex-based methods were mainly due to a lack of context: any entity that matched a regex would be recognized as an intervention. We hypothesized that a PLM can contextualize entity context and filter them out if appropriate, without undermining recall.

We experimented with both zero-shot and few-shot approaches (with examples sampled from the training set), using ChatGPT as the PLM. We tailored our prompts to frame the task as filtering (described in Appendix C). All other parameters were the same as described in Section 2.3.

3 Results and Discussion

Table 3 summarises our results, reporting precision, recall, and F1 scores for all methods. As expected, the regex-based method resulted in perfect recall (since it was employed to develop the dataset in the first place) but low precision. Both BERT-based and PubTator 3.0 approaches did not perform well, likely due to domain differences. The prompt-based methods resulted in slightly better precision, at the price of a large decrease in recall.

Our zero-shot filtering approach outperformed the baselines in F1 score, with only a small decrease in recall. It filtered 30% of the false positives (FPs) of the regex-based method. A three-shot variant, adding three true positive (TP) examples to the prompt, gave even better precision and F1 score, filtering almost two-thirds of all FPs.

3.1 Further analysis on few-shot prompting

We performed additional experiments analyzing the influence of adding positive examples to the prompt in our two-stage method. The regex-only baseline resulted in 67 TPs and 288 FPs: an ideal filtering layer should remove all FPs while keeping the original TPs.

Table 4 shows detailed results using two metrics: the total reduction in FPs and the total reduction in TPs (the latter framed as "TP price", since this should ideally be zero). In general, the higher the FP reduction, the higher the TP price. However, we did not see any particular trends when increasing

[7] The prompting templates (Appendix B) used for baselines were different from the templates (Appendix C) in the two-stage filtering method. For baselines, the prompting templates were adapted from Törnkvist (2024). For the two-stage filtering method, the prompting templates were created on our own.

Intervention extraction method	TP	FP	FN	Precision	Recall	F1 score
Baselines						
Regex-based	**67**	288	**0**	0.19	**1.00**	0.32
BERT-based	22	232	45	0.09	0.33	0.14
PubTator 3.0	43	369	24	0.10	0.64	0.18
ChatGPT (0-shot)	28	97	39	0.22	0.42	0.29
ChatGPT (3-shot)	24	**61**	43	0.28	0.36	0.32
GPT-4 (0-shot)	27	108	40	0.20	0.40	0.27
GPT-4 (3-shot)	30	94	37	0.24	0.45	0.31
Our approach						
Regex+ChatGPT (0-shot)	64	203	3	0.24	0.96	0.38
Regex+ChatGPT (3-shot)	58	107	9	**0.35**	0.87	**0.50**

Table 3: Results for all intervention extraction. The first three columns detail true positives (TPs), false positives (FPs) and false negatives (FNs), while the last three columns report our evaluation metrics.

the number of examples, except for an outlier result for 1-shot prompting. Our 3-shot results provided the best balance but more work is required to understand if further increasing the number of examples can result in better performance.

#Examples	TP	FP	TP price	FP reduction
Baseline	67	288	0 (0%)	0 (0%)
0	64	203	3 (4%)	85 (30%)
1	45	90	22 (33%)	198 (69%)
2	63	181	4 (5%)	107 (37%)
3	58	107	9 (13%)	181 (62%)
4	63	177	4 (5%)	111 (38%)
5	61	186	6 (8%)	102 (35%)

Table 4: Detailed results varying the number of examples using our two-stage approach. All examples were "positive" labels (entities labeled as interventions by the human annotator), sampled from the training set.

We also tried using negative examples for few-shot learning. However, this did not improve the performance compared to using positive examples only. We report detailed results in Appendix A.

3.2 Motivating case study

We discuss case studies for our motivation for employing the two-stage filtering method. We analyzed the False Positives (FPs) of the regex-based method. As this method annotated every text span matching the intervention dictionary indiscriminately, the 288 FPs came from context recognition errors, i.e. where an intervention term did not describe a relevant intervention in the context of a document. For instance, the potential intervention entity "quercetin" in PMID:36840284 (Table 5) was part of the molecular modeling results of a compound rather than a drug whose effects were directly studied.

PMID	36840284
Evaluation	False Positive
Context	"Molecular modeling results revealed that the compound's ellagic acid, epicatechin, catechin, kaempferol, quercetin, and apigenin have the potential to act as a dual inhibitor of acetylcholinesterase (AChE) and COX-2 and can be responsible for the improvement of both cholinergic and inflammatory conditions."
PMID	30618732
Evaluation	True Positive
Context	"This study aimed to evaluate the neuroprotective effect of quercetin against the detrimental effects of LPS such as neuroinflammation-mediated neurodegeneration and synaptic/memory dysfunction in adult mice."

Table 5: An example of the same entity string labeled as both an intervention and not an intervention in distinct contexts.

One may argue that removing "quercetin" from the regex dictionary would reduce FPs. However, the entity "quercetin" was also used as an intervention in other contexts. As shown in Table 5, PMID:30618732 assessed the effects of "quercetin" as an intervention for treating adult mice with neurodegenerative diseases. Therefore, an ideal method must contextually differentiate usages of the putative entity mentions.

3.3 PLM response outliers

A generative PLM may produce model responses out of the target answer space, requiring further processing. In the Regex+ChatGPT (0-shot) scenario, the model responded with "therapeutic" (cf. "intervention"/"not an intervention") for a potential intervention entity string "therapeutic" and a given context of PMID:25061594. In the Regex+ChatGPT (3-shot) scenario, the model responded with a copy of the prompting template for a potential intervention “potassium" for PMID:30548427.

For these two outliers, we reverted to the output of a RegEx phase, i.e. with the label “intervention”.

4 Conclusion

In this work, we proposed a two-stage approach for intervention extraction that combined a regex-based method with a filtering step done by prompting a generative LLM. This approach outperformed strong baselines, including standalone use of LLMs. Effectively, we show that LLMs can augment regex/dictionary-based methods by removing context recognition errors.

Future work involves extending our approach to all PICO entities, beyond just interventions. This will help automate important tasks in the literature review for AD, such as collecting data for systematic reviews, and support creating more precise knowledge graphs for literature-based discovery. The same approach could also be adapted with specific resources and be applied to other datasets and domains, such as clinical trials (Nye et al., 2018). Finally, different strategies to employ LLMs in the filtering step could be investigated, such as fine-tuning.

References

Alison Cooke, Debbie Smith, and Andrew Booth. 2012. Beyond PICO: The SPIDER Tool for Qualitative Evidence Synthesis. *Qualitative Health Research*, 22(10):1435–1443.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. OLMo: Accelerating the Science of Language Models. *Preprint*, arXiv:2402.00838.

Kaitlyn Hair, Emma Wilson, Olena Maksym, Malcolm Robert Macleod, and Emily Sena. 2023a. A Systematic Online Living Evidence Summary of experimental Alzheimer’s disease research. *MetaArXiv*.

Kaitlyn Hair, Emma Wilson, Charis Wong, Anthony Tsang, Malcolm R Macleod, and Alexandra Bannach-Brown. 2023b. Systematic online living evidence summaries: emerging tools to accelerate evidence synthesis. *Clinical Science (London, England : 1979)*, 137:773 – 784.

Jerry R. Hobbs. 2002. Information extraction from biomedical text. *Journal of Biomedical Informatics*, 35(4):260–264. Sublanguage - Zellig Harris Memorial.

Rezarta Islamaj, Robert Leaman, Sun Kim, Dongseop Kwon, Chih-Hsuan Wei, Donald C. Comeau, Yifan Peng, David Cissel, Cathleen Coss, Carol Fisher, Rob Guzman, Preeti Gokal Kochar, Stella Koppel, Dorothy Trinh, Keiko Sekiya, Janice Ward, Deborah Whitman, Susan Schmidt, and Zhiyong Lu. 2021. NLM-Chem, a new resource for chemical entity recognition in PubMed full text literature. *Scientific Data*, 8.

Feifan Liu, Jinying Chen, Abhyuday N. Jagannatha, and Hong Yu. 2016. Learning for biomedical information extraction: Methodological review of recent advances. *ArXiv*, abs/1606.07993.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.*, 55(9).

Ling Luo, Chih-Hsuan Wei, Po-Ting Lai, Robert Leaman, Qingyu Chen, and Zhiyong Lu. 2023. AIONER: all-in-one scheme-based biomedical named entity recognition using deep learning. *Bioinformatics*, 39(5):btad310.

Abigail M Methley, Stephen Campbell, Carolyn Chew-Graham, Rosalind McNally, and Sudeh Cheraghi-Sohi. 2014. PICO, PICOS and SPIDER: a comparison study of specificity and sensitivity in three search tools for qualitative systematic reviews. *BMC health services research*, 14(1):1–10.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain Marshall, Ani Nenkova, and Byron Wallace. 2018. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Melbourne, Australia. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 Technical Report.

Yiyuan Pu, Daniel Beck, and Karin Verspoor. 2023. Graph embedding-based link prediction for literature-based discovery in Alzheimer’s Disease. *Journal of Biomedical Informatics*, 145:104464.

Betty Törnkvist. 2024. Named Entity Recognition for Detecting Trends in Biomedical Literature. Master's thesis, Umeå University, Department of Computing Science.

Benyou Wang, Qianqian Xie, Jiahuan Pei, Zhihong Chen, Prayag Tiwari, Zhao Li, and Jie Fu. 2023. Pre-trained Language Models in Biomedical Domain: A Systematic Survey. *ACM Comput. Surv.*, 56(3).

Qianying Wang, Jing Liao, Mirella Lapata, and Malcolm Macleod. 2021. PICO entity extraction for preclinical animal literature. *Systematic Reviews*, 11.

Chih-Hsuan Wei, Alexis Allot, Po-Ting Lai, Robert Leaman, Shubo Tian, Ling Luo, Qiao Jin, Zhizheng Wang, Qingyu Chen, and Zhiyong Lu. 2024. PubTator 3.0: an AI-powered literature resource for unlocking biomedical knowledge. *Nucleic Acids Research*.

A Prompting with negative shot for a two-stage filtering method

#Examples	#TP	#FP	TP price	FP reduction
Baseline	67	288	0 (0%)	0 (0%)
0	64	203	3 (4%)	85 (30%)
1	41	104	26 (38%)	184 (63%)
2	52	118	15 (22%)	170 (59%)
3	58	167	9 (13%)	121 (42%)
4	61	167	6 (8%)	121 (42%)
5	53	106	14 (20%)	182 (63%)

Table 6: Detailed results varying the number of examples using our two-stage approach. Selected examples were with human labels ["Positive", "Negative", "Positive", "Negative", "Positive"], sampled from the training set.

B A prompting template for intervention extraction baselines

B.1 Zero-shot learning

Task description: Please identify any mention of interventions in the text. Answer only the detected interventions and if more than one is found, separate them with ';' not 'and'. The answer should only contain the names of the interventions and nothing else. If no intervention is found, answer 'None'.
Task content: Text: In this study we investigated the pharmacological influence of methylphenidate (MPH) on behavioral deficits of 5xFAD mice.

Table 7: An example of a prompting template for intervention extraction baselines (zero-shot). "Task description" is for the role of "system", while "Task content" is for the role of "user".

B.2 Few-shot learning

Task description: Please identify any mention of interventions in the text. Answer only the detected interventions and if more than one is found, separate them with ';' not 'and'. The answer should only contain the names of the interventions and nothing else. If no intervention is found, answer 'None'.
Task content structure: <Examples for few-shot learning> Learn from the examples and complete the following task. <Text>
Task content: Here are examples for the task. The following is the first example. Text: Purpose: To study the effect of vitamin B2 (VB2) on the development of Alzheimer's disease (AD). Identified interventions in the text: vitamin; vitamin B2. Learn from the examples and complete the following task. Text: In this study we investigated the pharmacological influence of methylphenidate (MPH) on behavioral deficits of 5xFAD mice.

Table 8: An example of a prompting template for intervention extraction baselines (few-shot). "Task description" is for the role of "system"."Task content" is for the role of "user". "Task content structure" is a structure to create "Task content".

C A prompting template for a two-stage filtering method

C.1 Zero-shot learning

Task description: You will be provided with a text span and a block of text. Your task is to decide an entity type for the text span by considering the block of text as a context.
Task content: Text span [X]: glutathione. A block of text: Moreover the reduced activities or contents of glutathione reductase superoxide dismutase (SOD) and reduced GSH within the cortex and hippocampus caused by scopolamine were elevated by the treatment of KD-501. Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the ['intervention', 'not an intervention']

Table 9: An example of a prompting template for a two-stage filtering method (zero-shot). "Task description" is for the role of "system", while "Task content" is for the role of "user".

C.2 Few-shot learning

Task description: You will be provided with a text span and a block of text. Your task is to decide an entity type for the text span by considering the block of text as a context.
Task content structure: <Examples for few-shot learning> Learn from the examples and complete the following task. <Text span> <A block of text> Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the [‘intervention’, ‘not an intervention’]
Task content: Here are examples for the task. The following is the first example. Text span [X]: Quercetin. A block of text: Prosencephalon/metabolism/ultrastructure, Quercetin/*administration & dosage. [X] belongs to an entity type intervention. Learn from the examples and complete the following task. Text span [X]: glutathione. A block of text: Moreover the reduced activities or contents of glutathione reductase superoxide dismutase (SOD) and reduced GSH within the cortex and hippocampus caused by scopolamine were elevated by the treatment of KD-501. Please fill in the slot [Z]: [X] belongs to an entity type [Z]. Choose an entity type [Z] from the [‘intervention’, ‘not an intervention’]

Table 10: An example of a prompting template for a two-stage filtering method (few-shot). “Task description" is for the role of “system". “Task content" is for the role of “user". “Task content structure" is a structure to create “Task content".