COMPUTER RESEARCH AND MODELING
2024 VOL. 16 NO. 7 P. 1593–1600
DOI: 10.20537/2076-7633-2024-16-7-1593-1600

Ки&М

SPECIAL ISSUE

UDC: 519.8

Extraction of characters and events from narratives

A. V. Kochergin[a], Z. Sh. Kholmatova[b]

Innopolis University,
1 Universitetskaya st., Innopolis, Russia

E-mail: [a] a.kochergin@innopolis.university, [b] z.kholmatova@innopolis.university

Received 25.10.2024, after completion — 13.11.2024
Accepted for publication 25.11.2024

Events and character extraction from narratives is a fundamental task in text analysis. The application of event extraction techniques ranges from the summarization of different documents to the analysis of medical notes. We identify events based on a framework named "four W" (Who, What, When, Where) to capture all the essential components like the actors, actions, time, and places. In this paper, we explore two prominent techniques for event extraction: statistical parsing of syntactic trees and semantic role labeling. While these techniques were investigated by different researchers in isolation, we directly compare the performance of the two approaches on our custom dataset, which we have annotated.

Our analysis shows that statistical parsing of syntactic trees outperforms semantic role labeling in event and character extraction, especially in identifying specific details. Nevertheless, semantic role labeling demonstrate good performance in correct actor identification. We evaluate the effectiveness of both approaches by comparing different metrics like precision, recall, and F1-scores, thus, demonstrating their respective advantages and limitations.

Moreover, as a part of our work, we propose different future applications of event extraction techniques that we plan to investigate. The areas where we want to apply these techniques include code analysis and source code authorship attribution. We consider using event extraction to retrieve key code elements as variable assignments and function calls, which can further help us to analyze the behavior of programs and identify the project's contributors. Our work provides novel understandings of the performance and efficiency of statistical parsing and semantic role labeling techniques, offering researchers new directions for the application of these techniques.

Keywords: event extraction, natural language processing, statistical parsing, semantic role labeling

Citation: *Computer Research and Modeling*, 2024, vol. 16, no. 7, pp. 1593–1600.

КОМПЬЮТЕРНЫЕ ИССЛЕДОВАНИЯ
И МОДЕЛИРОВАНИЕ 2024 Т. 16 № 7 С. 1593–1600
DOI: 10.20537/2076-7633-2024-16-7-1593-1600

Ки&М

СПЕЦИАЛЬНЫЙ ВЫПУСК

УДК: 519.8

Извлечение персонажей и событий из повествований

А. В. Кочергин[a], З. Ш. Холматова[b]

Университет Иннополис,
Россия, г. Иннополис, ул. Университетская, д. 1

E-mail: [a] a.kochergin@innopolis.university, [b] z.kholmatova@innopolis.university

Получено 25.10.2024, после доработки — 13.11.2024
Принято к публикации 25.11.2024

Извлечение событий и персонажей из повествований является фундаментальной задачей при анализе и обработке текста на естественном языке. Методы извлечения событий применяются в самых разных областях — от обобщения различных документов до анализа медицинских записей. Мы определяли события на основе структуры под названием «четыре W» (кто, что, когда, где), чтобы охватить все основные компоненты событий, такие как действующие лица, действия, время и места. В этой статье мы рассмотрели два основных метода извлечения событий: статистический анализ синтаксических деревьев и семантическая маркировка ролей. Хотя эти методы были изучены разными исследователями по отдельности, мы напрямую сравнили эффективность двух подходов на собранном нами наборе данных, который мы разметили.

Наш анализ показал, что статистический анализ синтаксических деревьев превосходит семантическую маркировку ролей при выделении событий и символов, особенно при определении конкретных деталей. Тем не менее, семантическая маркировка ролей продемонстрировала хорошую эффективность при правильной идентификации действующих лиц. Мы оценили эффективность обоих подходов, сравнив различные показатели, такие как точность, отзывчивость и F1-баллы, продемонстрировав, таким образом, их соответствующие преимущества и ограничения.

Более того, в рамках нашей работы мы предложили различные варианты применения методов извлечения событий, которые мы планируем изучить в дальнейшем. Области, в которых мы хотим применить эти методы, включают анализ кода и установление авторства исходного кода. Мы рассматриваем возможность использования методов извлечения событий для определения ключевых элементов кода в виде назначений переменных и вызовов функций, что в дальнейшем может помочь ученым проанализировать поведение программ и определить участников проекта. Наша работа дает новое понимание эффективности статистического анализа и методов семантической маркировки ролей, предлагая исследователям новые направления для применения этих методов.

Ключевые слова: извлечение событий, обработка естественного языка, статистический анализ, семантическая маркировка ролей

Introduction

Extracting characters and events from narratives is a fundamental task in natural language processing (NLP) and computer narrative analysis. The term "event" refers to an action described in the text, enriched with information and constituting an entity containing the following components:

- The actor;
- The action itself;
- The time of the event;
- The place.

This definition of an event aligns with the concept of the "four W": "Who", "What", "When", "Where". In some works, the researchers used the same concept, but with an extension to the fifth "W" corresponding to the question "Why" [Yu, Kim, 2021].

Extracting events requires accurate text analysis to understand the semantic constructions and obtain what this text describes and what events are transmitted in the narrative. The complexity of this task has gained popularity across different domains. For example, events are extracted from folk tales [Valls-Vargas, Zhu, Ontanón, 2014], news narratives [Zhang, Boons, Batista-Navarro, 2019; Glavaš et al., 2014], social media [Conte, Troncy, Naaman, 2014; Becker et al., 2012], medical notes [Isakov, Kovalchuk, 2023; Schäfer et al., 2023].

Event extraction has many applications, including:

- Decomposing text into events for convenient perception of text by a person in a short form;
- Analyzing text by a machine from the point of view of events to automatically determine real events in the world and quickly react to them;
- Searching for events in various sources to mark actions of interest for any task.

In this work, we address the task of extracting events from the text using two key techniques:

- statistical parsing into a syntactic tree;
- semantic role labeling.

Our work introduces a novel perspective by directly comparing two significant techniques within the same experimental framework. While the researchers investigated the methods individually, our work highlights the relative strengths and limitations of statistical parsing and semantic role labeling in handling event extraction from narrative texts. Applying the under techniques consideration to a custom dataset, we demonstrate their properties. The main contribution of this paper lies in a comprehensive analysis of both methods, that gives the researchers and practitioners new insights that can inform their application across diverse domains.

The paper is organized as follows. "Literature review" briefly summarizes the work done in the area of events and characters extraction. The research methodology is presented in "Methodology". "Results and discussion" is dedicated to the results and discussion, while "Conclusion" describes the conclusion of our work.

Literature review

There are many articles on the topic of extracting events from the text, as well as articles reviewing existing results. However, there is still insufficient information about what technologies exist to solve the problem of high-quality event extraction for further machine text processing.

[Isakov, Kovalchuk, 2023] proposed the methodology for extracting events from unstructured medical texts, which may contain errors, terms, and abbreviations. They assumed that to define an event, it is enough to consider the two main components that make the event — a trigger and an argument. A trigger is either a verb or a noun. An argument is the main entity that plays a role in an event, and an argument can also be a participant performing an action or time of the event. The researchers built a method that involves constructing syntactic trees for each text, with events identified based on the extraction of verbs, abbreviations, and collocations, with each verb forming the basis of an event within the tree. The study emphasizes the importance of accurately detecting the trigger, as accurate detection leads to successful event recognition.

[Hong et al., 2011] employed an approach that collects candidate sentences that contain entities, and then determines whether there is an event in them or not. To identify the type of entity in the process of recognized entities, the researchers made search requests and compared the information obtained with the prepared training data, thus determining the category of the entity. The authors managed to increase the efficiency of event extraction by using the cross-entity inference method. When a suspected event and its type are found, the system assumes the presence of other related events and searches for them.

[Peng et al., 2024] presented an approach for event extraction that shows high performance, and is suitable for compression and use in low-resource platforms. The researchers used a pre-trained language model (PLM) to extract events and bring new technology to the process that improves the completeness and accuracy of the results. They used templates and keywords so that the model extracts the necessary data from the event mentioned and fills in the template with it. This helped them achieve the controlled receipt of the expected result. The expected result in this case was a description of the event from the source text, limited by the provided template. Thus, according to the authors, more opportunities can be extracted from the great potential of understanding the language of the PLM used.

[Zhang, Boons, Batista-Navarro, 2019] needed to extract events from the text to solve the task associated with the analysis of news articles. Event extraction was built on semantic role labeling tools (SRL). Labeling semantic roles means defining predicates and their arguments within a sentence. They used a combination of two instruments: Semafor and Deep SRL. For a complete and most accurate definition of the actors in the events within a single sentence, it is necessary to have a context related to the persons appearing in the narrative in previous sentences. This task was solved using named entity recognition (NER) tools.

[Schäfer et al., 2023] extracted events related to taking medications from medical texts. To solve the problem of recognizing named entities, the researchers used the ClinicalBERT and Clinical-Longformer models, specially trained on medical texts, as well as several other pre-trained models based on BERT. Event extraction in this study was presented as a task similar to NER, because it was the recognition of events related to detected drugs that mattered, without enriching these events with context. Thus, the event recognition process included the use of the DeBERTa v3 and LinkBERT models.

Despite the advances considered in the existing works, challenges remain in achieving accurate event extraction, especially in working with unstructured texts and achieving contextual understanding.

Methodology

In this work, an event is defined as a construction in the text denoting an action that was done by a certain subject. The text may also contain secondary information about the event: time and place. We decided to use the verb as a trigger, by which the algorithm determines the presumed presence of an event in a sentence. The same was done in the work of [Isakov, Kovalchuk, 2023].

The minimum information that is sufficient to indicate an event is the actor and the action. The time and place may be missing, while the meaning of the event is not lost. For this reason, we skipped all the triggers for which the actor could not be found.

The task of extracting events from the text requires efficient parsing of the text into a specific structure that can be analyzed programmatically. To solve the problem of structuring the tokens (words) that form the text in a logical order suitable for analysis, we considered two commonly used approaches: semantic role labeling and parsing sentences into syntactic trees.

Semantic role labeling is the designation of semantic connections between words in a sentence. The semantic roles of words are determined relative to other words in a sentence. Thus, when the verb is chosen as a starting point when constructing an event from a sentence, the semantic connection of the selected verb with other words can be used to determine the necessary remaining components of the event, such as the actor, time, and place. The advantage of this method is that it accurately determines the character of the verb. Due to the unambiguous connections between words, it is possible to accurately determine this. With a Python tool named Spacy [Honnibal, Montani, 2017] we determined the actor and the object to which the action was directed, and tried to determine other circumstances of the action — the place and time.

Given a sentence S, the SRL process first selects the verb p and then try to identify other components of the event based on the syntactic relationships:

$$E = (A, T, P),$$

where A is the actor, T — time, and P — the place.

Through the Spacy tool, the SRL model assigns roles as follows:

$$P(A_i \mid p, S) = \arg\max_j P(A_j \mid \text{features}(p, S_j)),$$

where features(p, S_j) are surrounding word contexts and $\arg\max_j$ selects the most likely argument A_j for each role (actor, time, place).

Parsing sentences into syntactic trees is vital for event extraction because it reveals the grammatical structure and relationships between sentence elements, such as actors and actions.

Statistical parsing aims to find the most likely syntactic tree T for a sentence S:

$$T^* = \arg\max_T P(T \mid S),$$

where T^* is the optimal syntactic tree, and $P(T \mid S)$ is the probability of a tree T given a sentence S. Worth mentioning that the probability of a syntactic tree is usually presented as a product of conditional probabilities of the tree's nodes.

Stanford CoreNLP [Manning et al., 2014] provides the functionality of a statistical parser that analyzes a sentence based on a large amount of statistical data and builds a syntactic tree from it, transmitting the language structure with high accuracy. The nodes of the tree contain labels in the form of Penn Treebank II tags [Bies et al., 1995], which are divided into levels: Clause level, Phrase level, and Word level.

We implemented a syntax tree analysis using the depth-first-search algorithm. The algorithm goes through the tree and searches for certain syntactic structures that can be interpreted as a designation of an actor, action, place, and time. During the implementation process, we considered sentences with various syntactic constructions and their corresponding syntactic trees. In some cases, similar syntactic structures were interpreted by the parser into different trees, and these cases need to be validated by additional checks in the algorithm.

Results and discussion

To check the work of the approaches considered we worked on our dataset. To create a dataset, 100 sentences were taken from different sources. Each of the selected sentences contains all four necessary components of the event. We received a draft version of the annotation of these sentences in the form of extracted events using a large language model. Then we manually validated this annotation to create a golden list of examples.

We have separately calculated the metrics for each of the approaches. F1-score, precision, and recall are presented in Table 1 for the statistical parser and in Table 2 for the semantic role labeling approach.

Table 1. Metrics of automatic evaluation for Stanford parser

Event part	F1-score	Precision	Recall
Actor	97 %	97 %	97 %
Action	89.4 %	91.1 %	87.7 %
Time	55.8 %	55.8 %	55.9 %
Place	73.4 %	73.5 %	73.4 %

Table 2. Metrics of automatic evaluation for SRL with spacy

Event part	F1-score	Precision	Recall
Actor	91.8 %	92.3 %	91.3 %
Action	81.1 %	83.1 %	79.3 %
Time	50.1 %	50.6 %	49.7 %
Place	61.1 %	63.2 %	59.1 %

As can be seen from the table, the approach using statistical parsing and searching for syntactic constructions along the tree gave better results. This is because statistical parsing allows us to flexibly navigate through different sentences, dividing them into logical parts.

We have also manually checked the effectiveness of the algorithms considered. We concentrated on a detailed examination of individual cases and gained the following insights:

- Syntactic constructions translated into a tree with statistical parsing can be the same for different semantic cases. For example, “On Monday” was often interpreted by the algorithm as a place, not as a time, because in the tree this construction looks identical to the phrase “In the park”;
- Labeling semantic roles shows good results for determining the subject to which the verb belongs but does not provide sufficient results in correctly determining the circumstances of the action and contextual information.

As one can notice, while the semantic role labeling approach shows good results in determining the actor, statistical parsing allows for more flexible text processing and coping with a bigger number of cases, which makes it a more effective tool for extracting events.

Conclusion

In this work, we investigated two approaches for event extraction from narratives — statistical parsing and semantic role labeling. We also evaluated the performance of these methods using a custom dataset. The dataset was annotated using a large language model. Moreover, it underwent our manual validation in order to create a reliable gold standard for evaluation. We calculated precision, recall, and F1-scores for each approach and compared the results.

Our findings revealed that the statistical parsing approach shows better results than semantic role labeling. However, semantic role labeling performed well in identifying the actors of events, but it struggled with the identification of other contextual elements. The flexibility of statistical parsing allowed us to work with syntactic variations across different sentences.

Through manual analysis, we observed that some syntactic structures indicating circumstantial information, such as “On Monday” (time) and “In the park” (place), were sometimes misinterpreted by the algorithm. This observation highlighted the strengths of statistical parsing in identifying such subtle details during event extraction.

Overall, the results demonstrate that while both approaches have their advantages, statistical parsing can be considered as a more effective method for event extraction from text narratives.

In future work, we plan to apply event extraction techniques to the analysis of source code, considering code as a structured text [Romanov et al., 2020; Romanov, Ivanov, Succi, 2020; Ivanov et al., 2021]. By identifying key events like variable assignments, and function calls, we can analyze the underlying behavior of the code. We also plan to investigate the application of event extraction techniques to source code authorship attribution [Bogdanova et al., 2022]. These techniques can help us to identify individual developers’ contributions by extracting events that are stylistically unique to software engineers.

References

Becker H., Iter D., Naaman M., Gravano L. Identifying content for planned events across social media sites // Proceedings of the fifth ACM international conference on Web search and data mining. — 2012. — P. 533–542.

Bies A., Ferguson M., Katz K., MacIntyre R., Tredinnick V., Kim G., Marcinkiewicz M. A., Schasberger B. Bracketing guidelines for Treebank II style Penn Treebank project. — University of Pennsylvania, 1995. — Vol. 97. — P. 100.

Bogdanova A., Farina M., Kholmatova Z., Kruglov A., Romanov V., Succi G. Analysis of source code authorship attribution problem // 2022 International Conference on Computers and Artificial Intelligence Technologies (CAIT). — 2022. — P. 109–115.

Conte C. A., Troncy R., Naaman M. Extracting resources that help tell events’ stories // SoMuS@ ICMR. — 2014.

Glavaš G., Šnajder J., Kordjamshidi P., Moens M.-F. HiEve: A corpus for extracting event hierarchies from news stories // Proceedings of 9th language resources and evaluation conference. — 2014. — P. 3678–3683.

Hong Y., Zhang J., Ma B., Yao J., Zhou G., Zhu Q. Using cross-entity inference to improve event extraction // Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies. — 2011. — P. 1127–1136.

Honnibal M., Montani I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. — 2017. — To appear.

Isakov T., Kovalchuk S. Methodology of event extraction from unstructured medical texts on the example of the Russian language // Procedia Computer Science. — 2023. — Vol. 229. — P. 101–108.

Ivanov V., Romanov V., Succi G. et al. Predicting type annotations for python using embeddings from graph neural networks // Proceedings of the 23rd International Conference on Enterprise Information Systems (ICEIS 2021). — 2021. — P. 548–556.

Manning C. D., Surdeanu M. Bauer J., Finkel J. R., Bethard S., McClosky D. The Stanford CoreNLP natural language processing toolkit // Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations. — 2014. — P. 55–60.

Peng J., Yang W., Wei F., He L. Prompt for extraction: Multiple templates choice model for event extraction // Knowledge-based systems. — 2024. — P 111544.

Romanov V., Ivanov V., Succi G. Approaches for representing software as graphs for machine learning applications // 2020 International Computer Symposium (ICS). — 2020. — P. 529–534.

Romanov V., Ivanov V., Succi G. et al. Representing programs with dependency and function call graphs for learning hierarchical embeddings // Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS). — 2020. — Vol. 2. — P. 360–366.

Schäfer H., Idrissi-Yaghir A., Bewersdorff J., Frihat S., Friedrich C. M., Zesch T. Medication event extraction in clinical notes: Contribution of the WisPerMed team to the n2c2 2022 challenge // Journal of Biomedical Informatics. — 2023. — Vol. 143. — P. 104400.

Valls-Vargas J., Zhu J., Ontanón S. Toward automatic role identification in unannotated folk tales // Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. — 2014. — Vol. 10, No. 1. — P. 188–194.

Yu H.-Y., Kim M.-H. Automatic event extraction method for analyzing text narrative structure // 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM). — 2021. — P. 1–4.

Zhang H., Boons F., Batista-Navarro R. Whose story is it anyway? Automatic extraction of accounts from news articles // Information Processing & Management. — 2019. — Vol. 56, No. 5. — P. 1837–1848.