Streamlining Event Relation Extraction: A Pipeline Leveraging Pretrained and Large Language Models for Inference

Flores Gustavo Miguel[1,*ä†]ä Youssra Rebboud[1,†]ä Pasquale Lisena[1,†] and Raphäel Troncy[1,†]

[1]EURECOM, Sophia Antipolis, Biot, France

Abstract

Event relation extraction is crucial for understanding the temporal sequence and interconnections between events. To demonstrate this, we developed a Streamlit-based application that showcases our event relation extraction system, capable of identifying semantically accurate relations such as *Direct-cause, Enable, Intend, and Prevent.* The system features an API that simplifies inference and displays results in a user-friendly manner. Users can input text like a sentence and the application highlights extracted events and their corresponding relationships. The backend runs a series of pre-trained language models, trained on datasets focused on events and their semantic relations. The app allows users to switch between various models, including HuggingFace's RoBERTa, REBEL, and large language models like Zephyr. The demo is available at https://demo.kflow.eurecom.fr/.

Keywords

Event Relation Extraction, Event Knowledge Graphs, Causal Event Relations, Web Platform,

1. Introduction

Understanding the flow of events and their interconnections is crucial for tasks such as narrative com-prehension, historical analysis, and machine learning applications. The way information is represented significantly impacts the contextual knowledge models can access, often encoded through relational triplets. While knowledge about entities is important, understanding the context surrounding those entities especially events is equally vital. Events are instances that occur in time and space, inherently existing within a web of causal relationships that can provide critical insights into their nature and consequences [1].

Given the significance of event knowledge, researchers have developed various methods to represent events and their relationships. Event Relation Extraction (ERE) is the task of identifying and predicting relationships between events in text, enabling a deeper understanding of their progression and impact [2]. However, with the proliferation of different machine learning models, architectures, and tuning parameters, evaluating their performance on this task remains challenging.

This research aims to address this challenge by developing a comprehensive Event Relation Extraction pipeline. The pipeline allows users to experiment with various models and datasets, enabling them to input sentences, extract events, and visualize the relationships between them. For everyday users, this pipeline enhances the understanding of event flows in textual data. For researchers, it offers a qualitative evaluation tool that allows them to analyze and compare model performance on event relation extraction tasks, identifying potential strengths and weaknesses.To make this accessible, we developed a user-friendly Streamlit web application https://demo.kflow.eurecom.fr/ that visually presents the extracted events and their relations. The application supports

EKAW 2024: EKAW 2024 Workshops, Tutorials, Posters and Demos, 24th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2024), November 26-28, 2024, Amsterdam, The Netherlands.

[*]Corresponding author.

[†]These authors contributed equally.

gustavo.flores@eurecom.fr (F. G. Miguel); youssra.rebboud@eurecom.fr (Y. Rebboud); pasquale.lisena@eurecom.fr (P. Lisena); raphael.troncy@eurecom.fr (R. Troncy)

https://ryoussra.github.io/ (Y. Rebboud); https://pasqlisena.github.io/ (P. Lisena); http://www.eurecom.fr/~troncy (R. Troncy)

0009-0003-1018-0172 (F. G. Miguel); 0000-0003-3507-5646 (Y. Rebboud); 0000-0003-3094-5585 (P. Lisena); 0000-0003-0457-1436 (R. Troncy)

multiple pre-trained models and datasets, providing an interactive platform for generating and analyzing inferences.

The structure of this demo paper is the following: In Section 2 we cover the related work in event and relation extraction. Section 3 explains the pipeline architecture, detailing the tasks performed, how inferences are generated, and how users can interact with the system. Finally, Section 4 highlights observations, potential improvements, and future development directions.

2. Related work

The study of event relations has historically focused on temporal relationships, where researchers aimed to represent the temporal order of events [3]. Subsequently, attention shifted towards causal relations between events, which sought to understand the influence of one event on another. In these causal relationships, the cause is typically regarded as the subject, while the effect is viewed as the object.In our work, we aim to move beyond basic causality and focus on extracting fine-grained causal relationships between events. These nuanced event relations, initially introduced by [1], were accompanied by the creation of the first dataset specifically designed to capture such detailed event relations.

Event Relation Extraction (ERE) is generally divided into two main subtasks: (1) identifying the type of relation, and (2) extracting the corresponding spans of the subject and object from the sentence. Early work in this domain was carried out by the Linguistic Data Consortium (LDC) through the Automatic Content Extraction (ACE) program [4], which focused on texts from various domains such as newswire, broadcast news, conversational speech, weblogs, Usenet, and telephone conversations. The primary objective of ACE was to develop information extraction techniques that could facilitate the automatic processing of human language in textual form.

In recent years, neural models have gained prominence in event extraction tasks. With advancements in deep learning, researchers have explored the use of Convolutional Neural Networks (CNNs) [5], Recurrent Neural Networks (RNNs) [6], and, more recently, transformer-based models [7]. Pretrained language models (PLMs) have become a focal point in event extraction studies due to their ability to learn general-purpose representations from raw text, which aids in extracting relevant event relations [3]. BERT, in particular, has demonstrated strong performance in this area, as highlighted in a study [7] showing that BERT could achieve state-of-the-art results without the need for task-specific architectures or external resources [3]. Large Language Models (LLMs) have demonstrated strong performance in relation extraction tasks. In the work of [8], the Flan-T5 model [9] significantly outperformed previous baselines on the CoNLL04 dataset [10], underscoring the potential of LLMs for event relation extraction.

For precise event relations, such as *Direct-cause, Enable, Intend, and Prevent*, [11] proposed an approach to augment the dataset introduced by [1] using GPT. They then employed BERT [12] to perform event relation extraction tasks. While their method achieved good performance on the relation classification subtask, it showed limitations in the quality of event extraction.

In this work, we aim to provide an API based on an event relation extraction pipeline that leverages various pre-trained language models (PLMs) and large language models (LLMs) instead of relying solely on BERT [12]. Although detailed performance results cannot be shared here, as they are under review in another study, we offer insights into the models performance and provide access to the code and a link to the API.

3. Platform and API for Event Relation Extraction

3.1. Event Relation Extraction from Text

In our pipeline, the goal is to perform event relation extraction from textual data, focusing on four semantically precise event relations: *Direct-Cause, Enable, Intend, and Prevent* [1]. These relations are categorized under the broader supertype of *Cause*.

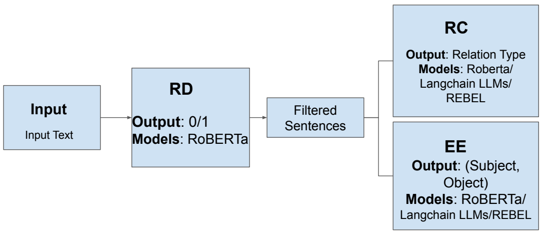

Input
Input Text
RD
Output: 0/1
Models: RoBERTa
Filtered Sentences
RC
Output: Relation Type
Models: Roberta/ Langchain LLMs/ REBEL
EE
Output: (Subject, Object)
Models: RoBERTa/ Langchain LLMs/REBEL

Figure 1: The ERE pipeline workflow

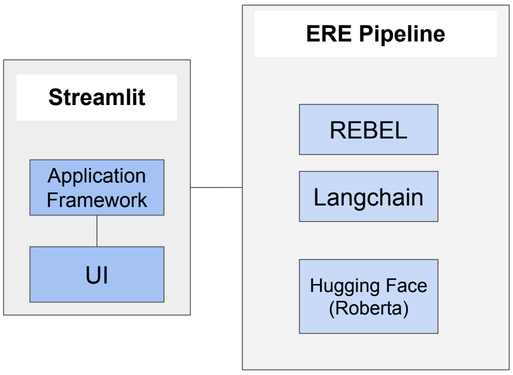


Figure 2: Streamlit UI and Application Framework

The pipeline performs three tasks: Relation Detection(RD), Relation Classification(RC), and Event Extraction(EE). Dividing the task into three subtasks could enable testing a broader combination of models for each task, allowing evaluation of strengths and weaknesses for each subtask independently. In the RD phase, the model filters out sentences that do not have a causal event relation, this task is not optional. The sentences containing a causal relation will passe to the RC module. At this level, the causal sentence will be given as input to the RC module to determine which type of event relation is in the sentence from the four relations. Finally, when we decide the relation type, the EE module will extract the subject and the object of the event relation in a given sentence. Figures 1 illustrates the pipeline modules.

The most integral component of the pipeline is the ERE models. The pipeline runs only one model for each task (RC, RD, and EE), chosen from the available options. At the present, the models included are:

- the BERT family of models by Hugging Face for (RC, RD, EE)[13];
- REBEL for (RD, EE)[14];
- the large language models (LLM) available through the LangChain library[1] for (RD, EE). The available LLMs are: Zephyr[15], DPO[16], UNA[17], SOLAR[18], and GPT4[19].

Both the BERT family models and REBEL were trained using a combination of two datasets, *The Event Relations Dataset* from [11], and the CausalNewsCorpus [20] which made a total of 5613 example sentences annotated with the four relations *Direct-Cause, Enable, Intend, and Prevent* together with the subject and object of each relation. On the other hand, The same prompt template [2] was designed for every LLM that we have been using. The chosen LLMs ranked among the top performers on the Huggingface Open LLM Leaderboard at the time of writing, excelling across various benchmarks, including the Multi-Task Language Understanding Benchmark (MMLU) [21].

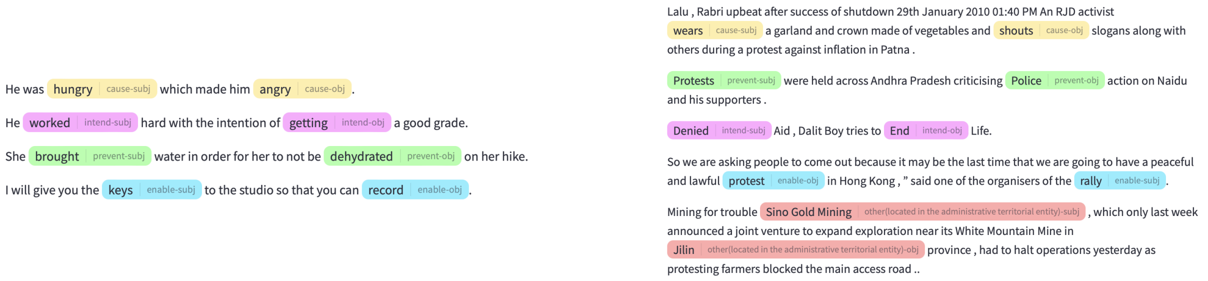
The RoBERTa model performed well in the relation detection task, achieving an average F1-score of 0.86. In contrast, the REBEL model excelled in both relation classification and event extraction tasks, with F1-scores of 0.975 and 0.829, respectively, showcasing its overall effective-ness. The performance details of these models are currently under review for another conference and cannot be disclosed at this time. However, the code and data for this work are available at: https://github.com/ANR-kFLOW/Relation_extraction/tree/main. Figures 3 shows an example of an accurate and an inaccurate prediction produced using RoBERTa as a filter (RD) and REBEL for both RC and EE.

3.2. Relation Extraction Pipeline

The pipeline is made in Python and the specifications for the inferences can be passed through: command line, a configuration file, or through the user interface developed for the pipeline.

[1] https://Python.langchain.com/

[2] https://github.com/ANR-kFLOW/Relation_extraction/blob/main/LLMs_as_Relation_Classifiors_and_ Event_Extractors/ prompt_template.yml



He was hungry cause-subj which made him angry cause-obj .

He worked intend-subj hard with the intention of getting intend-obj a good grade.

She brought prevent-subj water in order for her to not be dehydrated prevent-obj on her hike.

I will give you the keys enable-subj to the studio so that you can record enable-obj .

Lalu , Rabri upbeat after success of shutdown 29th January 2010 01:40 PM An RJD activist wears cause-subj a garland and crown made of vegetables and shouts cause-obj slogans along with others during a protest against inflation in Patna .

Protests prevent-subj were held across Andhra Pradesh criticising Police prevent-obj action on Naidu and his supporters .

Denied intend-subj Aid , Dalit Boy tries to End intend-obj Life.

So we are asking people to come out because it may be the last time that we are going to have a peaceful and lawful protest enable-obj in Hong Kong , " said one of the organisers of the rally enable-subj .

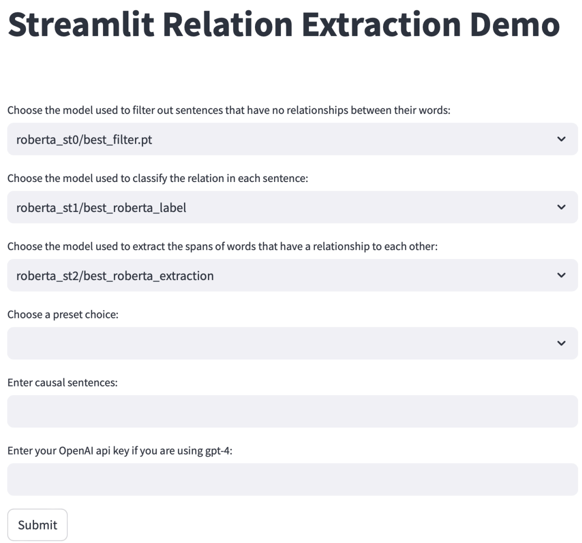
Mining for trouble Sino Gold Mining other(located in the administrative territorial entity)-subj , which only last week announced a joint venture to expand exploration near its White Mountain Mine in Jilin other(located in the administrative territorial entity)-obj province , had to halt operations yesterday as protesting farmers blocked the main access road ..

Figure 3: An Example of an Accurate and an Inaccurate Inference Produced by REBEL

After training, the pretrained model can be made available to the pipeline by saving the trained models in a common folder, making them available at for the inference stage.

Information that the pipeline can receive from users is: the path to a pretrained model for a given task, the choice to skip performing one of the tasks, and the user's OpenAI key (if GPT4 is used for any of the tasks). The pipeline has a default that configuration that is ran if the user does not input any instructions. If the user inputs instructions that does not cover all arguments then the pipeline will fill in the missing arguments with the default values.

3.3. Streamlit Platform Architecture



Streamlit Relation Extraction Demo

Choose the model used to filter out sentences that have no relationships between their words:

roberta_st0/best_filter.pt

Choose the model used to classify the relation in each sentence:

roberta_st1/best_roberta_label

Choose the model used to extract the spans of words that have a relationship to each other:

roberta_st2/best_roberta_extraction

Choose a preset choice:

Enter causal sentences:

Enter your OpenAI api key if you are using gpt-4:

Figure 4: This is a screenshot of the demo and the choices that users are able to make. The first slider determines which model does the pipeline use to filter out non-event relation sentences. The second drop down menu determines which model is used to classify the relation in the sentences. The third menu determines which model is used to extract the span of (Subject, Object) for the event relation in the sentence. The preset choice allows the user to use premade example sentences to demonstrate the capacity of the models. The choice below is for users that want to input their own sentences. The final drop down menu is where a valid OpenAI api key needs to be inserted if GPT4 is chosen for any of the tasks. A user would press submit to run the pipeline once the choices have been made.

This application serves to provide users with a curated demonstration of the capabilities of the models. This application is developed using Streamlit [3], which acts as both the web application and the web API as shown in 2. the Streamlit application receives input from the user and passes it along to the Python

[3] https://streamlit.io/

pipeline via a configuration file. The user can write their own text or use an input preset. After the user makes his choice of the model used for each task,the inference running in the Back-End will be produced.

The output returned to the user will include his original text used to produce the inference with highlights of subject and object of the extracted event relation. Next to the highlights are labels indicating what part of the span it is (subject or object) together with the event relation type.

There are two different versions for how the highlighting is formatted: one for spans that do not overlap, and one for spans that overlap. In the case of spans that do not overlap, the color of the highlights are color coded for the classification of the event relation and there are labels at the end of each highlight. In the case where the spans can overlap one another the spans are represented by being encased in color coded brackets. The color for the bracket indicates what part of the span the bracket contains(subject or object). The labels are placed at the closing bracket to avoid cluttering up the sentence. The classification for the relation can either be: cause, intend, prevent, enable, or other. Other refers to when the model producing the classifications gives a nonstandard response. Some models such as RoBERTa[13] identify multiple event relations in a given sentence. In that case the sentences that have multiple event relations will be displayed multiple times for each event relation detected. This makes it so that there is only one span displayed at a time, for visual clarity.
Figures 4 shows a screen of the demo with Streamlit.

4. Conclusion and Future Work

In this work we have constructed an API for event relation extraction based on a set of pretrained language models, BERT, RoBERTa, and REBEL together with few LLMs such as GPT4, and Zephyr. The API was created to help stream line the process of preforming inferences on textual input from a given user, and aiding the process of comparing ERE models to one another. The API is accessible at
https://demo.kflow.eurecom.fr/.

In the future, the platform will allow a user to compare multiple models in an A/B testing fashion. The A/B testing happens in the user comparing the inferences generated by the models side by side and recording their evaluation of how one model compares to another. First, an automatic test will apply widely adopted metrics – e.g. precision, recall and F1-score – on a predefined ground truth to evaluate the performance of the models. These comparisons and evaluations will be saved so that users in future can use these metrics to determine what are the best performing models. The best 3 models for a given task will be selected for human evaluation through an UI. A future addition to the pipeline can be including functionality to be able to train the models by using the pipeline.

Acknowledgements

This work has been partially supported by the French National Research Agency (ANR) within the kFLOW project (Grant nřANR-21-CE23-0028).

References

[1] Y. Rebboud, P. Lisena, R. Troncy, Beyond Causality: Representing Event Relations in Knowledge Graphs, in: Knowledge Engineering and Knowledge Management: 23rd International Conference, EKAW 2022, Bolzano, Italy, September 2629, 2022, Proceedings, Springer-Verlag, Berlin, Heidelberg, 2022, p. 121135. doi:10.1007/978-3-031-17105-5_9.

[2] Z. Hu, Z. Li, D. Xu, L. Bai, C. Jin, X. Jin, J. Guo, X. Cheng, Protoem: A prototype-enhanced matching framework for event relation extraction, 2023. URL: https://arxiv.org/abs/2309.12892. arXiv:2309.12892.

[3] K. Liu, Y. Chen, J. Liu, X. Zuo, J. Zhao, Extracting events and their relations from texts: A survey on recent research progress and challenges, AI Open 1 (2020) 22–39. URL: https://www.

sciencedirect.com/science/article/pii/S266665102100005X. doi:https://doi.org/10.1016/j.aiopen.2021.02.004.

[4] C. Walker, L. D. Consortium, Ace 2005 multilingual training corpus, 2006.

[5] Y. Chen, L. Xu, K. Liu, D. Zeng, J. Zhao, Event extraction via dynamic multi-pooling convolutional neural networks, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers, The Association for Computer Linguistics, 2015, pp. 167–176. URL: https://doi.org/10.3115/v1/p15-1017. doi:10.3115/V1/P15-1017.

[6] T. H. Nguyen, K. Cho, R. Grishman, Joint event extraction via recurrent neural networks, in: K. Knight, A. Nenkova, O. Rambow (Eds.), Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, San Diego, California, 2016, pp. 300–309. URL: https://aclanthology.org/N16-1034. doi:10.18653/v1/N16-1034.

[7] S. Yang, D. Feng, L. Qiao, Z. Kan, D. Li, Exploring pre-trained language models for event extraction and generation, in: A. Korhonen, D. Traum, L. Màrquez (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 5284–5294. URL: https://aclanthology.org/P19-1522. doi:10.18653/v1/P19-1522.

[8] S. Wadhwa, S. Amir, B. Wallace, Revisiting Relation Extraction in the era of Large Language Models, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 15566–15589. URL: https://aclanthology.org/2023.acl-long.868. doi:10.18653/v1/2023.acl-long.868.

[9] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, J. Wei, Scaling instruction-finetuned language models, Journal of Machine Learning Research 25 (2024) 1–53. URL: http://jmlr.org/papers/v25/23-0870.html.

[10] D. Roth, W.-t. Yih, A linear programming formulation for global inference in natural language tasks, in: Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004, Association for Computational Linguistics, Boston, Massachusetts, USA, 2004, pp. 1–8. URL: https://aclanthology.org/W04-2401.

[11] Y. Rebboud, P. Lisena, R. Troncy, Prompt-based Data Augmentation for Semantically-Precise Event Relation Classification, in: ESWC Workshops, 2023. URL: https://api.semanticscholar.org/CorpusID:260209196.

[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.

[13] L. Zhuang, L. Wayne, S. Ya, Z. Jun, A robustly optimized BERT pre-training approach with post-training, in: S. Li, M. Sun, Y. Liu, H. Wu, K. Liu, W. Che, S. He, G. Rao (Eds.), Proceedings of the 20th Chinese National Conference on Computational Linguistics, Chinese Information Processing Society of China, Huhhot, China, 2021, pp. 1218–1227. URL: https://aclanthology.org/2021.ccl-1.108.

[14] P.-L. Huguet Cabot, R. Navigli, REBEL: Relation Extraction By End-to-end Language generation, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 2370–2381. URL: https://aclanthology.org/2021.findings-emnlp.204. doi:10.18653/v1/2021.findings-emnlp.204.

[15] L. Tunstall, E. Beeching, N. Lambert, N. Rajani, K. Rasul, Y. Belkada, S. Huang, L. von Werra, C. Fourrier, N. Habib, N. Sarrazin, O. Sanseviero, A. M. Rush, T. Wolf, Zephyr: Direct distillation of lm alignment, 2023. URL: https://arxiv.org/abs/2310.16944. arXiv:2310.16944.

[16] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, C. Finn, Direct preference optimization: Your language model is secretly a reward model, 2024. URL: https://arxiv.org/abs/2305.18290. arXiv:2305.18290.

[17] fblgit, Una-thebeagle, 2024. URL: https://huggingface.co/fblgit/UNA-TheBeagle-7b-v1.

[18] D. Kim, C. Park, S. Kim, W. Lee, W. Song, Y. Kim, H. Kim, Y. Kim, H. Lee, J. Kim, C. Ahn, S. Yang, S. Lee, H. Park, G. Gim, M. Cha, H. Lee, S. Kim, Solar 10.7b: Scaling large language models with simple yet effective depth up-scaling, 2024. URL: https://arxiv.org/abs/2312.15166. arXiv:2312.15166.

[19] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, ukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, ukasz Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, B. Zoph, Gpt-4 technical report, 2024. URL: https://arxiv.org/abs/2303.08774. arXiv:2303.08774.

[20] F. A. Tan, A. Hürriyetoğlu, T. Caselli, N. Oostdijk, T. Nomoto, H. Hettiarachchi, I. Ameer, O. Uca, F. F. Liza, T. Hu, The Causal News Corpus: Annotating Causal Relations in Event Sentences from News, in: N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, S. Piperidis (Eds.), Proceedings of the Thirteenth Language Resources and Evaluation Conference, European Language Resources Association, Marseille, France, 2022, pp. 2298–2310. URL: https://aclanthology.org/2022.lrec-1.246.

[21] Y. Wang, X. Ma, G. Zhang, Y. Ni, A. Chandra, S. Guo, W. Ren, A. Arulraj, X. He, Z. Jiang, T. Li, M. W. Ku, K. Wang, A. Zhuang, R. R. Fan, X. Yue, W. Chen, Mmlu-pro: A more robust and challenging multi-task language understanding benchmark, ArXiv abs/2406.01574 (2024). URL: https://api.semanticscholar.org/CorpusID:270210486.