LLMs Struggle on Explicit Causality in Italian

Alessandro Bondielli[1,2,*], Martina Miliani[2], Luca Paglione[2], Serena Auriemma[2], Lucia Passaro[1,2] and Alessandro Lenci[2]

[1]Department of Computer Science, University of Pisa
[2]CoLing Lab, Department of Philology, Literature and Linguistics, University of Pisa

Abstract

The ability to recognize and interpret causal relations is fundamental for building robust intelligent systems. Recent research has focused on developing benchmarks and tasks to evaluate the inferential and causal reasoning capabilities of LLMs, such as the Pairwise Causal Discovery (PCD) task. However, most of these resources are limited to English. In this paper, we present Expli**CITA**, a translation of the English ExpliCa dataset [1], which is the first publicly available dataset for joint temporal-causal reasoning in Italian, enabling evaluation of LLMs on Italian PCD. We conduct an extensive empirical study across 20 Italian and multilingual models of varying sizes and training strategies, combining a perplexity-based evaluation of causal reasoning competence with multiple-choice prompting tasks in both zero-shot and few-shot settings. Our results show that all tested models, including the GPT family, struggle with the Expli**CITA** PCD task, more so than with the original English ExpliCa, in both evaluation scenarios. Moreover, native Italian models do not outperform fine-tuned multilingual alternatives. Consistent with prior findings, we observe that the linguistic competence of models, measured using perplexity-based metrics, is higher than their respective performances, measured via accuracy on prompting results; however, this gap tends to narrow with increasing model size. Finally, a per-class performance analysis reveals that models handle causal relations relatively better than temporal ones.

Keywords

LLMs, Causal Reasoning, Language Resources, Evaluation, Benchmarking

1. Introduction

Recognizing *causal relations* is a core human cognitive skill. Causal understanding is in fact fundamental to intelligent reasoning [2]. Thus, a strong AI system should be capable of performing causal reasoning.

The past few years have in fact seen a vigorous debate about the extent to which large language models (LLMs) are actually capable of genuine inference, beyond mere pattern matching [3, 4, 5]. Among the inferences a model should be able to perform lies the causal one. Therefore, several benchmarks targeting causality have emerged recently [6, 7, 8].

A popular evaluation paradigm for causal reasoning is Pairwise Causal Discovery (PCD), which aims to detect pairwise causal relations from observational data. In a PCD task a model must determine if a causal link exists between two events, along with the direction of causality [9, 10]. A common way to frame this task is to give two sentences as input to the model (i.e., *"Martina has less chances of getting the flu"* and *"Martina has been vaccinated against the flu"*), and to ask the model if the first sentence is a consequence of the first with a yes/no question (in this case, groundtruth: *"yes"*) [1, 10].

Temporality plays a crucial role in the context of causality, as every causal relation inherently implies a temporal one: If an event A causes an event B, A must necessarily occur (or begin to occur) before B. Conversely, the presence of a temporal relation between two events does not necessarily imply a causal link. For this reason, we extended the PCD task to include the identification of temporal relations, to explicitly disentangle the interplay between causality and temporal sequencing.

To address this issue, in previous works we introduced the ExpliCa (**Expli**cit **Ca**usality) benchmark [1], offering a more controlled experimental setup that jointly addresses temporal and causal reasoning. ExpliCa presents pairs of sentences, each describing a distinct event, without any surface-level linguistic cues for temporal and causal relation, except for a *connective* that explicitly encodes both the type of relation (i.e., causal and temporal), and the order between the two events. For example, in [1], we asked the models to choose which of four connectives (*so*, *because*, *then*, and *after*) best represents the relation between the sentences *"Martina has less chances of getting the flu"* and *"Martina has been vaccinated against the flu"* (in this case, groundtruth: *"because"*.)

Despite these progresses, resources for joint temporal-causal reasoning are still lacking, especially

CLiC-it 2025: Eleventh Italian Conference on Computational Linguistics, September 24 — 26, 2025, Cagliari, Italy
*Corresponding author.
alessandro.bondielli@unipi.it (A. Bondielli); martina.miliani@fileli.unipi.it (M. Miliani); l.paglione1@studenti.unipi.it (L. Paglione); serena.auriemma@phd.unipi.it (S. Auriemma); lucia.passaro@unipi.it (L. Passaro); alessandro.lenci@unipi.it (A. Lenci)
0000-0003-3426-6643 (A. Bondielli); 0000-0003-1124-9955 (M. Miliani); 0009-0006-6846-5826 (S. Auriemma); 0000-0003-4934-5344 (L. Passaro); 0000-0001-5790-43086 (A. Lenci)

in languages other than English. At the same time, a rich ecosystem of LLMs pre-trained on, or adapted to, languages other than English, including Italian, is rapidly emerging.

To partially fill this gap, we introduce **Expli**CITA (**Expli**cit **C**ausality in **ITA**lian). ExpliCITA is an Italian adaptation of ExpliCa and we believe it is the first benchmark dedicated to joint temporal and causal reasoning in Italian.

We also leverage the evaluation framework for ExpliCa to conduct the first large-scale evaluation of Italian language models on causal reasoning. The framework allows us to test both **competence** (what the model "knows" about the probability distribution of linguistic events) via perplexity, and **performance** (how it applies that "knowledge") via prompting [11, 12]. Specifically, the prompting task is formulated as a multiple-choice task, where models have to select the appropriate connective in a cloze-style prompt. We explore different generation settings: greedy decoding and the Outlines framework [13], under both zero- and few-shot regimes. Our evaluation includes a total 20 models across a spectrum of several sizes and training approaches: i.) seven native Italian models trained from scratch, ii.) four multilingual models fine-tuned on Italian, iii.) three open-weights multilingual models, iv.) an open-weight reasoning-specialized LLM, and v.) five commercial systems from the GPT family.

We make both the data and code available on GitHub to replicate our experiments.[1]

Our contribution is twofold:

- we present ExpliCITA, the first dataset for joint temporal-causal reasoning in Italian;
- we deliver an extensive empirical study across 20 Italian and multilingual models, following a robust evaluation framework combining an evaluation via perplexity with multiple-choice prompting in several settings. This allows us to highlight strengths, weaknesses, and performance variation across model types and sizes.

The remainder of the paper is organized as follows: Section 2 reviews related work; Section 3 introduces the ExpliCITA dataset; Section 4 details the experimental setup; and Section 5 presents and discusses the results.

2. Related Works

The study of causality and its linguistic expressions has recently regained momentum, particularly in the context of evaluating the reasoning capabilities of large language models (LLMs). In this domain, many evaluation datasets focus on presenting a contextual scenario to test causal inference [14, 6, 15, 16, 17, 18], while others challenge NLP systems to identify causal relations directly on the text [19, 16, 20], also along with temporal ones [21, 22, 23, 24, 25]. **Expli**CITA stems from ExpliCa [1], a dataset developed to evaluate the ability of LLMs to detect explicit causal and temporal relations between events. In ExpliCa, relations are annotated via crowdsourcing and are signaled exclusively through a connective linking a pair of sentences, carefully stripped of any additional contextual or lexical cues. This controlled setup minimizes the influence of surrounding context and enables a more focused assessment of the model's reasoning on explicit relational cues.

Due to its design, ExpliCITA shares its structure with other datasets that frame implicit causal relations in a sentence-pair format, where each sentence expresses an individual event. Notable among these are the COPA dataset [26], the e-CARE dataset [27], and tasks from the BIG-Bench benchmark [28], which also test models on explicit causal reasoning. COPA and e-CARE were both incorporated into the original ExpliCa dataset.

While resources for English are abundant, the availability of non-English datasets for causal reasoning remains limited. Nevertheless, contributions exist for Spanish [29], German [30], Arabic [31], and Persian [32]. Among multilingual efforts, MECI [20] stands out as a resource where causal relations are annotated across several language editions of Wikipedia.

Causal reasoning, and related tasks such as Pairwise Causal Discovery (PCD), belongs to a broader class of inference-based tasks in natural language understanding. These tasks aim to evaluate a model's ability to derive implicit information from textual input, whether through logical entailment, causal attribution, or commonsense associations. Within this wider inference landscape, Natural Language Inference (NLI) benchmarks like XNLI [33] test models on cross-lingual entailment across 15 languages, while datasets such as X-CSQA [34] focus on cross-lingual commonsense reasoning in a question-answering format.

In the Italian context[35], the first dataset for textual entailment was introduced during the EVALITA 2009 evaluation campaign, comprising 800 sentence pairs derived from Wikipedia revision histories [36]. More recently, the HellaSwag-it dataset, an adaptation of the original HellaSwag dataset [37], was developed to test commonsense inference by asking models to choose the most plausible ending to a given scenario. Additionally, for causal reasoning, the COPA dataset was translated into Italian (and other languages) as part of the XCOPA project [38]. Both XCOPA-it and HellaSwag-it were integrated into ItaEval [39], a benchmark for evaluating LLMs on Italian commonsense and factual reasoning. ItaEval was featured in the 2024 Italian NLP evaluation

[1] https://github.com/Unipisa/ExpliCITA

campaign, CALAMITA [40], which included a wide range of datasets to test commonsense and factual knowledge. Among them, Gita [41] is particularly relevant here: it focuses on physical commonsense in Italian, presenting pairs of plausible and implausible stories composed of sentence sequences. To the best of our knowledge, ExpliCITA is the first dataset specifically dedicated to evaluating explicit causal and temporal reasoning in a controlled setting for the Italian language.

3. The ExpliCITA Dataset

The ExpliCITA Dataset is a direct translation of ExpliCa [1]. The original dataset was designed as a benchmark for evaluating explicit causal reasoning in LLMs, with a particular focus on distinguishing causal relations from temporal ones, using the PCD task. A thorough description of the dataset and its properties is reported in [1]. In the following, we highlight some of its key aspects.

Approximately a third of the items in ExpliCa are based on other existing datasets [42, 28, 27]. The remaining two thirds are manually crafted. In total, 600 items are in the dataset. Each item of the dataset comprises a sentence pair S1 and S2, where each sentence describes an event.

The dataset has two key dimensions, namely the *type of relation* and the *order of presentation.* As for the type of relation, the items were selected by authors to be equally divided into three main subsets: i.) CAUSAL, where the relationship is causal, and possibly of temporal precedence; ii.) TEMPORAL, where the relation is only of temporal precedence, without causality; iii.) UNRELATED, that includes thematically related sentences that are neither causally nor temporally related. Potential biases in lexical elements are controlled for using Mutual Information between lexical elements of the sentence pairs. This is done to avoid having very different lexemes in the UNRELATED group with respect to the other groups. The differences in the association strengths between lexemes in the three groups are not statistically significant.

As for the order of presentation, it can be either ICONIC (in short form *Ic*), if the sequence of events expressed in the two sentences matches their chronological and/or logical-causal order (e.g., "S1 then S2"), or ANTI-ICONIC (in short form, *A-Ic*), if the sequence of events expressed in the two sentences is inverted compared to their chronological and/or logical-causal order (e.g., the effect is mentioned before the cause: "S2 because S1"). Note that, for each sentence pair, the dataset includes both the Iconic and Anti-Iconic order for a total of $600 \times 2 = 1,200$ items.

The type of relation and the order of presentation are expressed via one out of four *connectives*, that act as linguistic cues to explicitly signal the nature of the relationship. In the English version of the dataset, the connectives are: *so* (Causal, Iconic), *because* (Causal, Anti-Iconic), *then* (Temporal, Iconic), and *after* (Temporal, Anti-Iconic).

A defining feature of the dataset is that **the connective is the sole linguistic cue** indicating the semantic relation between sentence pairs. To ensure a controlled and challenging evaluation of causal reasoning, the dataset excludes any additional explicit marker, such as causal verbs, and removes anaphoric references by avoiding personal pronouns. This design compels models to rely exclusively on event semantics and the connective itself, without support from broader contextual cues.

The dataset was then annotated via crowdsourcing by English native speakers. Specifically, annotators were asked to rate the acceptability of a sentence pair linked by one of the connectives. Each sentence pair, in both orders, with all possible connectives ($600 \times 2 \times 4 = 4800$ total items) was rated by 15 participants. For each sentence pair in both orders of presentation, the connective with the highest acceptability rating was considered as the ground truth. Note that the ground truth based on human ratings do not overlap perfectly with the original distinction in CAUSAL, TEMPORAL, and UNRELATED groups made by authors when building the sentence pairs.

To build ExpliCITA from ExpliCa, we followed a semi-automatic translation procedure. First, we used ChatGPT via the web interface[2] to translate each sentence from the 600 pairs independently. Then, each sentence was manually evaluated to address errors in the automatic translation. Errors ranged from mistakes in gender assignment (e.g., "*Luca è stata [...]*") to completely missing idiomatic expressions (e.g., "Marco ran the red light", translated as "*Marco ha corso la luce rossa*" instead of "*Marco è passato col rosso*"). A significant number of translations needed manual verification. For ExpliCITA, we used the following four connectives:

Quindi - Indicates a causal relation in the iconic order. The event in S1 causes the event in S2.

Perché - Indicates a causal relation in the anti-iconic order. The event in S1 is caused by the event in S2.

E poi - Indicates a temporal relation in the iconic order. The event in S1 temporally precedes the event in S2.

Dopo che - Indicates a temporal relation in the anti-iconic order. The event in S1 follows the event in S2.

The choice of multi-token expression for the temporal connectives is due to the fact that no sufficiently frequent single word in Italian conveys the proper meaning.

ExpliCITA includes each sentence pair in both orders of presentation. Thus, the number of data points is $600 \times 2 = 1,200$. We consider as our ground truth the results of the crowdsourcing experiment for ExpliCa [1]. In

[2]Accessed on December 2024

Connective \ Group	CAUSAL	TEMPORAL	UNRELATED	Total
Quindi (Caus., Ic)	181	15	66	262
Perché (Caus., A-Ic)	183	33	72	288
E poi (Temp., Ic)	17	207	180	404
Dopo che (Temp., A-Ic)	19	145	82	246

Table 1
Distribution of connectives across groups in ExpliCITA.

Table 1 we report statistics on the dataset. We consider both the original division in the three groups (CAUSAL, TEMPORAL, UNRELATED) and the numerosity of each connective, both in the three groups and globally.

4. Experimental Setting

The goal of our experiments is to test LLMs on the PCD task of the ExpliCITA dataset from two perspectives. On the one hand, we want to assess the linguistic **competence** of the model: the fact that it encodes some linguistic knowledge about causal and temporal relations. We do so by leveraging a **perplexity-based evaluation**. On the other hand, we want to address the actual **performance** of the model on the dataset. We do so via a **prompt-based evaluation** in which the model has to solve our PCD task, by identifying the correct connective for a sentence pair. Our main goal is to evaluate Italian LLMs on Italian data. In addition to native Italian LLMs, we also consider other model classes. Specifically, we account for i.) Italian fine-tuned models, i.e. open-weights models fine-tuned on Italian, ii.) open-weights multilingual models, iii.) open-weights reasoning models, and iv.) closed commercial models. All tested models are listed in Section 4.1.

Perplexity-Based Evaluation. This experiment is an exact replica of the one conducted in [1]. For each sentence pair in the dataset (i.e., in both orders of presentation), we derive one sentence for each connective, in the form "`S1 {{ connective }} S2`". We obtain $1,200 \times 4 = 4,800$ sentences in total. For each of them, we compute a model's perplexity (PPL) over the whole sentence. We then rank the four sentences based on PPL, and consider the one with the lowest value as the "model connective choice". Finally, we compute the accuracy of the model choices against the ground truth. We call this metric **Accuracy on Perplexity Score (APS)**.

Prompt-Based Evaluation. For the prompt-based evaluation, we asked the models to identify the correct connective to use between S1 and S2. We chose to focus on a standard multiple-choice task, as it is one of the most widely used formats for evaluating LLMs, and replicates one of the prompting experiments in [1]. In the task, the model is presented with S1 and S2 and a list of choices, each representing a connective. The task is to provide the correct choice. We experiment in both zero-shot and few-shot scenarios. For the few-shot, the models saw one example for each connective, for a total of four examples. To avoid biases in the choices, both the order of options to choose from and the position of the correct answer is randomized. Note however that all models saw the same exact prompt for any item in the dataset. We use accuracy as our main metric. To distinguish from APS, we refer to values obtained via prompting as **Accuracy on Prompt Execution (APX)**.

The template for the prompt is shown in Appendix A. We used the Jinja template syntax.[3] The prompt is not a direct translation but it is heavily inspired to the one used in [1]. First, we provide the models with the description and format of the task; for the few-shot scenario, we provide the examples; then, we give clear instructions for how to complete the task; finally, we describe the task. Since we use both pre-trained only and instruction fine-tuned models, we used a template that would enable also pre-trained only models to answer. Note that we did not implement specific templating strategies (e.g., chat formatting, special tokens, etc.) for any model, and we fed all the models with exactly the same prompt.

The only exception was GPT, which was prompted using the chat format, as required by the model's API. However, the content of the prompt was the same as the one used for all other models, without the addition of any custom system messages, special tokens, or instruction-specific formatting.

We used a markdown-like syntax to highlight the sections of the prompt. We acknowledge that not formatting the prompt for each model may hinder performances in some cases. However, we argue that this ensure a more fair evaluation. The only exception was made for the reasoning model, for which we also include the `<think>` token at the end of the prompt, to ensure that the Chain-of-thought is started.

We used a greedy decoding strategy for all experiments, that is we always sample the next most likely token at each generation step. We let each model generate a maximum of 20 tokens in their response. For the reasoning model, we let it generate a maximum of 10,000 tokens. All models, with the exception of GPT variants, where used in their HuggingFace implementation.[4]

A notable issue with unconstrained text generation is that less performing models may yield text that do not conform to the standard asked for in the prompt. This remains true also for cases, like ours, where the expected answer can be the direct continuation of the prompt, rather than the answer to a question or the turn in a

[3] https://jinja.palletsprojects.com
[4] huggingface.co

conversation. To alleviate this issue, we proceeded in two ways. First, we implemented a post-processing strategy based on a set of regular expressions to parse each model response and extract one answer. The regexes were designed to extract one and only one option from the generated text. In cases where multiple answers or no answer were detected, it was counted as a mistake for the model. In Section 5, we report the results of the model after this post-processing. Some models consistently failed to provide appropriate answers in this setting.

Second, we employed **Outlines** [13],[5] a Python library built to provide structured text generation with LLMs (e.g., with type constraints, following regular expressions, or providing json-formatted outputs). In the case of multiple choices, it uses masking on the output probabilities to restrict the model outputs to a set of valid completions [13]. In our case, the possible completions are the "A", "B", "C", and "D" options for the tasks. This approach has become quite popular in the literature and has been adopted in several recent studies on generative LLMs [12]. Note that Outlines was not used for the GPT variants and one of the open-weights tested models, namely Gemma3. In fact, all GPT models consistently yielded properly formatted outputs, making an additional evaluation redundant (recall that the next-token prediction is performed in a greedy fashion) and economically costly. Moreover, a known bug in the current Outlines and HuggingFace implementations prevents all Gemma3 models to be run through Outlines at this stage.

4.1. Tested Models

We chose to experiment on a variety of models and model classes, to gain a broader and clearer picture of the problem. Our main goal was to evaluate native Italian LLMs on the PCD task. Thus, we considered the following native Italian model families/variants:

Minerva [43]. We considered all model sizes of the Minerva family (from 350M to 7B), including both the Instruction fine-tuned and pre-trained only ones.

Velvet [44]. We experimented with both available models, namely Velvet-2B and Velvet-14B.

We highlight that we were not able to ran experiments on the Italia-9B model due to issues with its loading via the HuggingFace library.

We also chose to experiment with non-native Italian models for a clear and fair comparison. These can be distinguished into four classes:

Italian Fine-Tuned models: This class includes `LLaMAntino-2-chat-7b-hf-UltraChat-ITA` [45], `LLaMAntino-3-ANITA-8B-Inst-DPO-ITA` [46] and `cerbero-7b` variants [47]. They are respectively fine tuned versions of LLaMA-2, LLaMA-3 and Mistral.

Open LLMs: We also evaluated the performances of strong contenders in the Open LLM space. To do so, we selected Meta's LLaMA-3.1-8B [48] and two versions of Google's Gemma3 [49], namely the 4B and 12B ones.

Reasoning LLMs: We also tested one reasoning model, namely `DeepSeek-R1-Distill-Llama-8B` [50], a distilled version of DeepSeek-R1 using LLaMA-3.1-8B. This allows us to explore how reasoning impact performances on our PCD task.

Commercial models: Finally, we tested the GPT-4x family as representative of commercial closed-source models. We evaluated both `gpt-4o` and `gpt-4o-mini` [51], and all the GPT-4.1 variants (`gpt-4.1`, `gpt-4.1-mini`, and `gpt-4.1-nano`) [52].

Depending on its size, each model required a time between 0.5 and 1 GPU hours to complete its run, that includes both the zero-shot and few-shot experiments, each consisting of: i.) generation with greedy decoding; ii.) generation with Outlines; and iii.) PPL scores computation. The `DeepSeek-R1-Distill-Llama-8B` model required around 10 GPU hours in total, due to its much higher demand for test-time compute. Experiments with the GPT-4x family were conducted using the official OpenAI Batch API.[6] The code for replicating the experiments is available on GitHub.

5. Results and Discussion

In this Section we present and discuss the results. We first look at the overall results based on Accuracy of models on the PCD task of ExpliCITA, in terms of both i.) linguistic competence with APS, and ii.) performance with APX in zero- and few- shot experiments, with and without Outlines. Then, we present additional results by considering two aspects. On the one hand, we look at the distribution of answers for each model, to highlight possible biases and failures in providing an answer. On the other hand, we look at per-class performances, to understand whether the tested LLMs show biases in modelling specific aspects of temporal and causal reasoning.

5.1. Overall Results

Our main findings for the evaluation of LLMs on ExpliCITA are summarised in Figure 1. The Figure shows the Accuracy of all tested models, in all scenarios. We divide the plot by model family, and sort each family by the model size.

[5] https://github.com/dottxt-ai/outlines

[6] https://platform.openai.com/docs/guides/batch

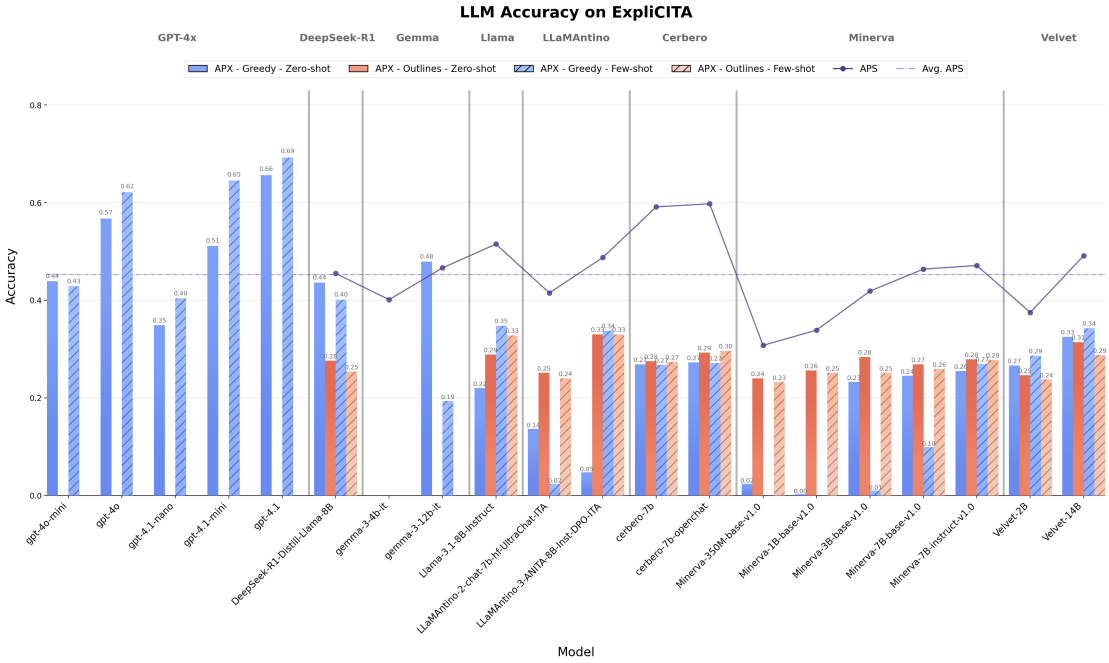

Figure 1: APS and APX scores for LLMs on ExpliCITA, grouped by model family and ordered by size.

The results are in line with the experiments reported for ExpliCa [1]. We highlight several interesting aspects in the following.

Overall performance. As for the raw performances, all models except the GPT-4x family show rather poor or at least somewhat brittle performances. The only models capable of approaching GPT-level performances are DeepSeek-R1 and Gemma3-12B. However, this is achieved either with the inclusion of reasoning for DeepSeek, or only in a specific setting for Gemma.

Zero- vs Few-Shot. As for the difference in zero-shot and few-shot settings, the GPT-4x family is again the only one where there is a clear and consistent trend, in this case in favour of the few-shot setting. In other cases, the few-shot examples are not always beneficial: for some models (e.g., Gemma3-12B, LLaMAntino-2 and Minerva-3B) it appears to be detrimental, while for other it is ineffective. However, for Minerva-7B we observe that while for the pre-trained variant the examples are detrimental, this is not true for the instruction-tuned one. This is possibly due to the instruction-tuning dataset of the model.

Impact of Outlines. It appears to be beneficial mostly for cases where zero- or few-shot performances are quite low (e.g., below 0.1). In other cases, the use of Outlines seems less influential. Nevertheless, the same accuracy may be obtained from a significantly different distribution of answers, as will be discussed in the following Sections.

Model sizes. As shown in [1], we observe that the size of the model is relevant for its downstream performances. In the open-weights model classes, the two best performing models are Gemma3 and Velvet, respectively in the 12B and 14B variants. Both also display above average APS scores. However, it is also interesting to note that while Gemma3-4B was not able to solve the task at all, the 2B variant of Velvet was consistent in its performance, which closely match those of some larger models.

Competence vs. Performance. It is important to notice that APS is always better than APX, with the sole exception of the `Gemma-3-12B` model. This further corroborates some of the findings in [1]: while models' internal representations and probability distribution encode, at least to some extent, knowledge about causal and temporal relations, this knowledge is not fully accessed via prompting. This is also in line with other research [11]. Moreover, it was shown in [1] that the gap between APS and APX shrinks with the size of the model. Given the wide array of tested open-weights model, we can further

corroborate this hypothesis by looking at Figure 2. We can clearly see that the rate of improvement in APX as models grow in size (red trendline) is higher than their respective rate of improvements in APS (blue trendline) on the task.

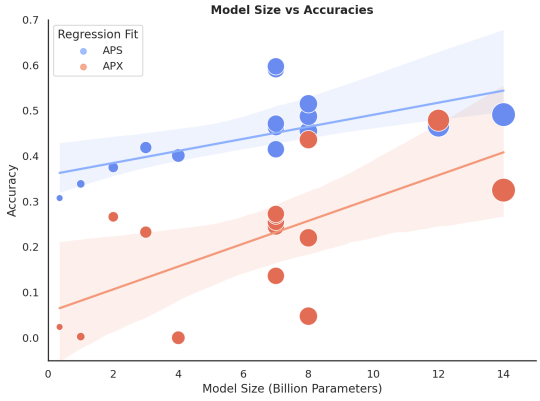

Figure 2: Difference between APS and APX across models of varying sizes.

We also highlight the following relevant findings associated to specific model classes:

Italian Models are weak; Native Italian pre-training is not beneficial. Native Italian models do not show relevant improvements with respect to fine-tuned alternatives, neither at the same size, nor at larger sizes. The Velvet family appears to provide relatively solid results at all scales; in contrast, smaller models in the Minerva family appear to be less robust on ExpliCITA. The fine-tuned Italian models display similar, if not better, performances than native ones. This could lead us to question whether it's truly necessary to train LLMs from scratch on Italian data. Results suggest that, albeit limited to this case study, it is not.

GPTs struggle. On ExpliCa, the GPT model family displayed performances that couldn't reach 0.8 Accuracy [1]. Changing the language of the dataset and the prompt highlight a stark contrast: the drop in performances for the same model is around .20 points, and even newer models cannot reach a 70% accuracy. Considering the fact that the task has remained exactly the same, and that GPT "speaks" fluent Italian, this may be indication that current LLMs are still limited in terms of actual causal reasoning, and still reliant on their internal probabilistic representations of texts.

Test-time compute is beneficial. We observed that the performances of the distilled DeepSeek-R1 drastically improve when it is allowed to use its "reasoning" abilities. This is particularly interesting, as it somewhat contrasts with the expectation that the task not require particular forms of reasoning, which may be instead required when modelling phenomena such as *implicit* causal relations. This issue will be further addressed in future works. We also note that while answers were provided in Italian, the chain-of-thought enclosed in the `<think>` tokens is almost exclusively in English.

5.2. Additional Analyses

Besides evaluating the accuracy of models on ExpliCITA, we also consider two other aspects that allow us to further understand the behaviour of the tested models in our setting.

Distribution of Answers. First, we explore how models actually answered to the multiple-choice task. The distribution of answers with greedy decoding and with outlines in the zero-shot setting is shown in Figure 3. We leave out the visualization of the few-shot setting due to space limitations, but they are very similar in nature.

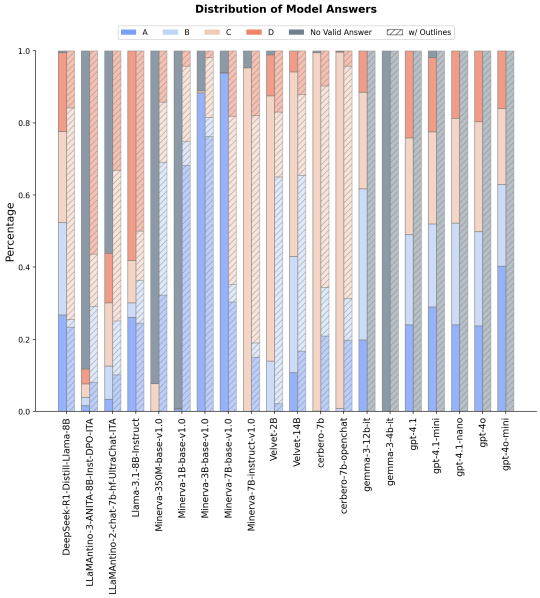

Figure 3: Model answer distribution in zero-shot multiple-choice tasks using greedy decoding and prompting via outlines.

We observe that some models consistently fail to provide an adequate answer, thus drastically lowering their performances. For example, it is possible that when ANITA actually answered it did so correctly, but it was able to answer on a very small fraction of the questions. Moreover, although we applied post-processing to the model responses (see Sec. 4), we still observed persistent

failure modes, primarily due to the model's inability to follow the expected output format. Such behaviors can be broadly described as faithful hallucinations caused by instructional inconsistency [53], in which the model's output is not properly aligned with the user's request. These failures often consisted in limitations in the number of requested output tokens, which the models were unable to respect, unintended rewriting of the input question, or, more generally, a lack of adherence to the structure and intent of the prompt.

We also observe that several models have a strong preference for a specific answer, which is often either "A" or "C". This is in line with research on biases in multiple choice tasks [54]. This is corroborated by the fact that, even with Outlines, these models still tend to prefer a specific answer over the others.

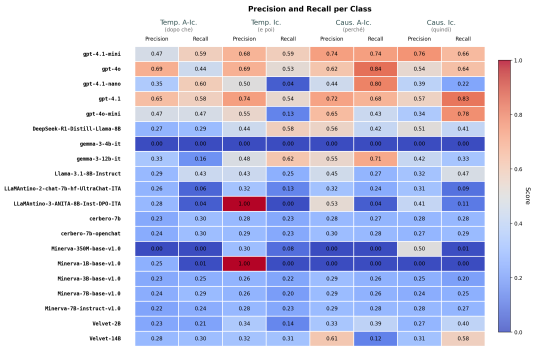

Figure 4: Per-class Precision and Recall for each model in the zero-shot setting.

Per-class Performances. Finally, Figure 4 shows the Precision and Recall performances of each model, divided by class. Again, we look at the zero-shot scenario and leave out the few-shot one due to space limitations. By looking at the plot, three main observations can be made. First, the GPT-4x models are the most consistent across classes, with only a few notable exceptions for the smallest models. Second, we observe that some of the models display a relatively strong bias towards a single or a pair of answers. Finally, if we zoom out and look at the bigger picture, we see that models have a slight preference towards causal relationships. The less biased models are the two biggest ones, namely gpt-4.1 and gpt-4o. This may further suggest that at smaller scales models rely more on distributional properties of words (e.g., causal connectives often imply a temporal relationship as well, but not vice versa) and are more sensitive to frequency effects linked to word combinations frequently encountered during training. In Italian, in fact, causal connectives such as *"perché"* or *"quindi"* are often used in syntactic constructions where the premise is explicitly connected to the consequence via one of these two connectives. The construction "`S1 connective S2`" is therefore typical for causal relationships.

In contrast, there is greater variability in how temporal sequential relationships can be expressed in Italian. These can be conveyed through temporal conjunctions such as *"e poi"* or *"dopo che"*, as well as through adverbs and adverbial expressions such as *"precedentemente"* ("previously"), *"successivamente"* ("subsequently"), or *"poco fa"* ("a short while ago"). Equally frequent are cases in which temporal relations are conveyed solely through verb tense agreement between the two clauses, for instance, through a past–present combination to express anteriority between S1 and S2. Compared to causal relationships, the temporal dimension is thus more susceptible to variability, both in terms of the range of constructions available to express the same temporal relation in Italian, and in terms of the diversity of contexts in which the same temporal adverb might occur.

Indeed, while causality pertains to a subset of verbs and situational contexts, temporal information, whether implicit or explicit, is present in all events expressed by a verb. This variability affects the generalization capabilities of the models, especially the smaller ones. In fact, larger models seem better able to properly evaluate the context and identify the correct relationship between events.

6. Conclusions and Future Works

In this paper, we presented the ExpliCITA dataset, the Italian translation of ExpliCa [1]. The dataset is designed to evaluate explicit temporal and causal reasoning in LLMs. We also replicated part of the experiments made on ExpliCa with several LLMs, including i.) natively-trained Italian models, ii.) multilingual models fine-tuned on Italian, iii.) multilingual open-weights models, iv.) a multilingual reasoning open open-weights model, and v.) closed-weights commercial models from OpenAI.

Our findings can be summarized as follows. First, consistently with [1], we observe two key facts. On the one hand, all tested models, including GPT, struggle to solve the task, in Italian more so than in English, both in the zero- and few-shot setting. We also see that this struggle is also due to their inability to reliably provide the answers required by the task, which is only partially alleviated by using the decoding method of Outlines. On the other hand, we observe that linguistic competence of models, measured with the APS, is consistently better than the respective performance when prompted. However, we see that this gap between APS and prompted accuracy tends to reduce with the model size.

Second, we observe that native Italian models are no better than the fine-tuned alternatives when it comes to

the ExpliCITA PCD task.

Third, we see that leveraging test-time compute appears to be beneficial for the task, possibly suggesting that the reasoning training is important to boost the ability to recognize semantic relations between events, even when these are linguistically expressed. We plan to conduct a more systematic investigation of the effects of both chain-of-thought reasoning and Outlines across different models and languages. This will include an in-depth error analysis aimed at understanding when and why such prompting strategies are effective, and whether their benefits depend on the structure of the prompt, the language used for reasoning (e.g., English vs. Italian), or the intrinsic capabilities of the models themselves.

Finally, we observe a slight improvement in managing the causal aspect of the relationship rather than the temporal one, highlighted by the per-class performances.

In the future, we plan to systematically compare the results obtained without chat-specific templating to those obtained by prompting each model using its native chat format. This will help better isolate the impact of instruction tuning and formatting on model performance. Furthermore, although a direct comparison with traditional NLP systems was beyond the scope of this work, future research could explore whether LLMs provide a competitive advantage in explicit causal reasoning (i.e., without task-specific training) compared to lightweight, specialized models. Finally, as part of future work, we plan to experiment with implicit causality as well. We also aim to further explore the impact of reasoning and test-time-compute on the performance of models on both explicit and implicit causal relations.

Acknowledgments

This work has been supported by the PNRR MUR project PE0000013-FAIR (Spoke 1), funded by the European Commission under the NextGeneration EU programme, and the EU EIC project EMERGE (Grant No. 101070918).

References

[1] M. Miliani, S. Auriemma, A. Bondielli, E. Chersoni, L. Passaro, I. Sucameli, A. Lenci, ExpliCa: Evaluating explicit causal reasoning in large language models, in: W. Che, J. Nabende, E. Shutova, M. T. Pilehvar (Eds.), Findings of the Association for Computational Linguistics: ACL 2025, Association for Computational Linguistics, Vienna, Austria, 2025, pp. 17335–17355. URL: https://aclanthology.org/2025.findings-acl.891/. doi:10.18653/v1/2025.findings-acl.891.

[2] J. Pearl, Causality, Cambridge University Press, New York, NY, USA, 2009.

[3] A. Lenci, Understanding natural language understanding systems, Sistemi intelligenti 35 (2023) 277–302.

[4] K. Mahowald, A. A. Ivanova, I. A. Blank, N. Kanwisher, J. B. Tenenbaum, E. Fedorenko, Dissociating language and thought in large language models, Trends in cognitive sciences (2024).

[5] E. Pavlick, Symbols and grounding in large language models, Philosophical Transactions of the Royal Society A 381 (2023) 20220041.

[6] Z. Wang, Causalbench: A comprehensive benchmark for evaluating causal reasoning capabilities of large language models, in: Proceedings of the 10th SIGHAN Workshop on Chinese Language Processing (SIGHAN-10), 2024, pp. 143–151.

[7] S. Chen, B. Peng, M. Chen, R. Wang, M. Xu, X. Zeng, R. Zhao, S. Zhao, Y. Qiao, C. Lu, Causal evaluation of language models, arXiv preprint arXiv:2405.00622 (2024).

[8] D. Dalal, P. Buitelaar, M. Arcan, Calm-bench: A multi-task benchmark for evaluating causality-aware language models, in: Findings of the Association for Computational Linguistics: EACL 2023, 2023, pp. 296–311.

[9] J. Gao, X. Ding, B. Qin, T. Liu, Is chatgpt a good causal reasoner? a comprehensive evaluation, in: Findings of the Association for Computational Linguistics: EMNLP 2023, 2023, pp. 11111–11126.

[10] G. Wan, Y. Wu, M. Hu, Z. Chu, S. Li, Bridging causal discovery and large language models: A comprehensive survey of integrative approaches and future directions, arXiv preprint arXiv:2402.11068 (2024).

[11] J. Hu, R. Levy, Prompting is not a substitute for probability measurements in large language models, in: Proceedings of EMNLP, 2023.

[12] C. Kauf, E. Chersoni, A. Lenci, E. Fedorenko, A. A. Ivanova, Log probabilities are a reliable estimate of semantic plausibility in base and instruction-tuned language models, in: Proceedings of the EMNLP BlackBoxNLP Workshop on Analysing and Interpreting Neural Networks, 2024.

[13] B. T. Willard, R. Louf, Efficient guided generation for llms, arXiv preprint arXiv:2307.09702 (2023).

[14] Z. Jin, Y. Chen, F. Leeb, L. Gresele, O. Kamal, L. Zhiheng, K. Blin, F. G. Adauto, M. Kleiman-Weiner, M. Sachan, et al., Cladder: Assessing causal reasoning in language models, in: Thirty-seventh conference on neural information processing systems, 2023.

[15] S. Ashwani, K. Hegde, N. R. Mannuru, M. Jindal, D. S. Sengar, K. C. R. Kathala, D. Banga, V. Jain, A. Chadha, Cause and effect: Can large language models truly understand causality?, arXiv preprint arXiv:2402.18139 (2024).

[16] H. Chi, H. Li, W. Yang, F. Liu, L. Lan, X. Ren, T. Liu,

B. Han, Unveiling causal reasoning in large language models: Reality or mirage?, Advances in Neural Information Processing Systems 37 (2024) 96640–96670.

[17] D. Mariko, H. Abi Akl, K. Trottier, M. El-Haj, The financial causality extraction shared task (fincausal 2022), in: Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022, 2022, pp. 105–107.

[18] A. Romanou, S. Montariol, D. Paul, L. Laugier, K. Aberer, A. Bosselut, Crab: Assessing the strength of causal relationships between real-world events, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023, pp. 15198–15216.

[19] P. Hosseini, D. A. Broniatowski, M. Diab, Predicting directionality in causal relations in text, arXiv preprint arXiv:2103.13606 (2021).

[20] V. D. Lai, A. P. B. Veyseh, M. Van Nguyen, F. Dernoncourt, T. H. Nguyen, Meci: A multilingual dataset for event causality identification, in: Proceedings of the 29th international conference on computational linguistics, 2022, pp. 2346–2356.

[21] J. Dunietz, L. Levin, J. G. Carbonell, The because corpus 2.0: Annotating causality and overlapping relations, in: Proceedings of the 11th Linguistic Annotation Workshop, 2017, pp. 95–104.

[22] Q. Ning, Z. Feng, H. Wu, D. Roth, Joint reasoning for temporal and causal relations, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2018, pp. 2278–2288.

[23] P. Mirza, R. Sprugnoli, S. Tonelli, M. Speranza, Annotating causality in the tempeval-3 corpus, in: Proceedings of the EACL 2014 Workshop on Computational Approaches to Causality in Language (CAtoCL), 2014, pp. 10–19.

[24] T. Caselli, P. Vossen, The event storyline corpus: A new benchmark for causal and temporal relation extraction, in: Proceedings of the Events and Stories in the News Workshop, 2017, pp. 77–86.

[25] N. Mostafazadeh, A. Grealish, N. Chambers, J. Allen, L. Vanderwende, Caters: Causal and temporal relation scheme for semantic annotation of event structures, in: Proceedings of the Fourth Workshop on Events, 2016, pp. 51–61.

[26] M. Roemmele, C. A. Bejan, A. S. Gordon, Choice of plausible alternatives: An evaluation of commonsense causal reasoning, in: 2011 AAAI spring symposium series, 2011.

[27] L. Du, X. Ding, K. Xiong, T. Liu, B. Qin, e-care: a new dataset for exploring explainable causal reasoning, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 432–446.

[28] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al., Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, arXiv preprint arXiv:2206.04615 (2022).

[29] J. R. Portela, N. Perez, R. Manrique, Esnlir: A spanish multi-genre dataset with causal relationships, arXiv preprint arXiv:2503.08803 (2025).

[30] I. Rehbein, J. Ruppenhofer, A new resource for german causal language, in: Proceedings of the Twelfth Language Resources and Evaluation Conference, 2020, pp. 5968–5977.

[31] J. Sadek, F. Meziane, Learning causality for arabic-proclitics, Procedia computer science 142 (2018) 141–149.

[32] Z. Rahimi, M. ShamsFard, Persian causality corpus (percause) and the causality detection benchmark, arXiv preprint arXiv:2106.14165 (2021).

[33] A. Conneau, G. Lample, R. Rinott, A. Williams, S. R. Bowman, H. Schwenk, V. Stoyanov, Xnli: Evaluating cross-lingual sentence representations, arXiv preprint arXiv:1809.05053 (2018).

[34] B. Y. Lin, S. Lee, X. Qiao, X. Ren, Common sense beyond english: Evaluating and improving multilingual language models for commonsense reasoning, arXiv preprint arXiv:2106.06937 (2021).

[35] L. C. Passaro, M. Di Maro, V. Basile, D. Croce, Lessons learned from evalita 2020 and thirteen years of evaluation of italian language technology, IJCoL. Italian Journal of Computational Linguistics 6 (2020) 79–102.

[36] J. Bos, F. M. Zanzotto, M. Pennacchiotti, Textual entailment at evalita 2009, Proceedings of EVALITA 2009 (2009) 2.

[37] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, Y. Choi, HellaSwag: Can a machine really finish your sentence?, 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. 4791–4800. URL: https://aclanthology.org/P19-1472/. doi:10.18653/v1/P19-1472.

[38] E. M. Ponti, G. Glavaš, O. Majewska, Q. Liu, I. Vulić, A. Korhonen, Xcopa: A multilingual dataset for causal commonsense reasoning, arXiv preprint arXiv:2005.00333 (2020).

[39] G. Attanasio, M. La Quatra, A. Santilli, B. Savoldi, et al., Itaeval: A calamita challenge, in: Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024), 2024.

[40] G. Attanasio, P. Basile, F. Borazio, D. Croce, M. Francis, J. Gili, E. Musacchio, M. Nissim, V. Patti, M. Rinaldi, et al., Calamita: Challenge the abilities of

language models in italian, in: Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024), Pisa, Italy, 2024.
[41] G. Pensa, B. Altuna, I. Gonzalez-Dios, A multilayered approach to physical commonsense understanding: Creation and evaluation of an italian dataset, in: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 819–831.
[42] L. D. Wanzare, A. Zarcone, S. Thater, M. Pinkal, A crowdsourced database of event sequence descriptions for the acquisition of high-quality script knowledge, in: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), 2016, pp. 3494–3501.
[43] R. Orlando, L. Moroni, P.-L. Huguet Cabot, S. Conia, E. Barba, S. Orlandini, G. Fiameni, R. Navigli, Minerva LLMs: The first family of large language models trained from scratch on Italian data, in: F. Dell'Orletta, A. Lenci, S. Montemagni, R. Sprugnoli (Eds.), Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024), CEUR Workshop Proceedings, Pisa, Italy, 2024, pp. 707–719. URL: https://aclanthology.org/2024.clicit-1.77/.
[44] A. Team, Almawave presents velvet: The sustainable and high-performance italian ai, 2025. URL: https://www.almawave.com.
[45] P. Basile, E. Musacchio, M. Polignano, L. Siciliani, G. Fiameni, G. Semeraro, Llamantino: Llama 2 models for effective text generation in italian language, 2023. arXiv:2312.09993.
[46] M. Polignano, P. Basile, G. Semeraro, Advanced natural-based interaction for the italian language: Llamantino-3-anita, 2024. arXiv:2405.07101.
[47] F. A. Galatolo, M. G. Cimino, Cerbero-7b: A leap forward in language-specific llms through enhanced chat corpus generation and evaluation, arXiv preprint arXiv:2311.15698 (2023).
[48] A. G. et al., The llama 3 herd of models, 2024. URL: https://arxiv.org/abs/2407.21783. arXiv:2407.21783.
[49] G. Team, Gemma 3 technical report, 2025. URL: https://arxiv.org/abs/2503.19786. arXiv:2503.19786.
[50] DeepSeek-AI, Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL: https://arxiv.org/abs/2501.12948. arXiv:2501.12948.
[51] OpenAI, Gpt-4o system card, 2024. URL: https://arxiv.org/abs/2410.21276. arXiv:2410.21276.
[52] O. AI, Introducing gpt-4.1 in the api, 2025. URL: https://openai.com/index/gpt-4-1/.
[53] A. Saxena, P. Bhattacharyya, Hallucination detection in machine generated text: A survey (2024).
[54] C. Zheng, H. Zhou, F. Meng, J. Zhou, M. Huang, Large language models are not robust multiple choice selectors, in: The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024, OpenReview.net, 2024. URL: https://openreview.net/forum?id=shr9PXz7T0.

A. Prompt template

An example of the ExpliCITA PCD task, framed as a multiple-choice prompting task, is provided in the box below.

Multiple-choice Prompt

```
# Compito di scelta multipla

## Descrizione del Compito
Ti sara' fornito un compito. Avrai a disposizione due frasi, Frase 1
e Frase 2, e una lista di parole connettivo. Il tuo compito e' quello
di scegliere dalla lista di parole connettivo la parola piu'
appropriata per collegare le due frasi in maniera logica e coerente.
La parola scelta dovrebbe essere grammaticalmente e contestualmente
corretta. Per scegliere la parola devi scrivere la lettera
corrispondente alla parola scelta nel campo risposta.

## Formato del Compito
Frase 1: [Frase 1]
Frase 2: [Frase 2]

Opzioni:
A. [parola A]
B. [parola B]
C. [parola C]
D. [parola D]

Risposta: [Lettera dell'opzione corrispondente alla parola corretta]

{% if examples %}
## Esempi
{% for example in examples %}
### Esempio
Frase 1: {{ example.S1 }}
Frase 2: {{ example.S2 }}

Opzioni:
A. {{ example.option_A }}
B. {{ example.option_B }}
C. {{ example.option_C }}
D. {{ example.option_D }}

Risposta: {{ example.correct_answer }}
{% endfor %}
{% endif %}

## Istruzioni del Compito
1. Leggi attentamente la Frase 1 e la Frase 2;
2. Esamina l'elenco delle parole fornite;
3. Seleziona l'opzione corrispondente alla parola che meglio collega
le due frasi, nell'ordine in cui ti sono fornite, in maniera logica
e coerente. ATTENZIONE: scrivi nel campo "Risposta" **SOLO** la
lettera dell'opzione (A, B, C, o D) corrispondente alla parola che
meglio collega le due frasi nel campo risposta, ad esempio
"Risposta: C".

## Compito:
Frase 1: {{ sentence_a }}
Frase 2: {{ sentence_b }}

Opzioni:
{{ options }}

Risposta:
```

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT (OpenAI) and Grammarly in order to: Paraphrase and reword, Improve writing style, and Grammar and spelling check. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.