The Clinical Trial Duration Prediction Dataset: A Resource for AI Research
Supplemental material 1. Prompts

A. Prompt to extract time frame

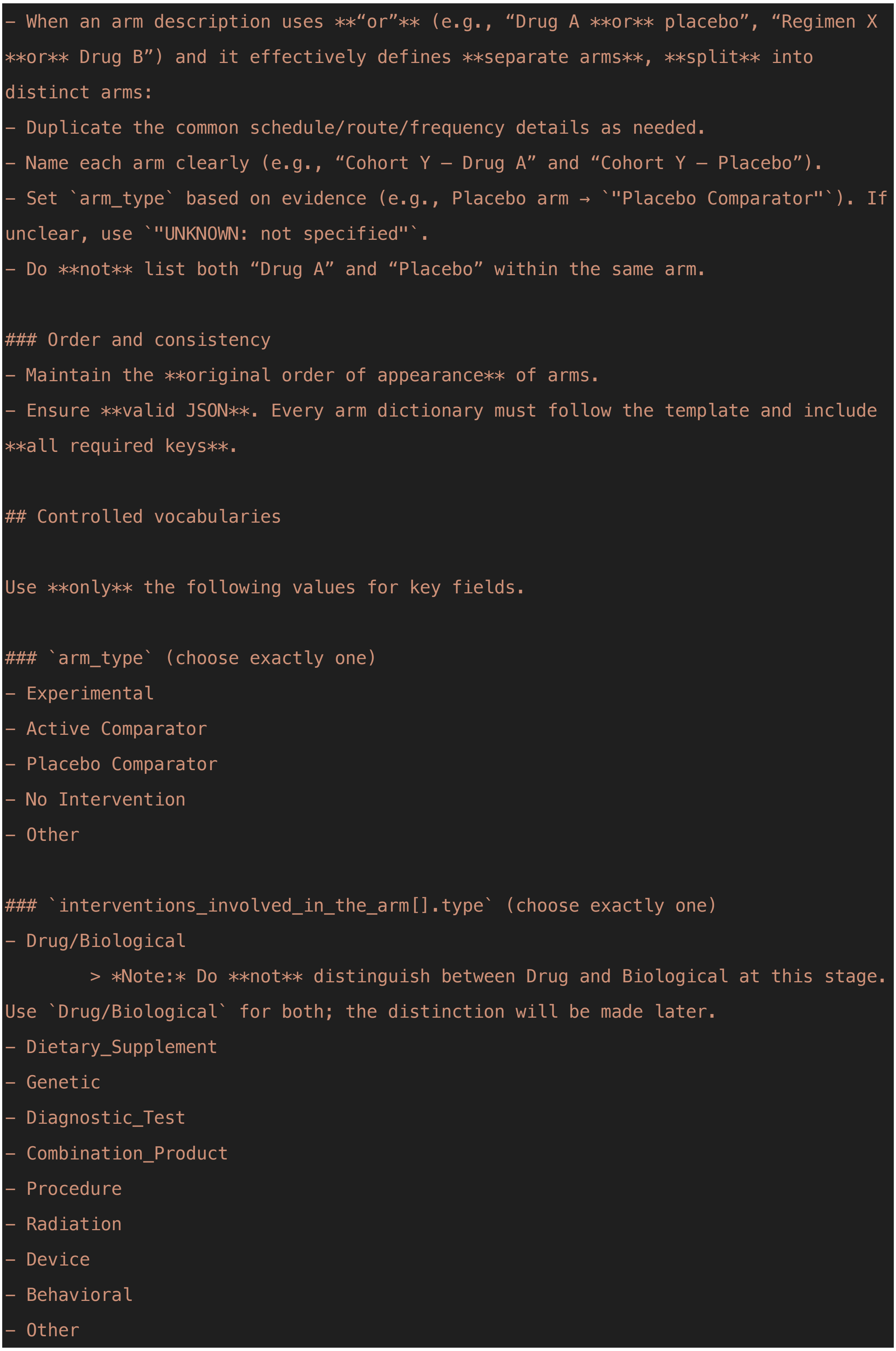
```
You are a medical expert. Your task is to extract the time duration from the
following text. The time duration can be in various formats, such as "3 months", "6
weeks", "1 year", etc. If the time duration is not mentioned, return None.
The time duration should be in the format of list with 2 items in a JSON style.
The first item is the number, and the second item is the singular unit. Even if the
time duration is mentioned in plural form, return the singular form.
Make sure that the unit is one of the following: "year", "month", "week", "day",
"hour", "minute", "second".
Be careful that the time duration can be mentioned in various forms, not only
"number units" but also "unit number" such as "Day 3", "Month 6", or their
abbreviations like "D3" or "M6".
If multiple time durations are mentioned, return the latest one. Be careful, as
sometimes the latest one is not the last one mentioned. Check the units.
If the time duration is not mentioned, return a null list, [].
Make sure not to return any other text, just the JSON list.
The text is as follows:
```

Following this prompt, we input each value in the time_frame.

B. Prompt to restructure arm-intervention relationship

```
# Task
You are given clinical trial text describing treatment arms.
Extract each arm and structure it strictly as a JSON array of dictionaries
following the template below.
Output **JSON only** (no extra text or explanation).

## Rules

### Identify true interventions
- Determine true interventions primarily from the **arm name** and **detailed
description**.
- Treat **arm_type** (e.g., Experimental, Active Comparator) only as supporting
evidence.
```

```
- If the arm type conflicts with the description or name, do **not** force a
decision. Instead, indicate uncertainty, e.g., `"arm_type": "UNKNOWN: conflict
between arm type and description"`.

### Indicate uncertainty
- For any missing or conflicting data (**name, dose, frequency, route, schedule,
duration**), write `"UNKNOWN: <reason>"` (e.g., `"UNKNOWN: not specified"`,
`"UNKNOWN: conflict between sources"`).

### Expand regimen names into individual drugs
- If an intervention is a **regimen** (e.g., R-CHOP), list **each component drug**
as a separate intervention inside `interventions_involved_in_the_arm`.
- Use drug names explicitly mentioned in the text whenever possible.
- If a regimen’s standard composition is well-known but not fully spelled out, you
may expand it from general knowledge, while keeping any missing
**dose/frequency/route/schedule/duration** as `"UNKNOWN: not specified"`.
- If a regimen is given **together with** an additional drug (e.g., Drug A **in
combination with** Regimen B), list **Drug A** and **each of Regimen B’s
components** separately.

### Handle placebos carefully
- Include placebo if it is **clearly described** as part of the true treatment
(e.g., Placebo Comparator, double-placebo).
- If placebo is mentioned only for masking and you are confident it is **not** part
of the true treatment, **omit** it.
- If confidence is low, include it and mark uncertainty, e.g., `"UNKNOWN: whether
placebo is true intervention"`.
- Phrases like **“A matching placebo”, “placebo matching A”** are treated as
placebo interventions; you may note matched details in the `schedule` field.

### Non-drug interventions
- For behavioral or other non-drug interventions, use the appropriate type (e.g.,
`"Behavioral"`).
- If route is not applicable, write `"UNKNOWN: not applicable (behavioral)"`.

### Arm splitting when “or” implies multiple arms
```



```
- When an arm description uses **“or”** (e.g., “Drug A **or** placebo”, “Regimen X **or** Drug B”) and it effectively defines **separate arms**, **split** into distinct arms:
- Duplicate the common schedule/route/frequency details as needed.
- Name each arm clearly (e.g., “Cohort Y – Drug A” and “Cohort Y – Placebo”).
- Set `arm_type` based on evidence (e.g., Placebo arm → `"Placebo Comparator"`). If unclear, use `"UNKNOWN: not specified"`.
- Do **not** list both “Drug A” and “Placebo” within the same arm.

### Order and consistency
- Maintain the **original order of appearance** of arms.
- Ensure **valid JSON**. Every arm dictionary must follow the template and include **all required keys**.

## Controlled vocabularies

Use **only** the following values for key fields.

### `arm_type` (choose exactly one)
- Experimental
- Active Comparator
- Placebo Comparator
- No Intervention
- Other

### `interventions_involved_in_the_arm[].type` (choose exactly one)
- Drug/Biological
        > *Note:* Do **not** distinguish between Drug and Biological at this stage. Use `Drug/Biological` for both; the distinction will be made later.
- Dietary_Supplement
- Genetic
- Diagnostic_Test
- Combination_Product
- Procedure
- Radiation
- Device
- Behavioral
- Other
```

~~~

If uncertainty remains about the appropriate category, pick the closest match and document the uncertainty (e.g., in an `UNKNOWN:` note).

Output template
```json
[
{
    "arm_type": "",
    "arm_name": "",
    "interventions_involved_in_the_arm": [
    {
        "name": "",
        "type": "",
        "dose": "",
        "frequency": "",
        "route": "",
        "schedule": "",
        "duration": ""
    }
    ]
}
]
```
~~~

Following this prompt, we input JSON-dumped strings for each value in the arms field. After restructuring, we attempt to map each extracted intervention name to a SMILES string registered in the PubChem database.

C. Prompt to detect negative controls

```
# task
You are a medical expert. Your task is to determine whether the following intervention is a drug or biological.
Return 1 if it is a negative control, 0 otherwise.
# rules
- If the input means a negative control (e.g., placebo, sham, vehicle, etc.), return 0.
- Only return 1, or 0, without any explanation.
# examples
```

```
- "R-CHOP" -> 0
- "s1" -> 0
- "placebo" -> 1
- "aspirin" -> 0
- "vitamin C" -> 0
- "sham" -> 1
- "behavioral therapy" -> 0
- "exercise" -> 0
```

Following this prompt, we input each intervention for which a corresponding SMILES string could not be found.

Interventions that were not predicted as negative controls (0) were advanced to the next step: active ingredient extraction.

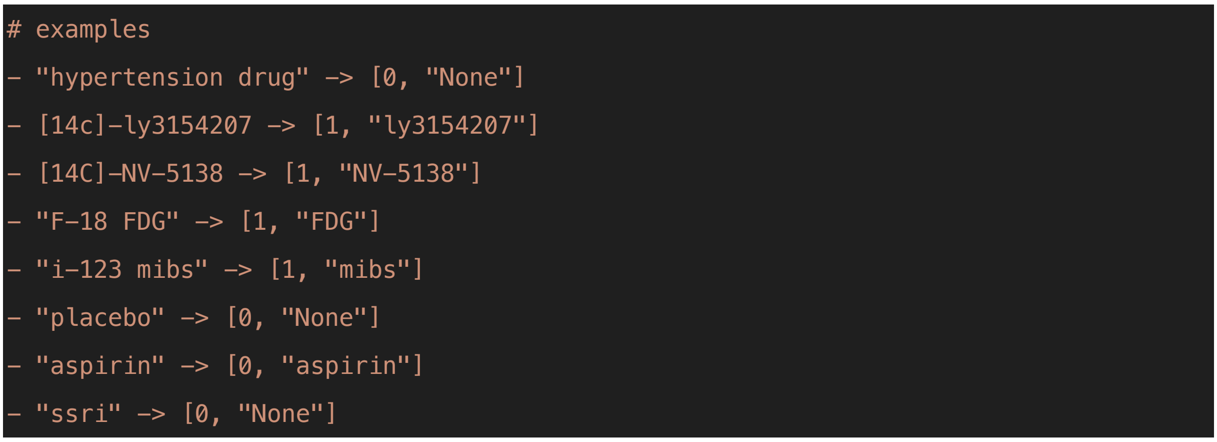
D. Prompt to extract active ingredient

```
# task
You are a medical expert. Your task is to extract the true intervention name from
the following text which should indicate only one intervention.
Return the intervention name only.
# rules
- If the input includes not only the name but also other information, return only
the name.
# examples
- nitric oxide by inhalation inomax -> nitric oxide
- surgical resection of tumor residue -> surgical resection of tumor residue
- finafloxacin i.v. solution 200 mg -> finafloxacin
- olt1177 gel -> olt1177
```

Following this prompt, we input each intervention name not predicted as negative controls. And we attempt to map them again.

E. Prompt to remove radionuclide

```
# task
You are a medical expert. Your task is to detect whether the givin intervention has
radioisotopes or not.
Return [1, <name without radioisotope>], [0, "None"] otherwise, without any
explanation.
# rules
- radioisotopes include: I-123, I-124, I-125, I-131, F-18, C-11, C-14, N-13, O-15,
Tc-99m, In-111, Ga-68, Y-90, Lu-177, etc.
```



```
# examples
- "hypertension drug" -> [0, "None"]
- [14c]-ly3154207 -> [1, "ly3154207"]
- [14C]-NV-5138 -> [1, "NV-5138"]
- "F-18 FDG" -> [1, "FDG"]
- "i-123 mibs" -> [1, "mibs"]
- "placebo" -> [0, "None"]
- "aspirin" -> [0, "aspirin"]
- "ssri" -> [0, "None"]
```

Following this prompt, we input each intervention for which a corresponding SMILES string could not be found in the previous step. And we attempt map them again.

Interventions that could not be mapped using the steps described above were treated as having no associated SMILES strings in CTDP.