Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 523–530
Marseille, 11–16 May 2020

PÂTÉ: A Corpus of Temporal Expressions for the In-car Voice Assistant Domain

Alessandra Zarcone[1], Touhidul Alam[2,3], Zahra Kolagar[2]
Fraunhofer IIS ([1]ALabs, [2]AME-S), [3]Universität Stuttgart (IMS)
Am Wolfsmantel 33, 91058 Erlangen, Germany, Pfaffenwaldring 5b, 70569 Stuttgart, Germany
{alessandra.zarcone, alamtl, kolagaza}@iis.fraunhofer.de

Abstract

The recognition and automatic annotation of temporal expressions (e.g. *Add an event for tomorrow evening at eight to my calendar*) is a key module for AI voice assistants, in order to allow them to interact with apps (for example, a calendar app). However, in the NLP literature, research on temporal expressions has focused mostly on data from the news, from the clinical domain, and from social media. The voice assistant domain is very different than the typical domains that have been the focus of work on temporal expression identification, thus requiring a dedicated data collection. We present a crowdsourcing method for eliciting natural-language commands containing temporal expressions for an AI voice assistant, by using pictures and scenario descriptions. We annotated the elicited commands (480) as well as the commands in the Snips dataset following the TimeML/TIMEX3 annotation guidelines, reaching a total of 1188 annotated commands. The commands can be later used to train the NLU components of an AI voice assistant.

Keywords: voice assistant, TIMEX3, TimeML, crowdsourcing

1. Introduction

Voice assistants are becoming pervasive in our lives. In order to support their users with different tasks (for example, booking a table at a restaurant for a specific time), they rely on so-called Natural Language Understanding (NLU) components, such as intent classification or slot filling (Tur and De Mori, 2011). NLU components map natural-language commands (e.g. *Call the restaurant*) to abstract representations of the user's intentions (e.g. Call(Restaurant)) by relying on usage patterns, in order to trigger actions (e.g. responses, queries on a knowledge base, operation of home automation devices)[1]. NLU components typically use supervised methods and rely on large-scale labeled domain-specific datasets for training and testing (Tur et al., 2018), which should reflect the way people address voice assistants (Fraser and Gilbert, 1991). In order to help users with everyday scheduling tasks, AI voice assistants need to identify and extract temporal information from temporal expressions (TEs, e.g. *tomorrow at eight*), for example to correctly set reminders, or help the user interact with a calendar.

The automatic identification, tagging and normalization of temporal expressions has been a widely researched topic in NLP (Pustejovsky et al., 2009; UzZaman et al., 2013), and has typically focused on large text documents, for example in the news domain (e.g. TimeBank and AQUAINT, both used as gold standard in TempEval-3) or in the clinical domain (Styler IV et al., 2014), with the aim to identify relations between events mentioned in the text and their chronological order. However, to our knowledge, no dataset with a rich annotation of TEs for TE identification in the voice assistant domain[2] is currently available. In the domain of voice assistants, some datasets such as Snips (Coucke et al., 2018) are available for benchmarking intent classification, but do not provide a rich annotation of temporal expressions beyond simply identifying time-related entities.

In this paper, we present PÂTÉ, a Personal Assistant dataset with Temporal Expressions, created by crowdsourcing 480 commands directed at an AI assistant. The method we used was specifically aimed at eliciting commands containing time expressions to be used for data-driven supervised machine learning approaches to temporal tagging. We conducted an annotation following the TimeML/TIMEX3 annotation both on the elicited data (480 commands) and on the Snips dataset (708 commands), reaching a total of 1188 annotated commands, 1050 of them containing TEs. The combined dataset (PÂTÉ + Snips) contains a total of 1714 TEs (see also Table 2). We show how this domain is very different than the typical domains that have been the focus of work on temporal expression identification, thus requiring a dedicated data collection.

2. TimeML/TIMEX3 annotation

TimeML provides a widely-adopted standard framework for annotating time, events, and event relations in text, including tags for several types of temporal information and expressions (TIMEX3 tags). We will focus only on the TIMEX3 annotation of time expressions, as the other data structures defined in TimeML (EVENT, SIGNAL and LINK) have a more limited relevance for calendar-related tasks in simple, single-turn voice assistants, where the focus is more on anchoring an event to a point in time rather than on finding temporal relations between mentioned or implicit events.

TEs can fall into four categories generally as defined in TimeML/TIMEX3: DATE (e.g. *September 2000*), TIME (e.g. *3 p.m.*), DURATION (e.g. *two weeks*), SET (e.g. *twice a month*). The category of the TE is labeled as the TIMEX3

[1]We use the term *commands* to also refer to queries, such as *When do I have time tomorrow?*

[2]In the context of temporal tagging, documents are typically clustered into domains with regard to characteristics relevant for temporal tagging (Strötgen and Gertz, 2016). Within the voice assistant domain one can also identify (sub-)domains, such as the calendar domain, the infotainment domain, and so on.

```
Add my appointment at Varin Salon on

<TIMEX3 tid="t1" type="DATE" value="2020−04−27">
April 27th
</TIMEX3>

from

<TIMEX3 tid="t2" type="TIME" value="2020−04−27T10:30"
anchorTimeID="t1">
10:30 a.m.
</TIMEX3>

to

<TIMEX3 tid="t3" type="TIME" value="2020−04−27T11:30"
anchorTimeID="t1">
11:30 a.m.
</TIMEX3>

<TIMEX3 tid="t4" type="DURATION" value="PT1H"
beginPoint="t2" endPoint="t3" />

to the calendar.
```

(a) Time expressions with TIMEX3 tags.

```
{
    "data": [
        {
            "text": "I want to book a"
        },
        {
            "text": "restaurant",
            "entity": "restaurant_type"
        },
        {
            "text": "on"
        },
        {
            "text": "january third"
            "entity": "timeRange",
            "TIMEX3": [
                {
                    "expression": "january third",
                    "tid": "t582",
                    "type": "DATE",
                    "value": "2019−01−03"
                }
            ]
        }
    ]
}
```

(b) An example of the JSON formatting of Snips, enriched with TIMEX3 tags.

Figure 1

`type` attribute. See an example in Figure 1a, where the TimeML annotation is outputted in XML.

TIMEX3 tags are identified by unique IDs (`tid` attribute). The `value` attribute contains a normalized (machine-readable) format for the TE, following the ISO 8601 standard. In case the expression is of type DURATION, the `beginPoint` and `endPoint` indicate the TEs the DURATION is delimited by. Furthermore, if the duration is not explicitly mentioned, but can be inferred, then an empty content TIMEX3 is used, as in the example provided. The `anchorTimeID` attribute indicates the ID of the TE to which the tagged TIMEX3 is anchored. Please refer to the TimeML guidelines for further details on the TIMEX3 annotation (Saurı et al., 2006; TimeML Working Group, 2009).

3. Existing Datasets

We compare our dataset to three existing English-language datasets: the TBAQ dataset, the Twitter dataset and Snips.

The TBAQ dataset[3] was used as a gold standard for the TempEval-3 challenge, and it includes the TimeBank dataset (Pustejovsky et al., 2003) and the AQUAINT data (news domain) (UzZaman et al., 2013). TBAQ was annotated by experts following the TimeML annotation guidelines (Saurı et al., 2006).

The Twitter dataset (Zhong et al., 2017)[4] is a manually-annotated dataset of 942 tweets (written text), of which each contains at least one time expression. The tweets were also annotated with TimeML/TIMEX3 tags. As it is typical with tweet data, sentences are rather short and the format is rather noisy, as it contains several cases of alternative spelling, incomplete syntax, as well as hashtags.

Snips (Coucke et al., 2018)[5] is a crowdsourced dataset for the voice assistant domain, specifically for seven intents[6], which is widely used for benchmarking NLU components of voice assistants. However, no explicit details are provided on how the data was created or collected, and it does not appear to come from a real-world interaction with a voice assistant (sentences from Snips can at times be rather odd, albeit grammatical, e.g. *Get me a table for 2 people 1 second from now* or *In twenty three hours and 1 second my daughter and I want to eat at a restaurant*). The intents *GetWeather* and *BookRestaurant* contain the most TEs, but as the calendar (sub-)domain has a better potential for more diverse TEs (one typically only asks for the weather in the immediate future), we chose to focus on the calendar domain, and thus the *BookRestaurant* intent was the most fitting. In this paper we only refer to the commands from *BookRestaurant* as Snips.

Time expressions in Snips are simply identified as entities under the entity label(s) *timeRange*, but they are not annotated more specifically than this. For this reason, we annotated the time expressions present in the Snips dataset (for the *BookRestaurant* intent), following the TimeML annotation guidelines and adding the necessary TIMEX3 tags. As it is typical in this domain, Snips is formatted using the JSON format. In order to enrich Snips with a TIMEX3 annotation, it was thus necessary to add the relevant fields to each *timeRange* entity, as shown in Figure 1b. When needed, fields for `mod`, `anchorTimeID`, `beginPoint` and `endPoint` were added too.

[3] `https://www.cs.york.ac.uk/semeval-2013/task1/index.php%3Fid=data.html`

[4] `https://github.com/xszhong/syntime`

[5] `https://github.com/snipsco/nlu-benchmark`

[6] *BookRestaurant*, *AddToPlaylist*, *GetWeather*, *PlayMusic*, *RateBook*, *SearchCreativeWork*, *SearchScreeningEvent*

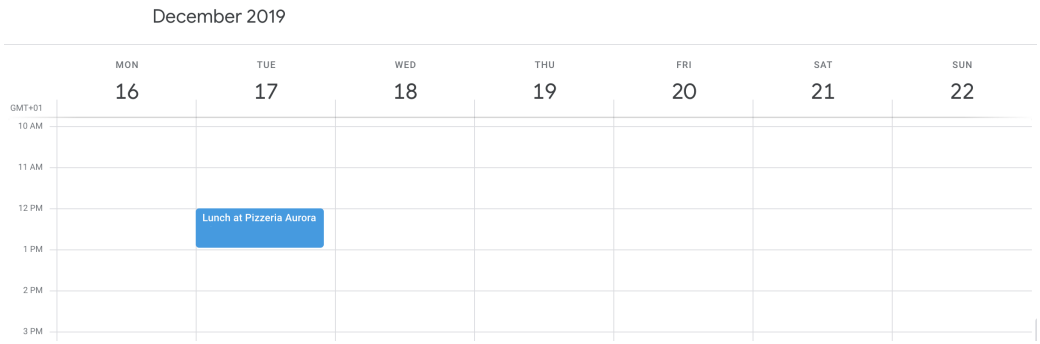

Figure 2: An example picture from our task for the intent *Event.book*, *restaurant* scenario: *You would like to go eat pizza with your friends. You want your assistant to call the restaurant for you.*

Table 2 sums up relevant descriptive statistics about the datasets[7].

4. Data Collection

Crowdsourcing can be a useful source to collect data for the voice assistant domain (Wang et al., 2012; Coucke et al., 2018), as it provides an effective way to collect high-quality data from non-experts (Snow et al., 2008). The quality of the collected data, however, is highly dependent on the way the task is presented to the participants. In particular, the data collection method should be carefully designed to (1) elicit naturalistic data in a controlled setting, and (2) avoid undesired biases, for example prompting the participants to repeat the same syntactic structures or lexical choices that are in the task description, which could happen if participants are just asked to paraphrase a command.

Wang et al. (2012) suggests eliciting user commands by using a description of the user intent and a list of slots, e.g. *Intent: The user wants to switch the lights on; slot: (bedroom)[room]* to generate the command or query *I want lights in the bedroom right now!*. We adopt the approach in Wang et al. (2012), but we modify it with the goal of immersing the participant in the relevant situation, albeit within the limitation and controlled setting of a crowdsourcing platform (where commands are typed and no voice interface is actually present). Work in psycholinguistics and cognitive science has shown that language is *situated*, meaning that it is always acquired and used in physical contexts within complex, real-world situations (Barsalou, 2008). There is also evidence supporting the hypothesis that providing relevant situational information can activate abstract concepts (Pecher et al., 2011; McRae et al., 2018; Davis et al., 2020).

We thus aimed at activating the user's intent without explicitly phrasing a command. We asked participants on a crowdsourcing platform (Amazon Mechanical Turk[8]) to imagine they were sitting in a smart car with an AI virtual assistant, which can help them manage their schedule and appointments. We provided the participants with *scenario descriptions*,which consisted of two sentences, a *situation* description (e.g. *You will attend a conference in another city and you would like to stay at the Holiday Inn Express*), providing background information about the situation the user should imagine to find themselves in, and an *intent* description (e.g. *Let your assistant make the reservation for you*). We avoided, whenever possible, words that could be used in the command by the user. Each scenario was accompanied by a picture, a screenshot of a web calendar app, showing the relevant time for the event mentioned in the scenario. Sometimes the time would be showed explicitly (e.g. *8 pm*), sometimes it needed to be inferred by looking at the time axis (as in Figure 2). Then we asked the participants how they would ask the assistant to help with the described goal. No time limit was imposed and presentation order of the items was randomized.

We collected commands for 5 intents in the calendar domain (*Event.book*, *Event.new*, *Event.search*, *Event.change*, *Event.delete*). For each intent, we selected four possible scenarios and wrote scenario descriptions (e.g. for the *restaurant* scenario: *You would like to go eat pizza with your friends. You want your assistant to call the restaurant for you.*). For each scenario, depending on the granularity of the event, we created two pictures for either day and week view or week and month view, and for each of those we created 4 different pictures with different times or days, thus obtaining a total of 160 items (5 intents, 4 scenarios per intent, 2 calendar views per scenario, 4 instances per calendar view). Each item was presented to 3 participants, so in the end we collected a total of 480 commands. We required that the participants had the Master Worker qualification, were fluent in English and had a minimum acceptance rate of 95%. A total of 30 participants took part in our collection, each of them providing answers for 16 items on average. We checked the elicited sentences and found we did not have to reject any of them.

After collecting the data, we manually checked it for spelling, grammar and general coherence with respect to the scenario and intent and we found that we did not need to discard any data. However, due to the limitation of the task, which required that the participants typed their answers, some responses contained digits and hyphens (e.g. *March*

[7]Token count for the Twitter dataset is reported as the word count from the paper (Zhong et al., 2017), where the authors do not specify if words are tokens and how they tokenized the dataset. For the sake of comparison with the other datasets, we converted numerical expressions in Snips (e.g. *twenty* minutes) into digits.

[8]`https://www.mturk.com/`

3-4). We modified such items by making them more similar to a speech transcript (e.g. *March 3-4* → *March 3 to 4*). We then formatted the data in a structured JSON format comparable with the one used in Snips, as it is typical for datasets in the voice assistant domain. The data is additionally labeled with intent labels and with scenario labels (e.g. *restaurant, hairdresser*). Time expressions were enriched with TIMEX3 tags as in 1b, see below for details on the annotation. An annotation of other relevant entities (e.g. *restaurant_type*) is in progress.

5. Time Expression Annotation

One expert annotator annotated the PÂTÉ dataset and the Snips dataset following the TimeML annotation guidelines for English (Saurı et al., 2006; TimeML Working Group, 2009) and adding TIMEX3 tags. We did not annotate any event or relationship between events. These are relevant for longer texts with a narrative, but for the voice assistant domain the main goal is to identify TEs and normalize them into a machine-readable format.

5.1. Inter-annotator agreement

Two more annotators additionally annotated 100 randomly-selected commands from Snips and 100 randomly-selected commands from PÂTÉ. We computed inter-annotator agreement on the sample annotated by all three annotators. In order to evaluate to what extent the annotators agreed on the TE `type` (class assignment), we computed agreement as Cohen's Kappa score (Multi-κ among three annotators (Davies and Fleiss, 1982)), resulting in 0.93 for the Snips dataset and 0.94 for PÂTÉ. For the `value` annotation, which is not a class assignment, we computed F-scores between annotator pairs, resulting in 92% and 97% for the Snips dataset and 87% and 98% for PÂTÉ. See (Bethard and Parker, 2016) for a detailed discussion of F-score in computing inter-annotator agreement for the value of TIMEX3 tags.

Most of the disagreement came down to small mistakes and was easy to overcome after the agreement computation, when the disagreements were analyzed. For example, one assigned the value *2019-11-XX* to *this month*, with the *XX* marking that the day can not be determined from the context, while another assigned the value *2019-11* (Snips). Other cases seem to be more truly ambiguous: in *What is on the schedule for me to do this week?* (PÂTÉ) one annotated *this week* as DURATION (value: P1W), that is a time range within events need to be searched, while another annotated it as DATE (value: 2019-W47), that is the (underspecified) date of the events to retrieve.

The inter-annotator agreement study was only performed on this sample due to time and resource limits. However, the results were encouraging enough to justify not running the entire dataset through more annotators.

5.2. Domain-specific guidelines

The voice assistant domain is quite different than the news and clinical domains, and typically does not require reporting of past events, but rather commands regarding that are yet are to come. In some cases, when annotating Snips and PÂTÉ, this required making some decisions regarding cases that were not addressed by the existing guidelines.

Reference to the future In a narrative or news text, where typically past events are reported, we can expect most TEs to refer to past events. In the voice assistant domain, on the other hand, and in particular in the calendar and scheduling sub-domain, typically the reference is the (immediate) future (*in November* would typically refer to next November, *on Tuesday* would typically refer to next Tuesday, etc.). Reference is often made to the later part of the present day. When the relevant time anchor is the present (e.g. *today*) then `anchorTimeID` is $t0$, which as a convention we set to be the day and time we started the annotation on.

Parts of the day In some cases people refer to certain moments of the day that either do not have a conventionally specific time (e.g. *lunch, dinner*), or denote a part of the day with fuzzy boundaries (e.g. *tonight, this evening*) or happen at different times depending on the season (e.g. *sunset / sunrise*). We thus created an (albeit arbitrary) lookup table to assign values to these expressions, which is an extension of the table provided in the TIMEX2 guidelines (Ferro et al., 2005). The assigned values could be still rather vague (AF for *afternoon*) or a specific time. These, however, are conventions and may need to be customized for different voice assistants (or even for different users).

Part of the day	assigned `value`	Part of the day	assigned `value`
afternoon∗	AF	midday∗	MD
breakfast	MO	morning∗	MO
brunch	MO	night∗	NI
daytime∗	DT	sunrise	MO
dinner	EV	sunset	EV
elevenses	MO	supper	EV
evening∗	EV	tea	AF
lunch	MI		

Table 1: Lookup table for values assigned to different times of the day. ∗ marks those already present in the TIMEX2 guidelines.

World knowledge Sometimes the participants just say *Book a table at 8*, without specifying if a.m. or p.m. In these cases, unless breakfast or brunch were explicitly mentioned, we assumed it was p.m. Such a decision, of course, relies on world knowledge of when people typically eat out, and on the fact that they are more likely to eat dinner at a restaurant in the evening.

Holidays In particular in Snips, we encountered some time expressions referred to a wide range of holidays (*First Day of Sukkot, Orthodox Good Friday, St. Patrick's Day, All Saint's Day*) as well as to some special days that are not widely known as holidays, for example *Pioneer Day* or *Thomas Jefferson's Birthday*. Assigning a normalized value to these TEs would require a rich lookup table. Furthermore, some of these holidays do not occur on the same day every year. We typically refer to the next coming occurrence of the holiday given the `anchorTimeID` $t0$.

In x time In some cases people asked the assistant to make a reservation *in x time*, e.g. *in an hour*. *An hour* is then annotated as a DURATION. However, the reservation needs to be done for a specific punctual time that is not expressed here. Such cases are very common in the voice assistant domain, and would have to be handled by a voice assistant by inferring the time of the reservation from the current time and the expressed time span (DURATION) from the current time to the target time. The inferred target time is then the `endPoint` time. See an example in Figure 3.


```
{
    "data": [
        {
            "text": "in one hour"
            "entity": "timeRange",
            "TIMEX3": [
                {
                    "expression": "one hour",
                    "tid": "t457",
                    "type": "DURATION",
                    "value": "P1H",
                    "beginPoint": "t0",
                    "endPoint": "t458"
                },
                {
                    "expression": "",
                    "tid": "t458",
                    "type": "TIME",
                    "value": "2019−09−03T21:00"
                }
            ]
        }
    ]
}
```


Figure 3: Time expressions with TIMEX3 tags - *in x time* (*I need seats for six at a vegan bar in one hour*).

From x to y In this domain it is common to find expressions such as *from x to y*, either for a duration (e.g. *reserve meeting room from 10 AM to 11 AM*) or to mark a change of time (e.g. *Change my dentist appointment from 10 AM to 11 AM*). The two TIME expressions are then tagged as two TIMEX3 tags, but in the first case with an additional DURATION element with the two tid of the time expressions as `beginPoint` and `endPoint`, in the other one without a DURATION element.

Sets Snips does not contain any expression of type SET. In PÂTÉ, however, they were present and were at times challenging to annotate. One challenge, for example, were sentences like the one in Figure 4 (*Add yoga to my calendar every Monday through Saturday at 6 a.m.*, where *Monday* and *Saturday* are both a set (as in *every Monday*, *every Saturday*) and the beginning and end of a range (as in *from Monday to Saturday*). We thus annotated only *every* as SET. Such an example shows the limitation of only using TIMEX3 tags and the necessity to annotate more complex relationships between temporal expressions (Bunt and Pustejovsky, 2010).

Uncertain expressions Whenever an expression did not have an obvious value (as it is the case, for example, for the ones in Table 1, we added an attribute *Uncertain* and we set it as *True*. Teasing apart defined and uncertain expressions can be useful to identify in which cases the developers of a voice assistant need to make a decision or adopt an arbitraty convention to resolve uncertain time expressions (for example, setting a specific time for *this evening*, or for *later this week*).


```
{
    "entities": [
      {
        "values": "every Monday through Saturday",
        "entity": "datetime",
        "TIMEX3": [
          {
            "expression": "every",
            "tid": "t1",
            "type": "SET",
            "value": "P1W",
            "quant": "EVERY"
            "freq": "6D"
          },
          {
            "expression": "Monday",
            "tid": "t2",
            "type": "DATE",
            "value": "XXXX−WXX−1"
          },
          {
            "expression": "Saturday",
            "tid": "t3",
            "type": "DATE",
            "value": "XXXX−WXX−6"
          },
          {
            "expression": "",
            "tid": "t4",
            "type": "DURATION",
            "value": "P6D",
            "beginPoint": "t2",
            "endPoint": "t3"
          }
        ]
      }
    ]
}
```


Figure 4: Time expressions with TIMEX3 tags - *sets* (*Add yoga to my calendar every Monday through Saturday at 6 a.m.*)

6. The Dataset

The PÂTÉ dataset is composed of 480 commands for the calendar domain. We collected commands for 5 intents, using different scenarios and calendar pictures. Even if the sentences in PÂTÉ are on average shorter (11.29 tokens per sentence against 13.67 in Snips, 19.32 for Twitter and 22.17 in TBAQ), the dataset is very rich in time expressions, with an average of 1.4 TEs per sentence, or 1.5 if we are including also empty content tags, which are used for example to infer the *endPoint* of durations in expressions such as *an hour from now*. Snips, on the other hand, has an average of 1.1 TEs per sentence (1.3 with empty tags), Twitter 1.2 and TBAQ only 0.4. The 480 annotated commands in PÂTÉ as well as the annotated Snips commands are available for research purposes[9].

PÂTÉ dataset contains very naturally-sounding commands to a virtual assistant in the calendar domain, which contain many TEs, covering all four TIMEX3 types. See in Ta-

[9]The dataset is available here: `https://doi.org/10.5281/zenodo.3697930`. The last access date, relevant for the analysis reported here, is March 13th, 2020.

Dataset	Domain	Tokens	Vocabulary	Avg. Sent. Length	Sentences	Sentences Containing TIMEXes	# of TIMEXes
TBAQ	News	99420	10024	22.17	4485	1459	1822
Twitter	Social Media	18199	4709	19.32	942	942	1129
Snips	Voice Assistant	9677	1531	13.67	708	697	947
PÂTÉ	Voice Assistant	5633	706	11.29	499	353	767

Table 2: Descriptive statistics about the datasets. PÂTÉ contains 480 commands but some are made up of two sentences (499 sentences total).

Intent	DATE	TIME	DURATION	SET
Event.book	79	34	28	-
Event.new	100	97	54	22
Event.search	40	-	2	-
Event.change	85	127	4	-
Event.delete	73	13	9	-
Restaurant.book	396	381	170	-

Table 3: TE type distributions in PÂTÉ and for Snips (Restaurant.book)

ble 3 TE type distributions per intent in PÂTÉ and for the *Restaurant.book* intent in Snips. Even given the obvious limitation of the task, that is that the commands were typed and not spoken in a real interaction with a voice assistant, the participants seem to have immersed themselves in the task. This is shown by examples as the following ones.

> *Get rid of my movie night with Ben from my schedule*
>
> *Take my 12 pm appointment off the calendar please*
>
> *I want you to prepone my group meeting by half an hour*
>
> *Move my haircut over to April 11th at 9 am.*

Participants used a wide variety of expressions to indicate scheduling changes, including some metaphorical ones (*move over, take off*).

> *Would you remind me when my appointment at hairdresser from the Varin Salon is coming up? Set the reminder for an hour before.*

Here a command was elicited for the Event.search intent, but the participant already anticipated the next intent (set reminder) which would naturally follow in a dialogue.

> *Move my haircut to Thursday please*
>
> *Change my dental examination from the 9th to the 10th at the same time.*

Here, as it is very common in spoken language, many contextually-supported metonimies (Schumacher, 2014) are used (*haircut* for *haircut appointment*, *dental examination* for *the time of the dental examination*).

> *Get me a hotel at Ibis from October 21 and checking out on October 22.*

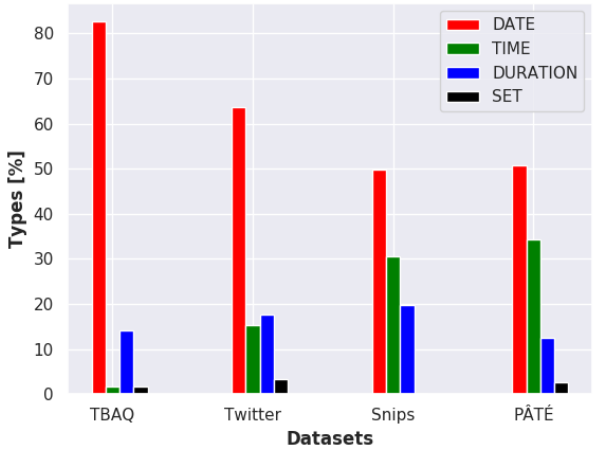

Figure 5: Distribution of different TE types in the four datasets.

Rather than saying *from ... to*, here the participant is using *checking out* to indicate the endpoint of his stay. This is very easy for us to understand, but for a voice assistant to understand this it would require some inference that the checking out day is the end of the stay.

7. Conclusion

We presented PÂTÉ (Personal Assistant Time Expressions dataset), a crowdsourced collection of commands for the calendar domain, collected by prompting participants with scenarios and calendar pictures in order to elicit commands containing time expressions. Crowdsourcing can be an effective way to collect data for a specific domain and presenting scenario helps participants immerse themselves in the situation and provide natural commands, albeit with the limitation that the participants are typing their commands and it is not a vocal interaction. Of course, interacting with a conversational agent is not natural at all, and for some people it may actually be particularly challenging, but providing relevant situational information helped the participants immerse themselves in the task and provide commands for a voice assistant that they could have reasonably also produced in a real interaction.

We focused on calendar-related tasks for simple, single-turn voice assistants, and thus TIMEX3 was the most relevant data structure and the only one we considered. In future work, however, moving away from single-turn systems

and going towards more complex tasks, for example going through the steps of a recipe, relationships between events would be relevant too and other TimeML data structures may be extremely useful. Reference to past events (e.g. asking when the last occurrence of an event happened) may in this case also become more relevant.

The dataset consists of 480 commands and can be integrated with the already-existing Snips dataset to reach a total of 1188 commands. Data augmentation techniques (Malandrakis et al., 2019) can then be used to create larger datasets to train the NLU components of voice assistants. Such a dataset, which contains time expressions as well as their machine-readable value, can also be extremely useful for Natural Language Generation, because they provide different ways of phrasing temporal expressions for the same value.

Presenting participants on a crowdsourcing platform with our task also allowed us to have a closer look at how people may talk to voice assistants. Uncertain expressions are actually common and even with a suitable knowledge base they constitute a challenge for virtual assistant designers (Rong et al., 2017; Tissot et al., 2016), which will need to find optimal strategies to deal with uncertainty or underspecification of time expressions.

8. Acknowledgements

The authors are grateful to Yanqing Hu and Paula Rothenberger for their valuable help with the annotation and formatting of our dataset, and to Katherine Munro, Anne Schleicher and three anonymous reviewers for providing valuable feedback on the paper.

9. Bibliographical References

Barsalou, L. W. (2008). Grounding symbolic operations in the brain's modal systems. In *Embodied grounding: social, cognitive, affective, and neuroscientific approaches*, pages 9–42.

Bethard, S. and Parker, J. (2016). A semantically compositional annotation scheme for time normalization. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3779–3786.

Bunt, H. and Pustejovsky, J. (2010). Annotating temporal and event quantification. In *Proceedings of 5th ISA Workshop*.

Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., et al. (2018). Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Davies, M. and Fleiss, J. L. (1982). Measuring agreement for multinomial data. *Biometrics*, pages 1047–1051.

Davis, C. P., Altmann, G. T., and Yee, E. (2020). Situational systematicity: A role for schema in understanding the differences between abstract and concrete concepts. *Cognitive Neuropsychology*, pages 1–12.

Ferro, L., Gerber, L., Mani, I., Sundheim, B., and Wilson, G. (2005). Standard for the annotation of temporal expressions-tides.

Fraser, N. M. and Gilbert, G. N. (1991). Simulating speech systems. *Computer Speech & Language*, 5(1):81–99.

Malandrakis, N., Shen, M., Goyal, A., Gao, S., Sethi, A., and Metallinou, A. (2019). Controlled text generation for data augmentation in intelligent artificial agents. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 90–98.

McRae, K., Nedjadrasul, D., Pau, R., Lo, B. P.-H., and King, L. (2018). Abstract concepts and pictures of real-world situations activate one another. *Topics in cognitive science*, 10(3):518–532.

Pecher, D., Boot, I., and Van Dantzig, S. (2011). Abstract concepts: Sensory-motor grounding, metaphors, and beyond. In *Psychology of learning and motivation*, volume 54, pages 217–248. Elsevier.

Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003). The TimeBank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

Pustejovsky, J., Verhagen, M., Nianwen, X., Gaizauskas, R., Hepple, M., Schilder, F., Katz, G., Saurí, R., Saquete, E., Caselli, T., et al. (2009). Tempeval2: Evaluating events, time expressions and temporal relations. *SemEval Task Proposal*.

Rong, X., Fourney, A., Brewer, R. N., Morris, M. R., and Bennett, P. N. (2017). Managing uncertainty in time expressions for virtual assistants. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 568–579. ACM.

Saurı, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006). TimeML Annotation Guidelines Version 1.2.1.

Schumacher, P. B. (2014). Content and context in incremental processing:âthe ham sandwichâ revisited. *Philosophical Studies*, 168(1):151–165.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Y. (2008). Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics.

Strötgen, J. and Gertz, M. (2016). *Domain-sensitive temporal tagging*. Morgan & Claypool Publishers.

Styler IV, W. F., Bethard, S., Finan, S., Palmer, M., Pradhan, S., de Groen, P. C., Erickson, B., Miller, T., Lin, C., Savova, G., et al. (2014). Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

TimeML Working Group. (2009). Guidelines for temporal expression annotation for English for TempEval 2010.

Tissot, H., Del Fabro, M. D., Derczynski, L., and Roberts, A. (2016). Normalisation of imprecise temporal expressions extracted from text. *Knowledge and Information Systems*, pages 1–34.

Tur, G. and De Mori, R. (2011). *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Tur, G., Celikyilmaz, A., He, X., Hakkani-Tür, D., and Deng, L. (2018). Deep learning in conversational lan-

guage understanding. In *Deep Learning in Natural Language Processing*, pages 23–48. Springer.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9.

Wang, W. Y., Bohus, D., Kamar, E., and Horvitz, E. (2012). Crowdsourcing the acquisition of natural language corpora: Methods and observations. In *2012 IEEE Spoken Language Technology Workshop (SLT)*, pages 73–78. IEEE.

Zhong, X., Sun, A., and Cambria, E. (2017). Time expression analysis and recognition using syntactic token types and general heuristic rules. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 420–429.