Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 3595–3600
Marseille, 20-25 June 2022

The IARPA BETTER Program Abstract Task
Four New Semantically Annotated Corpora from IARPA's BETTER Program

Carl Rubino, Timothy McKinnon
Intelligence Advanced Research Projects Activity (IARPA)
Bethesda, Maryland USA
{Carl.Rubino, Timothy.McKinnon}@iarpa.gov

Abstract

IARPA's Better Extraction from Text Towards Enhanced Retrieval (BETTER) Program created multiple multilingual datasets to spawn and evaluate cross-language information extraction and information retrieval research and development in zero-shot conditions. The first set of these resources for information extraction, the "Abstract" data will be released to the public at LREC 2022 in four languages to champion further information extraction work in this area. This paper presents the event and argument annotation in the Abstract Evaluation phase of BETTER, as well as the data collection, preparation, partitioning and mark-up of the datasets.

Keywords: CLIR, CLIE, zero shot, human-in-the-loop

1. IARPA's BETTER Program

The Intelligence Advanced Research Projects Activity (IARPA) kicked off the BETTER program in Boston on October 2019 to advance research in multilingual, cross-lingual and zero-shot information extraction (IE) and information retrieval (IR). BETTER was designed as a 42-month program, broken into three phases of decreasing length (18/12/12 months), each with its own evaluation language(s) and topic domain(s). The program goal was to develop enhanced methods for personalized, multilingual semantic extraction and retrieval from multilingual newswire text. Four large teams (or "Performers") were competitively selected to participate in the program based on their responses to a Broad Agency Announcement (BAA). The prime contractors for these teams were the University of Southern California Information Sciences Institute (USC-ISI), Brown University, Johns Hopkins University, and Raytheon BBN. Performers were contracted to develop systems that quickly and accurately extract complex semantic information from raw text documents in multiple languages in a way that is adaptive to the information needs of a specific monolingual English user. Systems were expected to leverage this extracted information to enable automatic IR and efficient human triage of relevant documents from massive stores of text documents. A major goal of the program was to incentivize the rapid adaptation of the technologies to new languages and domains with minimal effort.

For each of the three phases of the program IE and IR from a specific surprise domain (e.g., corporate mergers and acquisitions, protests, government corruption, epidemiology, etc.) was levied on the Performers by IARPA's Test and Evaluation Team. Within each phase, Performer systems were evaluated on three IE tasks and one human-in-the-loop (HITL) IR task. Within each phase, the initial task focused on extraction of *Abstract events*, which we describe in Section 2, while the second and third tasks focused on extraction of events comprising increasingly granular semantic information (so-called *Basic* and *Granular* events), previously referred to as "fine-grained" and "finer grained" events in Beieler (2018).

A major technical challenge posed by BETTER is that Performer systems must carry out 'zero-shot' learning: Performers are only provided training data in English, but their systems are evaluated on one or more surprise target languages, announced at the beginning of each program phase. This scheme ensures that Performers develop integrated, end-to-end IE/IR systems that are adaptive to analyst information needs and perform across differing domains and languages.

The corpora we present here were used for the first IE evaluation: the Abstract Task detailed in Section 2. Training data for this evaluation were provided in English only, and the evaluation was conducted in different languages: Arabic in Phase I, Persian in Phase II and a surprise language to be announced April 2022 for Phase III.

2. Abstract Task

The Abstract event task is the first of a series of three IE tasks deployed in IARPA's BETTER program. True to its name, the Abstract IE task is designed to focus on high-level, domain-independent event properties, abstracting away from the construction-specific peculiarities of specific event types. The dataset was designed not only to evaluate zero-shot cross-language event and argument identification across multiple domains, but also to serve as a resource for transfer-learning: It is intended that, as a result of pretraining on the Abstract task, neural systems should require less data when fine-tuning to domain-specific IE tasks requiring extraction of finer-grained semantic information (such as the Basic and Granular IE tasks).

One hallmark of the Abstract event extraction task is that the varied elements, or *arguments*, that are usually distinguished in most domain-specific event representations, such as _who_ did _what_ to _whom_, _when_, and _where_, are reduced to just two: **Agent** and **Patient**.

The notions of Agent and Patient are purely semantic, and do not depend in any way on the particulars of how an event and its participants happen to be expressed syntactically within a sentence. Agents are those things that—whether sentient or not, animate or mechanical—have caused an event or in some way set it in motion. Patients, on the other hand, are those things that are most directly affected by the event.

All categories required in the Abstract datasets are annotated at the word level even though affix annotation would suffice in certain languages.

The Abstract event extraction task attempts to capture attributes of events that are almost completely domain independent, and in doing so it is hoped that the extraction models derived from this domain-independent event data will prove useful across all subsequent phases of the BETTER Program, though probably to different degrees and in different ways.

2.1 Events

Abstract events are extremely abstract, defined roughly as "anything that indicates change, whether physically (in the world) or psychologically (in a person's head).

The sky is blue (no event)
My laptop was upgraded (event).

All events are annotated in the abstract corpus regardless of their epistemic status, whether the event actually occurred or not. Irrealis or hypothetical events are tagged, as well as events that occur with negative polarity such in the following sentences where the event anchor, or key indication of the event's occurrence, is italicized:

1) John will *cover* the fish tank,
2) If you *beat* him, I will *call* the police.
3) I will not *vote* for that candidate.

In addition to events that capture material change in the world, events that are mostly or purely verbal, encode mental changes of state or that attribute thoughts or beliefs to sentient actors are annotated. In English, these are often anchored by verbs like *said*, *announced*, *think*, or *believe*.

4) Rafi was *scared* by the snake.
5) Maggie *announced* to her constituents…"
6) I *think* they will *thwart* Dr. Sim's *execution*.

Events are not always mentioned with verbs, as shown in the previous sentences. A single sentence can mention zero, one, or multiple events, and can also refer to the same event in multiple ways.

7) My BMW's *repair* cost was astronomical.
8) The *injured* soldier was not Peruvian.
9) The Agno River *flooding*$_i$ was a *catastrophe*$_i$ that could have been *avoided*$_j$.

A description of a state of affairs, often indicated in English by an *is* or *has* verb, is not a taggable event. Examples of states of affairs would be:

10) John has brown eyes.
11) Justin Trudeau is Prime Minister of Canada.

However, words that indicate a change in a state-of-affairs, as in "Justin Trudeau was *elected* in 2015," are annotated as events.

2.2 QuadClasses

The Abstract event task also assigns a "QuadClass" to each event. The QuadClass is a simplified event type that indicates how an event should be classified according to two orthogonal dimensions: **Material-Verbal** and **Helpful-Harmful**. QuadClasses were originally conceived of as a simple alternative to elaborate geopolitical ontologies such as the Conflict and Mediation Event Observations (CAMEO) framework (Gerner *et al.* 2002). Beieler (2016), citing earlier work (Schein et al. 2016), argues that QuadClasses provide a low-dimensional, domain-independent means of representing many of the semantic contrasts that CAMEO encompasses.

Material events are those whose dominant or primary effects are physical, e.g., things moving in space, changes in the physical status of an object, such as breaking, etc., whereas Verbal events are those whose primary impact are the informational or cognitive change that they bring about, e.g., someone announcing something, or a person learning something, or coming to a new conclusion. Events are categorized as Helpful, Harmful or Neutral based on the impact they have, or are generally construed as having, on the patient.

2.3 Agents

When an event is mentioned, it will often include an indication of the person, organization or other kind of thing that has instigated or brought about the event, such as *the angry students* in the sentence "The angry students protested across the University of California's campuses on Wednesday." These are tagged as event *agents*. Some sentences include events that are associated with multiple agents (as in 12 below), while others describe events without mentioning any agents explicitly (as in 13 below). In the following sentences the agents are italicized:

12) Both *students* and *faculty* protested the quarantine measures.
13) The *protest* did not seem to have a specific goal.

Although agents often appear in English as subjects of active verbs, this is not always a reliable way of determining their role. BETTER annotators were taught to consider—for each event—what person, group of people, organization, or other actor, or even what event or state of affairs brought about the change that is described by the event. Notice the syntactic categories in English of the following agents, which are italicized.

14) The car was repaired by *Mr. Rao*.
15) Mr. Rao died from *dengue fever*.
16) Mr. Rao's abduction by *the local mafia* this morning has been widely reported.

2.4 Patients

Some events indicate the thing or things that were changed or otherwise affected by virtue of the event, such as "the community center" in the sentence: "John and Mary painted the community center". Notice that this event (*painted*) has two separate agents—John and Mary—which are annotated separately rather than as a single conjunctive phrase.

In the case of ditransitive verbs (verbs that take two objects), the patient is the entity impacted by the event. In English, this is typically the indirect object, and in such sentences the direct object is not annotated. The patients in the following sentences are italicized.

17) Sam gave *his teacher* a present.
18) Mary passed the ball to *Peter*.
19) Joan baked a cake for *her brother*.

This guideline also applies to nominalizations or other representations of this type of event, for example: "The gift of the book to the *baby* from his grandparents was a sentimental offering."

2.5 Events as arguments

Events can be agents of patients of other events. For example, in the sentence "The snowstorm prevented the students from writing their exams on Friday," the patient of the prevented event is the writing event. In this case, as a shortcut, we annotate the event anchor ("writing") as a proxy for the whole event.

When an event of speech or belief is identified, the patient of the event is defined to be the main event or proposition being asserted or believed. For example, in the sentences "The court announced that the hearing room had been filled" or "The judge believes the witness lied on the stand," the patient of "announced" is the filled event ("the hearing room had been filled"), and the patient of "believes" is the lied event ("the witness lied"). The patient of an assertion/belief event can also be a state of affairs that is not an event, such as "Ilya announced that Colleen was Kuwaiti."

2.6 Argument spans

While event anchors are identified through the minimal set of words necessary to clearly identify the nature of the event (ignoring any words conveying tense or other modifying information), agents and patients are annotated with the full noun phrase used to describe them. In English this includes determiners (the, a(n), some, any, my, your, their, etc.) and pre-nominal modifiers (such as adjectives), but excludes any additional dependent clauses.
Normally post-nominal prepositional phrases are excluded as well, such as "Three armed robbers from Switzerland" or "John Smith of Generic Corporation". The intent is to exclude clauses that are not needed to clarify the meaning of the head noun. Phrases or clauses that could sensibly be set off with commas or parentheses, are likewise excluded from argument annotation. However, when there is a function word that requires a completion such as "members of" or "president of", the post-nominal clause is included in the span: "Eight members of the Polar Bear club" or "The president of the board".

3. Data Selection

In Phase I, sentences were randomly chosen from newswire texts in the target languages harvested from the Common Crawl. An Amazon Mechanical Turk pipeline was set up to confirm the suitability of the extracted sentences according to genre (news) and language, e.g. Modern Standard Arabic for Phase 1 vs. dialectal variants. Training data for the Abstract event extraction task was presented at the sentence level on sentences that had been removed from their larger context and annotated in isolation. Each sentence included annotations for every Abstract event that occurred in the sentence. Some sentences contained no Abstract events while others contained more than one. Approximately 5,000 English sentences annotated at the Abstract event level were incrementally released to the Performers to enable them to train their multilingual extraction models.

3.1 Data Partitioning

Abstract data was partitioned by the BETTER T&E Team to optimize their use with regard to both training, system development, meaningful evaluation and error analysis. For Phase 1 of the program, for instance approximately 5,000 English sentences annotated at the Abstract event level were provided to the Performers to enable them to train their multi-lingual extraction models. This training corpus contained 15,416 Abstract events, divided into four official partitions:

- Training: 12,390 events
- Development test: 1,499 events
- Analysis: 1,527 events
- Training: 14,793 events

Save for a hidden partition that was retained by the T&E Team to test the effects of subsequent retraining, all partitions were released after the Performers had submitted their containerized software to IARPA. The Performers developed their systems using the released partitions prior to the evaluation, and the T&E Team subsequently subjected Performer systems to additional training during the evaluation.

The data consisted of Common Crawl newswire that had been annotated by the T&E Team and delivered JSON documents conforming to the BETTER Program (BP) schema discussed in Section 4.

4. Annotation

Annotators for the abstract task were chosen from a pool of bilingual candidates that demonstrated proficiency in the Abstract task. In Phase 1, all annotators passed a four-hour web-based qualifying exam authored by MITRE in the target language which they took after having received relevant written training in Modern Standard Arabic. In Phases 2 and 3 of the program, annotators were selected via in-person interviews to gauge their task suitability. Training for the last two phases was conducted in-person in a classroom environment. Annotators met on a weekly basis to review questions as a team, and maintained open communication as well via a slack channel hosted by the University of Maryland for any questions or concerns. Training was conducted in English across multi-language teams, to maximize information delivery and emphasize the semantic vs. syntactic nature of the annotation.

Each sentence was doubly annotated. Where discrepancies occurred in the annotation, a third annotator adjudicated to provide the official result. The adjudicators were often also the original trainers.

Annotation was performed in a tool created and hosted by MITRE. The tool is web-based, so much of the annotation was performed remotely.

5. Scoring

For each Abstract event, the following are provided to enable system scoring via the BP JSON format described in 5.1:

- A list of distinct agents participating in the event, specified as a list of mention strings for each of the distinct agents
- A list of distinct patients participating in the event, specified as a list of mention strings for each of the distinct patients
- A string representing the event "trigger"—a word or phrase that is the strongest lexical indicator of the event that is being annotated
- The values of each of the "Quad-Class" event attributes
    - Material-Verbal: Material, Verbal, Both, Unknown
    - Helpful-Harmful: Helpful, Harmful, Neutral/Unknown

5.1 BP JSON Format

All annotated data in the program—training data, system-generated data, and the reference data against which system output is to be scored—are in a program-specific JSON format called BP JSON. MITRE provided Performer teams with tools to aid in the creation, parsing and syntax checking of BP JSON data. Our example data in this section will illustrate the format using a reference data example. The specific "role" of a BP JSON file/corpus is indicated by the value in the "annotation-role" field—one of "reference", "training", "system" or "unannotated". Reference data will usually include additional fields that are ignored for strict scoring but can be useful for T&E analysis, such as performing error analyses, computing inter-annotator reports and tracing data though various stages of processing.[1]

The BP JSON format consists of two main parts: (1) a set of metadata fields that *describe* various attributes of the corpus; and (2) a particular field, the "entries" field, whose value is *the data that constitutes* the corpus. The corpus data can consist of raw data (for example, the text of a sentence, paragraph, or whole document), annotations (at any of the various types already mentioned, such as abstract events, basic events, etc.), or a combination of those two (where the annotations are derived from the raw (text) data). In the description of this format below, optional and required fields are identified. In general, it is expected that Performer systems will only bother to generate the required data fields, while the reference and training data created by MITRE will usually include all the optional fields.

The BP JSON format captures annotations on texts, and organizes "raw" or un-annotated data. Performer systems are required to read and process such un-annotated data as part of the evaluation. For example, when systems are asked to perform automatic Abstract event extraction, this level of analysis is performed on individual sentences, so it is useful to provide all of the Performer systems with a clearly defined corpus of sentences (the automatic recognition of sentence boundaries was not itself a key technology to be evaluated within the BETTER Program). This enables system output to unambiguously identify the text segment from which a particular set of annotations is derived. It also affords the system-generated output to be more succinct, avoiding the need to emit the text segment text itself and any other associated metadata, so long as the output provides the required unique identifiers of the text segments.

5.1.1 Corpus Metadata

In the top-level metadata fields that describe the corpus, there are a small set of required fields and a larger number of optional fields. The required fields are listed below, with a description of their purpose and format.

- format-type: which must be the string "bp-corpus"
- format-version: a string, currently "v8f"
- corpus-id: a string, ideally unique to the corpus
- entries: a table of corpus entries indexed by each corpus entry identifier (the structure of each corpus entry value is described below)
- annotations-role: should be one of *system*, *reference*, *training* or *raw* (the latter is used when there are no annotations in the corpus, such as when the corpus is merely capturing "raw," un-annotated data for processing)
- annotator-id: the identifier of the annotator, a string; for system output, this would be the identifier of the Performer system, including its specific version number, and the value would be "none" in the event the corpus is only a repository of un-annotated data
- annotator-type: indicates whether the annotator was a MITRE corpus development team member, an anonymous Amazon Mechanical Turk worker, an automated system, etc.

The top level fields that provide various metadata about a corpus are provided below. All fields are required, except the "annotation-types" field.

```
{"format-type": "bp-corpus",
 "format-version": "7.1",
 "corpus-id": "sents-set-10",
 "annotations-role": "reference",
 "annotator-id": "turker-1234",
 "annotator-type":  "turker",
 "annotation-types": ["abstract-events", "basic-
events"],
 "entries":
```

1 Examples of such additional information that might be specified in reference annotated data include: the text being annotated, the event trigger strings, metadata about the source of the data and the annotations, etc.

```
  {"sent-10-1": {...},
   "sent-10-2": {...},
    ...}
}
```

5.1.2 Corpus Entries

The value of the entries field in the BP JSON corpus data structure is a table of JSON table elements, each indexed by a unique entry-id string. These elements are intended to capture all the annotations relative to a specific segment of text. Note that in the case of Abstract event annotations, the unit of analysis is the sentence, so the annotations on each distinct sentence in the corpus are found in separate entry records. Only annotations that are derived from the identified segment of text will be found within a distinct entry—these entries can be considered independent from any annotations found in other entries elsewhere in the corpus.

```
{"entry-id":      "sent-10-2",
 "source":       "sents-set-10.txt",
 "segment-type":  "sentence",
 "segment-start": 42,
 "segment-end":   84,
 "segment-text":  "Ellen opened the door to welcome
the chef.",
 "annotation-sets": {
   "abstract-events": {...}
   "basic-events":    {...},
   ...
}
```

The value of each corpus entry is itself a table, in which there is only one required field—the entry-id field. Because Performer systems process data that is already in BP JSON format, where individual segments of text have already been identified with an entry-id,[2] system-generated output only needs to include the entry-id information. For training and reference purposes, however, it is useful to include either the text itself (in the segment-text field) and/or information on where to find the raw text (in the source, segment-start and segment-end fields).

5.1.3 Annotation Sets

A corpus entry captures annotation data by including the `annotation-sets` field, which is a table of sets of annotations indexed by their annotation type (abstract-event, basic-event, or granular-event). The two main fields in each Abstract event annotation-set table entry are "span-sets" and "events". Abstract events are fully specified by the value of the "quadclass" category and the agents and patients. The anchor field is optional, though it is almost always provided in both training and reference data.

Since the data format needs to support reference and training data as well as system output, the representation of event arguments must capture all the different mentions of an event argument that might be mentioned in a sentence. For this reason, the arguments in the events are captured in "span sets"—effectively the set of entity (or other non-entity) mentions that are co-referential. The BETTER program is not explicitly evaluating co-reference resolution, so system output will be judged correct if it identifies *at least one* of the co-referring mentions of an event argument. If there are some mentions that are not pronoun mentions, system output will not be judged correct unless one of the non-pronominal mentions is included in the set of spans. For readability, the event arguments are specified via a span set identifier (effectively like an entity identifier), and the various strings (spans) that are co-referring are listed separately in a table of span-sets, where the span-set-id is the index into that table.

Events can have multiple agents and/or patients, and each of those agents and patients may have multiple mentions somewhere within the body of the text. In the following example, the event being annotated has two agents, and one of the agents is referred to in three different ways. The reference data needs to account for all of these valid mentions of "John", although system-generated output would only need to mention one of the valid mentions.

```
{"entry-id":      "sent-12-4",
 "source":       "sents-set-12.txt",
 "segment-type":  "sentence",
 "segment-text":  "Frieda, with her friend and fellow
chef, John,
                managed to open the door at the last
minute.",
 "annotation-sets":
   {"abstract-events":
     {"events":
       {"e1": {"quadclass": {"mv":"material",
"hh":"helpful"},
               "agents":    ["s1", "s4"],
               "patients":  ["s2"],
               "anchors":    "s3"}}
      "span-sets":
        {"s1": {"spans": [{"string": "Frieda"}]},
         "s2": {"spans": [{"string": "the door"}]},
         "s3": {"spans": [{"string": "open"}]},
         "s4": {"spans": [{"string": "John"},
                          {"string": "her friend"},
                          {"string": "fellow chef"}]}}}}}
```

5.2 A Complete Example

In this section we show how the previous elements of the BP JSON corpus format can be merged into a single complete example. This example focuses on reference data, as that way it can illustrate all the optional fields that are supported in BP JSON. It should be emphasized that system-generated output is expected to be much more minimal, focusing on only those elements that are required for processing and scoring. Notice the absence of any offsets into the sentence text, as well as the exclusion of the sentence text itself. Also note that while it is required that the event and span-set identifiers are unique within the annotation-set, there is no requirement that the identifiers

[2] In the case of Abstract events, these segments consist of full sentences.

match in any way the identifiers that happen to be used in the reference data set.


```
{"format-type":       "bp-corpus",
 "format-version":    "7.1",
 "corpus-id":         "sents-set-10",
 "annotations-role":  "reference",
 "annotator-id":      "turker-1234",
 "annotator-type":    "turker",
 "annotation-types":  ["abstract-events"],
 "entries":
  {"entry-id":     "sent-10-2",
   "source":       "sents-set-10.txt",
   "segment-type": "sentence",
   "segment-start": 42,
   "segment-end":   84,
   "segment-text": "Ellen opened the door to
welcome the chef.",
   "annotation-sets": {
    "abstract-events":
     {"events":
      {"e7": {"quadclass": {"mv":"material",
"hh":"helpful"},
              "agents":   ["ss6"],
              "patients": ["ss5"],
              "anchors":   "ss4"},
       "e5": {"quadclass": {"mv":"verbal",
"hh":"helpful"},
              "agents":   ["ss6"],
              "patients": ["ss2"],
              "anchors":   "ss3"}}
      "span-sets":
       {"ss6": {"spans": [{"string": "Ellen"}]},
        "ss5": {"spans": [{"string": "the door"}]},
        "ss4": {"spans": [{"string": "opened"}]},
        "ss3": {"spans": [{"string": "welcome"}]},
        "ss2": {"spans": [{"string": "the chef"}]}
   }}}
```


6. Conclusion

The BETTER Abstract dataset represents a unique contribution to the resources available within the information extraction space. Unlike prior event and relation extraction tasks, which have tended to emphasize syntactic predicate-argument structure, the Abstract task adopts a semantically-based event annotation scheme that focuses on understood events and a reduced valence frame that comprises only the two most semantically salient arguments. We hope that this shift in perspective will substantially benefit research into cross-domain and multilingual event extraction.

7. Acknowledgements

The authors would like to thank the BETTER T&E Team for their work on data compilation, annotation and evaluation, which ultimately resulted in these datasets: David Day, Allison Powell, Leland Vakarian, Ian Soboroff, Michelle Morrison, and Aric Bills. We would also like to thank Daniel Child for his comments and feedback. This effort is supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

8. Bibliographical References

Beieler, J. (2018). BETTER Broad Agency Announcement. *https://bit.ly/33eIu5D*

Beieler, J. (2016). Generating politically-relevant event data. *arXiv preprint arXiv:1609.06239*.

Castor, A. and Pollux, L. E. (1992). The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

Gerner, D. J., Schrodt, P. A., Yilmaz, O., & Abu-Jabr, R. (2002). Conflict and mediation event observations (CAMEO): A new event data framework for the analysis of foreign policy interactions. *International Studies Association, New Orleans*.

Schein, A., Zhou, M., Blei, D., & Wallach, H. (2016, June). Bayesian Poisson Tucker decomposition for learning the structure of international relations. In *International Conference on Machine Learning* (pp. 2810-2819). PMLR.