Cold Start Knowledge Base Population at TAC 2013

Task Description[1]

Version 1.1 of July 31, 2013

Introduction

Since 2009, the TAC Knowledge Base Population Track has evaluated performance on two important aspects of knowledge base population: entity linking and slot filling. However, the ability of a system to use these technologies to actually construct a knowledge base (KB) from the information provided in a text collection had not been exercised. The Cold Start task was designed to evaluate a system's ability to do just that. Participants build a software system that processes a large text collection and creates a knowledge base that is consistent with and accurately represents the content of that collection. The knowledge base is then evaluated as a single connected resource.

The Entity Linking and Slot Filling tasks have done a good job of evaluating key components of knowledge base population. They do not, however, evaluate every aspect of an automatically generated knowledge base. Things one might like to know about such a knowledge base include:

- Are the entities in the knowledge base correctly tied to real-world mentions of those entities? The TAC Entity Linking task measures this.
- Are the facts and relations in the knowledge base accurate reflections of the facts and relations described in the source documents? The TAC Slot Filling task measures this.
- Are entity linking and slot filling correctly coordinated to produce a meaningful knowledge base? The TAC Cold Start task measures this.
- Can the knowledge base correctly perform inference over the extracted entities, such as temporal reasoning, confidence estimation, default reasoning, transitive closure, etc.? Cold Start does not yet measure this, but is designed to facilitate this kind of evaluation in future years.

We call the task *Cold Start* Knowledge Base Population to convey two features of the evaluation: it implies both that a knowledge base schema has been established at the start of the task, and that the knowledge base is initially unpopulated. Thus, we assume that a schema exists for the facts and relations that will compose the knowledge base; it is not part of the task to automatically identify and name facts and relationships present in the text collection. We will use the schema that is implicitly specified by the TAC 2013 Slot Filling task. Thus, the schema will include three entity types (person, organization and geopolitical entity) and forty-one relation types. For relations whose fills are themselves entities (such as `per:siblings` or `org:subsidiaries`), systems will be required to link that slot to the KB node representing the correct entity. Slots whose fills are strings (such as `per:title` or `org:website`) will continue to use strings to represent the information.

Cold Start also implies that the knowledge base is initially empty; in particular, we assume that a Wikipedia dump is *not* the starting point for the knowledge base. To avoid solutions that rely on

[1] The TAC organizing committee welcomes comments on this Task Description, or on any aspect of the TAC evaluation. Please send comments to `tac-kbp@nist.gov`.

verifying content already present in Wikipedia or other large data sources about entities, the document collection used in Cold Start will be dominated by entities that are not present in Wikipedia. In 2013, the Cold Start collection will likely be a subset of the TREC KBA Stream Corpus 2013.[2] NIST will distribute the collection; participants will not need to use Amazon S3 to access it.

Participating systems will receive the following inputs:

1. a *knowledge base schema;*
2. a *document collection;* and possibly
3. the output of the BBN Serif information extraction system run over each of the documents in the collection.[3]

From these, systems will produce a knowledge base. This KB will be submitted to NIST as a set of augmented triples; in future years, the entire KB (including inference capability) may be submitted and evaluated. Participating systems must tie each entity mention in the document collection to a particular KB entity node; in this way, the knowledge base can be queried without first aligning it to a reference knowledge base.

The submitted knowledge base will then be evaluated by NIST. Evaluation will use a set of KB evaluation queries. Each query will start at a character in a document, identify the knowledge base entity that has a mention that includes that character, follow a sequence zero or more relations within the knowledge base, and end in a slot fill. The resulting slot fills will be assessed and scored in much the same way as is now done in the Slot Filling task. For example, a KB evaluation query might ask 'what are the ages of the siblings of the *Bart Simpson*[4] mentioned in Document 42?' A system that correctly identified descriptions of Bart's siblings in the document collection, linked them to the appropriate node in the KB, and also found evidence for and correctly represented the ages of those siblings would receive full credit.

Schema

[2] http://trec-kba.org/kba-stream-corpus-2013.shtml

[3] We are exploring this possibility as a way to decrease the barriers to entry for teams that would like to participate, but who do not yet have their own extraction system. Details will be provided in a later version of this document.

[4] Many of the examples used to illustrate the Cold Start task are drawn from *The Simpsons* television show. Readers lacking a detailed working knowledge of genealogical relationships in the Bouvier/Simpson family need not agonize over what they have been doing with their lives for the past quarter century, but may simply visit http://simpsons.wikia.com/wiki/Simpson_Family.

Relation	Inverse(s)
per:children	per:parents
per:other_family	per:other_family
per:parents	per:children
per:siblings	per:siblings
per:spouse	per:spouse
per:employee_or_member_of	{org,gpe}:employees_or_members*
per:schools_attended	org:students*
per:city_of_birth	gpe:births_in_city*
per:stateorprovince_of_birth	gpe:births_in_stateorprovince*
per:country_of_birth	gpe:births_in_country*
per:cities_of_residence	gpe:residents_of_city*
per:statesorprovinces_of_residence	gpe:residents_of_stateorprovince
per:countries_of_residence	gpe:residents_of_country*
per:city_of_death	gpe:deaths_in_city*
per:stateorprovince_of_death	gpe:deaths_in_stateorprovince*
per:country_of_death	gpe:deaths_in_country*
org:shareholders	{per,org,gpe}:holds_shares_in*
org:founded_by	{per,org,gpe}:organizations_founded*
org:top_members_employees	per:top_member_employee_of*
{org,gpe}:member_of	org:members
org:members	{org,gpe}:member_of
org:parents	{org,gpe}:subsidiaries
org:subsidiaries	org:parents
org:city_of_headquarters	gpe:headquarters_in_city*
org:stateorprovince_of_headquarters	gpe:headquarters_in_stateorprovince*
org:country_of_headquarters	gpe:headquarters_in_country*

Table 1. Entity-valued slots. Slots with asterisks represent relations that are newly defined for Cold Start because they are not part of the current Slot Filling task definition. The type qualifier of each relation (per, org or gpe) is the type of its subject, while the type qualifier for its inverse is the type of its object. A set of types means that any of those types is acceptable for that slot. All submitted slot

The schema for Cold Start 2013 is derived directly from the Slot Filling task specification. Slot Filling defines forty-one slots. Twenty-six of these have fills that are themselves entities, as shown in Table 1. The remaining fifteen slots have string fills, as shown in Table 2. Each entity-valued slot will have an inverse. Some slots, such as per:siblings, are symmetric. Others, such as per:parents, have inverses that are already Slot Filling task slots (in this case, per:children). The remaining slots (e.g., org:founded_by) have no corresponding slot in the Slot Filling task; Cold Start specifies new slot names for these inverses. All such newly-introduced slots are list-valued. All inverse relations must be explicitly identified in the submitted knowledge base. That is, if the KB asserts that relation R holds between entities A and B, then it must also assert that relation R^{-1} holds between B and A.

In 2013, the per:employee_of and per:member_of have been merged into a single slot, per:employee_or_member_of.

per:alternate_names	org:alternate_names
per:date_of_birth	org:political_religious_affiliation
per:age	org:number_of_employees_members
per:origin	org:date_founded
per:date_of_death	org:date_dissolved
per:cause_of_death	org:website
per:title	
per:religion	
per:charges	

Table 2. String-valued slots.

Document Collection

Details on the document collection, including format, will be forthcoming. Sample documents will be included to allow participants to verify that their systems work on the test data. Because the task can begin as soon as the document collection is available, the full collection will not be distributed until the start of the task window. We anticipate a collection of approximately 30,000 to 100,000 documents.

Task Output

Systems must produce a knowledge base as output. The first line of the output file must contain a unique run ID, which is a single token that contains no white space and no pound sign, and that does not begin with a colon. The remainder of the KB is represented as a set of augmented triples. Assertions will appear, one-per-line, in tab-separated format. The output file will be automatically converted to RDF statements during evaluation. All output must be encoded in UTF-8.

Each triple appears in the output file in subject-predicate-object order. For example, to indicate that entity-4 has entity-7 as a sibling, the triple might be:

```
:e4    per:siblings  :e7
```

If entity-4 has siblings in addition to entity-7, these relations should be entered as separate triples.

Entities

Each entity specification begins with a colon, followed by a sequence of letters, digits and underscores. Examples of legal entity specifications include `:Entity42`, `:EE74_R29`, and `:there_were_two_muffins_in_the_oven`. No meaning is ascribed to this sequence by the evaluation software; it is used only as a unique identifier. Any subsequent use of the same colon-preceded sequence will be taken as a reference to the same entity.

Predicates

The legal predicates are those shown in Table 1 (including inverses) and Table 2, plus `type`, `mention`, and `canonical_mention`. Predicates found in Table 1 must have entity specifications in both the subject and object positions. Predicates found in Table 2 must have an entity specification in the subject slot, and a double quote-delimited string in the object position; the string in the object

position will exactly correspond with the slot fill for that relation in the Slot Filling task. A backslash character must precede any occurrence of a double quote or a backslash in such a string.[5]

Each entity will be the subject of exactly one `type` triple. The object of that triple will be either `PER`, `ORG` or `GPE` depending on the type of the entity. It is up to submitting systems to correctly identify and report the type of each entity.

Each entity will be the subject of one[6] or more `mention` triples. Together with the provenance information (see below), these triples indicate how the knowledge base is tied to the document collection. Each named entity mention in the collection should be submitted as the object of a `mention` triple. For example, if an entity is mentioned by name five times in a document, five `mention` triples should be generated. The object of a `mention` triple is the double-quoted mention string; document ID and offset appear under provenance information (see below). Mentions may not overlap. Put another way, no character in any document may be part of the object of more than one `mention` triple.

In the Slot Filling task, all slot fills are strings. Assessors verify the validity of a slot fill by looking for that string in the specified document, using the provenance information provided in the system response. In a submitted KB, slots that are filled with entities hold not strings, but pointers to the KB structure for the appropriate entity. During assessment, the assessor must be presented with a string that represents such an entity. Thus, for each document that mentions an entity, one of the `mentions` must be identified as the *canonical mention* for that document; it is the string that will be seen by the assessor when that entity appears as a slot fill, supported by that document. This implies that a document attesting to a relation must contain named mentions of both the subject and (if it is an entity) the object of the relation. Canonical mentions are expressed using a `canonical_mention` triple. The arguments for `canonical_mention` are the same as for `mention`. Note that there is no requirement that submissions select a single, global canonical mention for an entity. While such a name might be useful (and is a part of the current Entity Linking KB), here we require that a name be provided within each document for the assessor to use. Each `canonical_mention` is also a `mention`. As a convenience, if a submitted KB does not contain a `mention` triple for each `canonical_mention` triple, the missing relations will be inferred (albeit with a warning). This shortcut is provided to make submitted KBs easier to view, and does not relieve submitters from the requirement to provide each of the required `mentions` and `canonical_mentions`.

At least one instance of each unique subject-predicate-object triple will be evaluated. If more than one instance of a given triple appears in the output (with each triple having different provenance), LDC will assess the instance with the highest confidence value (see below), and will assess additional instances if resources allow. If more than one such triple shares the same confidence value, the triple that appears earlier in the output will be considered to have higher confidence.

Provenance

Each triple will be followed by a set of augmentations (again using tabs as separators). Except for the `type` slot (which does not require explicit support from a document) the first annotations will describe the provenance of the assertion. Cold Start requires three justification fields for subject, relation, and object, in that order. Note that this order is the same as in the Cold Start 2012 task, but

---

[5] Each backslash used to quote the following character doesn't itself have to be preceded by another backslash.

[6] While unmentioned but inferred entities may play a role in future TAC evaluations, Cold Start 2013 will work only with entities that have named mentions.

differs from the order in the 2013 Slot Filling task (which uses object, subject, predicate). Thus, provenance for a triple is expressed as four tab-separated values: the document ID,[7] offsets for the subject, offsets for the sentences justifying the relation, and offsets for the object. These four tab-separated values follow the subject-predicate-object values, separated from them by a tab. The `mention` and `canonical_mention` slots will have only a single justification, representing the location of the mention in the text. As in slot filling, the `per:alternate_names` slot may omit provenance for the relation if no textual context was used for the extraction (this will result in two adjacent tabs in the output file).

To assist with this transition, the Cold Start validator script will convert a properly constructed submission file from 2012 to 2013 format. Run the validator with the –h flag to see the appropriate switches.

In 2013, provenance will follow the requirements listed in the 2013 *English Slot Filling – Regular and Temporal* Task Description (with the exception of the ordering of the offsets, as described above). The offsets that document the provenance of an extracted relation are used to narrow the assessor's focus within the document when assessing the correctness of that relation. Briefly, each offset pair is the character positions of the first and last characters of the word or words that best conveys the entity or predicate, separated by a dash. At most two offset pairs may be included in a justification field, separated by commas. Subject and object justifications are the offsets of one or two entity mentions, of which at least one must be a named entity mention. The predicate justification will be the offsets of at most two clauses or sentences (entries with more than two sentences in the Predicate justification will be automatically judged as incorrect). The Cold Start task will follow exactly the 2013 Slot Filling task rules for how offset pairs are to be specified; please see the documentation for that task for a complete specification of provenance offsets.

Confidence Measure

To promote research into probabilistic knowledge bases and confidence estimation, each triple may have an associated confidence score. Confidence scores will not be used for any official TAC 2013 measure. However, the scoring system may produce additional measures if confidence scores are included. For these measures, confidence scores will be used to induce a total order over the facts being evaluated (ties are broken when two scores are equal by assuming that the assertion appearing earlier in the submission has a higher score). Any submitted confidence score must be a positive real number between 0.0 (exclusive, representing the lowest confidence) and 1.0 (inclusive, representing the highest confidence), and must include a decimal point (no commas, please) to clearly distinguish it from a document offset. Confidence scores, if present, will appear at the end of each output line, separated from the provenance information with a tab. Confidence scores may not be used to qualify two incompatible fills for a single slot; submitter systems must decide amongst such possibilities and submit only one. For example, if the system believes that Bart's only sibling is Lisa with confidence 0.7 and Milhouse with confidence 0.3, it should submit only one of these possibilities. If both are submitted, it will be interpreted as Bart having two siblings.

Comments

Output files may contain comments, which begin at any occurrence of a pound sign (#) and continue through (but do not include) the end of the line. Comments and blank lines will be ignored.

---

[7] TAC 2013 will operate only over relations that are fully attested within a single document.

The first line of an output file must contain the unique run ID (i.e., it may not be blank). Submitters may like to add a comment to this line giving further details about the run.

Examples

The following three lines show examples of a triple without any annotations, one with only provenance annotation, and one with both provenance and confidence annotations.

```
:e4    type          PER
:e4    per:siblings  :e7    D00124  283-288     173-179  274-281
:e4    per:age       "10"   D00124  180-181     173-179  182-191  0.9
```

Evaluation

Evaluation Queries

NIST will evaluate each submitted knowledge base by running a set of evaluation queries against it and assessing the results. An outline of the NIST process is given below.

All evaluation queries start with an *entry point* into the knowledge base being evaluated. An entry point is simply a character position in a document that corresponds to an entity mention in that document. For example, the position of the *'S'* in a mention of *Bart Simpson* in Document 42 might be an entry point for an evaluation query. Given a knowledge base, the starting node in the knowledge base for a query will be the node that has a `mention` relation that includes the entry point. Because no character in the document collection may be part of more than one `mention` relation, this uniquely identifies a node in the knowledge base if the KB includes such a node. The proper specification of `mention` relations in a KB is therefore important for scoring well; participants should therefore take care to ensure that every entity mention in the evaluation collection serves as a `mention` relation for a node in the KB.

Evaluation queries could take many forms. For example, a query that asked for slot fills for an entity mentioned in a particular document would look very much like the Slot Filling task. For Cold Start, most of the evaluation queries will start from an entry point, select the corresponding KB entity, follow a single entity-valued relation (from Table 1), then ask for a single slot value (from either Table 1 or Table 2). For example, an evaluation query corresponding to the question 'what are the ages of the siblings of the *Bart Simpson* mentioned in Document 42?' would be of this form. Such "one-hop" queries will verify that the knowledge base is well-formed in a way that goes beyond basic entity linking and slot filling, without allowing combinations of errors to drive scores to zero. Note that unlike the Slot Filling task, each Cold Start evaluation query will ask for a specific slot, not all slots for which there is information in the document collection.

The NIST evaluation of a KB will proceed by finding all entries in the KB that fulfill an evaluation query. For example, if the evaluation query 'schools attended by the siblings of *Bart Simpson*' finds two siblings for the node specified by the entry point, and the KB indicates that those siblings attended two and one schools respectively, then three results would be assessed by NIST. These results will be converted to a form similar to the results of the Slot Filling task. Results will be pooled across all submissions, and assessors will judge the validity of each result. Finally, a scoring script will report a variety of statistics for each submitted run.

Evaluation queries and the answers to them produced from the submitted knowledge bases will be made available to participants, but only after the evaluation is concluded.

Assessment

Cold Start 2013 assessment and scoring will be similar to Slot Filling assessment and scoring. The results for each assessment query will be pooled, and each response will be assessed by a person. For one-hop queries, the result of following the first relation will be assessed as if it were a Slot Filling query (where the canonical name of the object entity in the supporting document will be used for the slot fill). The second relation in the query will also be assessed as a Slot Filling query, but only if the fill for the first relation is correct. For example, if the query asks for the ages of the siblings of "Bart Simpson," and the submitted knowledge base gives "Lisa age 8" and "Milhouse age 10" as siblings, then only the reported age of Lisa will be assessed (Milhouse is not Bart's sibling).

Cold Start uses *pseudo-slot* scoring, in which each evaluation query is treated as if it selects a single indivisible slot. For example, an evaluation query that asks for the children of the siblings of an entity will be scored as if it were a query about a virtual `per:nieces_and_nephews` slot.[8] The Slot Filling guidelines specify whether each of the component slots of a pseudo-slot is single-valued (*e.g.,* `per:date_of_birth`) or list-valued (*e.g.,* `per:employee_of`, `per:children`). A pseudo slot is single-valued if each of its component slots is single-valued, and list-valued otherwise. In contrast to the Slot Filling task, Cold Start submissions may contain multiple fills for single-valued slots. If such are present in the submission, LDC will assess the slot fill with the highest confidence value, and will assess additional slot fills if resources allow. If more than one such slot fill shares the same confidence value, the slot fill that appears earlier in the output will be considered to have higher confidence.

As with the Slot Filling task, the object of each component relation that makes up a single evaluation query response is rated as correct, inexact, or wrong. Pseudo-slots will be scored just as slots in the Slot Filling task, with the additional constraint that both the slot fill and the path leading to that fill must be correct for the entirety to be judged correct. To receive credit for identifying Maggie Simpson as Patty Bouvier's niece, the knowledge base must not only include Maggie as the slot fill, but must also represent Maggie as Marge's child, and Marge as Patty's sibling:[9]

Evaluation query:	Nieces and nephews of Patty Bouvier (`per:siblings`, `per:children`)
Ground Truth:	`:PattyBouvier per:siblings :MargeSimpson`
	`:MargeSimpson per:children :MaggieSimpson`
Submission:	`:PattyBouvier per:siblings :MargeSimpson`
	`:MargeSimpson per:children :MaggieSimpson` ⇒ **correct**

A KB that indicated that Maggie was Patty's niece because she was Patty's sister Selma's child would be scored as incorrect:

Evaluation query:	Nieces and nephews of Patty Bouvier (`per:siblings`, `per:children`)
Ground Truth:	`:PattyBouvier per:siblings :MargeSimpson`
	`:MargeSimpson per:children :MaggieSimpson`
Submission:	`:PattyBouvier per:siblings :SelmaBouvier`
	`:SelmaBouvier per:children :MaggieSimpson` ⇒ **incorrect**

A response is inexact if it either includes only a part of the correct answer or includes the correct answer plus extraneous material. No credit is given for inexact answers:

---

[8] A pseudo-slot is similar to the concept of a *role chain,* which is supported by some knowledge representation systems based on description logic, including OWL 2.

[9] In each of these examples, only the subject, predicate and object are shown, and only a subset of the relevant knowledge base is presented. Each entity is named after the mention that gave rise to it.

Evaluation query: Titles of parents of Bart Simpson (`per:parents`, `per:title`)
Ground Truth:
```
:BartSimpson per:parents :HomerSimpson
:HomerSimpson per:title "Attack-dog trainer"
```
Submission:
```
:BartSimpson per:parents :HomerSimpson
:HomerSimpson per:title "dog trainer Pitiless Pup"
```
⇒ **inexact**

In addition, the object of the *final* relation in a pseudo-slot may be rated as redundant if it is equivalent to another fill for the pseudo-slot. No credit is given for redundant answers:

Evaluation query: Nieces and nephews of Patty Bouvier (`per:siblings`, `per:children`)
Ground Truth:
```
:PattyBouvier per:siblings :MargeSimpson
:MargeSimpson per:children :MaggieSimpson
:MaggieSimpson per:alternate_names "Margaret Simpson"
```
Submission:
```
:PattyBouvier per:siblings :MargeSimpson
:MargeSimpson per:children :MaggieSimpson
```
⇒ **correct**
```
:MargeSimpson per:children :MargaretSimpson
```
⇒ **redundant**

However, objects of relations other than the final relation will never be rated as redundant:

Evaluation query: Nieces and nephews of Patty Bouvier (`per:siblings`, `per:children`)
Ground Truth:
```
:PattyBouvier per:siblings :MargeSimpson
:MargeSimpson per:children :LisaSimpson
:MargeSimpson per:children :BartSimpson
:MargeSimpson per:alternate_names "Marjorie Simpson"
```
Submission:
```
:PattyBouvier per:siblings :MargeSimpson
:PattyBouvier per:siblings :MarjorieSimpson
:MargeSimpson per:children :LisaSimpson
```
⇒ **correct**
```
:MarjorieSimpson per:children :BartSimpson
```
⇒ **correct**

Here, Marge Simpson and Marjorie Simpson represent the same person in the ground truth, but two distinct entities in the KB. However, because the query is about Marge's children and not about Marge herself, both responses to the evaluation query are assessed as correct.

Since in Cold Start the facts being evaluated come from sequences of triples, confidence scores would need to be combined if we wanted to generate confidence scores for a derived pseudo-relation. The proper way to combine scores of course depends on the meaning of those scores, and for now, Cold Start is not mandating any particular meaning. Three general score combination functions are min, max and product; we welcome comments from the community on which combination methods to report.

Scoring

Given the above approach to assessment, basic scoring for a given evaluation query proceeds as follows:

Correct = total number of system output pseudo-slots judged correct

System = total number of system output pseudo-slots

Reference = number of single-valued pseudo-slots with a correct response + number of equivalence classes[10] for all list-valued pseudo-slots

[10] See the Slot Filling Task Guidelines for further information on how and when two slot fills are treated as equivalent.

Recall = Correct / Reference

Precision = Correct / System

F = 2 * Precision * Recall / (Precision + Recall)

The F score is the primary metric for the 2013 Cold Start Knowledge Base Population system evaluation.

In 2012, a system received no credit for an evaluation query if it missed the mention of the starting entity, even if it had all of the relations represented correctly. To get a more nuanced view of a submitted knowledge base, in 2013 each evaluation query may have more than one instantiation. Each such instantiation will have the same relations, but will start at a different mention of the starting entity. To give equal weight to each evaluation query, the 'Correct' score for a single query will be the average of the 'Correct' scores for each of its variants. Put another way, evaluation queries will be macro-averaged across the variants.

Submissions

A two-week window from August 5 to August 19 will be available for downloading the Cold Start document collection, building the knowledge base, and submitting results. Systems should not be modified once the corpus has been downloaded. Participants may submit up to five knowledge bases, ranked in order of evaluation preference. The top-ranked submission must be made as a 'closed' system; in particular, it must not access the Web during the evaluation period. All submissions must obey the external resource restrictions in place for the TAC 2013 Slot Filling task. In addition, because Cold Start focuses on the condition where the knowledge base is initially empty, we ask that each participating site submit at least one run that consults external entity knowledge bases only after entities and relations have been extracted from the document collection. The number of submissions actually judged will depend upon resources available to NIST. Details about submission procedures will be communicated to the track mailing list. Tools to validate formats are available on the TAC Web site.

Sample Collection

A sample Cold Start collection will be available from the NIST Web site (`http://www.nist.gov/tac/2013/KBP/ColdStart/data.html`) shortly. Note that this is not a training collection; it serves only to illustrate the various facets of the task and the evaluation. The sample includes:

- A file describing the collection (`README.txt`).
- A document collection, comprising seventeen documents drawn from the domain of *The Simpsons* television show. Each `<DOC>` tag includes the original Web source, of which the text in the collection is a snippet.
- A reference KB for the collection. Note that a reference KB will not be created for the actual Cold Start task.
- A set of sample participant submissions, including a variety of errors.

Change History

- Version 1.1

    - Description of the use of the validator to convert from 2012 to 2013 format
    - Increase in size of document collection
    - Update of year to 2013 in document title
- Version 1.0
    - Original version, based on the 2012 specification
    - Improved description of evaluation queries
    - Changes to relations to be extracted (merge of `employee_of` and `member_of` slots)
    - Changes to offset specifications (conjoining of provenance offsets with dashes)
    - Addition of evaluation query variants