HLTCOE Participation in TAC KBP 2016: Cold Start and EDL

Tim Finin
University of Maryland
Baltimore County

Dawn Lawrie
Loyola University
Maryland

James Mayfield
Johns Hopkins University
HLTCOE

Paul McNamee
Johns Hopkins University
HLTCOE

Jessa Laspesa and Micheal Latman
Loyola University
Maryland

Abstract

The JHU HLTCOE participated in the Cold Start and the Entity Discovery and Linking tasks of the 2016 Text Analysis Conference Knowledge Base Population evaluation. For our fifth year of participation in Cold Start we continued our research with the KELVIN system. We submitted experimental variants that explore use of linking to Freebase across three languages and add relations beyond those required by Cold Start. This is our second year of participation in EDL. We used KELVIN in all runs.

1 Introduction

The JHU Human Language Technology Center of Excellence has participated in the TAC Knowledge Base Population exercise since its inception in 2009. Our focus over the past year was on developing our KELVIN system (McNamee et al., 2012; McNamee et al., 2013; Mayfield et al., 2014; Finin et al., 2014) as a core technology for multiple TAC tasks. This year we used KELVIN in both the Cold Start and the Entity Discovery and Linking (EDL) tasks.

This is the fifth year that we used KELVIN in the Cold Start task. This year we enhanced our system to allow for entities with no named mentions, modified the process of establishing links to freebase, and merged entities using information from across the knowledge base. We also enhanced translation to cope with the tri-lingual data set. We extended the system to be able to capture human annotations through the use of an iPad App. The App allows a user to view annotations that overlay the documents. The user can add, correct, and remove annotations.

In the rest of the paper we present our systems, which are architecturally similar to our 2015 submission, and briefly discuss our experimental results.

2 Cold Start KB Construction

The TAC-KBP Cold Start task is a complex task that requires application of multiple layers of NLP software. The most significant tool that we use is a NIST ACE entity/relation/event detection system, BBN SERIF (Ramshaw et al., 2011). SERIF provides a substrate that includes entity recognition, relation extraction, and within-document coreference resolution. In addition to SERIF, significant components that we rely on include: a maximum entropy trained model for extracting personal attributes (FACETS, also a BBN tool); a cross-document entity coreference resolver (the HLTCOE *Kripke* system); and a procedurally implemented rule system.

KELVIN is organized as a pipeline with three stages: (i) document level processing done in parallel on small batches of documents; (ii) cross-document coreference resolution to produce an initial KB; and (iii) knowledge-base enhancement and refinement through inference and relation analysis.

An optional fourth stage loads the knowledge base into an iPad app to collect human annotations on the document set. The next section describes the major steps in these stages.

3 Cold Start System Description

KELVIN runs from two Unix shell scripts[1] that execute a pipeline of operations. The input to the system is a file listing the source documents to be processed; the files are presumed to be plain UTF-8 encoded text, possibly containing light SGML markup. During processing, the system produces a series of tab-separated files, which capture the intermediate state of the growing knowledge base. At the end of the pipeline the resulting file is compliant with the Cold Start guidelines.

Our processing consists of the following steps, which are described in detail below:

1. Document-level processing
2. Extended Document-level processing
3. Cross-document entity coreference resolution
4. KB cleanup and slot value consolidation
5. Linking entity mention chains to an external background KB
6. Applying inference rules to posit additional assertions
7. KB-level entity clustering
8. KB cleanup and slot value consolidation
9. Selection of the best provenance metadata
10. Post-processing

The *Margaret* script performs the document-level processing in parallel on our Sun Grid Engine computing cluster. *Fanny* executes the balance of the pipeline; many of these steps are executed as a single process.

3.1 Document-Level Processing

BBN's SERIF tool[2] (Boschee et al., 2005) provides a considerable suite of document annotations that are an excellent basis for building a knowledge base. The functions SERIF can provide are based largely on the NIST ACE specification;[3] they include:

- identifying named entities and classifying them by type and subtype;
- performing intra-document coreference resolution, on named, nominal, and pronominal mention;
- parsing sentences and extracting intra-sentential relations between entities; and,
- detecting certain types of events.

We run each document through SERIF, and extract its annotations.[4] Additionally we run another module named FACETS, described below, which adds attributes about person entities. For each entity with at least one named mention, we collect its mentions and the relations and events in which it participates. Entities comprising solely nominal mentions were included in 2016 for both Cold Start and EDL, per the task guidelines. Finally, the output from each document is entered into a Concrete object, (Ferraro et al., 2014), which is our standard representation for information extracted from a document.

FACETS takes SERIF's analyses and produces role and argument annotations about person noun phrases. FACETS is implemented using a conditional-exponential learner trained on broadcast news. Attributes FACETS can recognize include general attributes like religion and age (which anyone might have), as well as some role-specific attributes, such as employer for someone who has a job, (medical) specialty for a physician, or (academic) affiliation for someone associated with an educational institution.

3.2 Extended Document-Level Processing

Five additional steps are taken once SERIF and FACETS are run. These steps generally address shortcomings in the tools or add additional information that was not found by the primary tools.

The first two steps focus on augmenting relations. In Step 1 relations are identified using pattern matching, which relies on entity type as well as string matches. In Step 2, new relations are found using an open information extraction system. Here facts are aligned to TAC relations using a bootstrapping approach from relations identified by both SERIF and FACETS as well as the Open IE system.

[1] Named Margaret and Fanny after Lord Kelvin's wives.

[2] Statistical Entity & Relation Information Finding

[3] http://www.itl.nist.gov/iad/mig/tests/ace/2008/doc/ace08-evalplan.v1.2d.pdf

[4] We used an in-house version of SERIF, not the annotations available from LDC.

Both of these steps are described in greater detail in a prior system description (Finin et al., 2015) and are referred to as *Extra Relations* when describing the experimental runs.

The third step focuses on refining canonical mentions. This approach uses Freebase to assist in the selection of descriptive names that do not contain ancillary information. We used a similar approach in the 2015 system (Finin et al., 2015); however, the implementation in 2016 was changed to improve the performance of the system.

In the fourth step dates identified by SERIF are modified to bring them into compliance with TAC guidelines. Problem with dates include the format when parts of the date are missing (*e.g.* "1948" rather than "1948-XX-XX") and the existence of relative dates rather than absolute dates.

Finally in the fifth step, entities from the headline, dateline, and author fields are extracted. In previous years the lack of identified entities from the fields led to a substantial number of misses in queries where annotators favored these mentions. This year the process of matching named entities was optionally extended to the rest of the document. This means that if exact matches of named entities already identified in the document are found, these new mentions will be added to the mention chain. This is referred to as *Exact Match* when describing the experimental runs.

3.3 Cross-Document Coreference Resolution

In 2013 we developed a tool for cross-document coreference named *Kripke* that takes as input a serialized TAC knowledge base and produces equivalence classes that encode entity coreference relations. *Kripke* is an unsupervised, procedural clusterer based on two principles: (a) to combine two clusters each must have good matching of both names and contextual features; and (b) a small set of discriminating contextual features is generally sufficient for disambiguation. To avoid the customary quadratic-time complexity required for brute-force pairwise comparisons, *Kripke* maintains an inverted index of names used for each entity. Only entities matching by full name, or some shared words or character n-grams are considered as potentially coreferential. While *Kripke*'s approach allows it to work well on many languages, in 2016 we made the n-gram length a language-dependent parameter, using a smaller value of n for Chinese mentions.

Contextual matching is based exclusively on named entities that co-occur in the same document. Between candidate clusters, the sets of all names occurring in any document forming each cluster are intersected. Each name is weighted by normalized Inverse Document Frequency, so that rare, or discriminating names have a weight closer to 1. The top-k (*e.g.,* k=10) weighted names were used, and if the sum of those weights exceeds a cutoff, then the contextual similarity is deemed adequate. This technique can distinguish George Bush the 41st U.S. president from his son, the 43rd U.S. president, through co-occurring names (*e.g.,* Al Gore, Barbara Bush, Kennebunkport, James Baker versus Dick Cheney, Laura Bush, Crawford, Condolezza Rice). The system runs by executing a cascade of clustering passes, where in each subsequent pass the requirements for sufficient name and contextual matching are relaxed. The higher precision matches made in earlier cascade phases facilitate more difficult matches in subsequent phases. Additional details can be found elsewhere (Finin et al., 2014; Finin et al., 2015).

3.4 KB Cleanup and Slot Value Consolidation

This step, which is repeated several times throughout the pipeline, ensures that the inverse of each relation is asserted in the KB, culls relations that violate type or value constraints, and reduces the number of values to match common sense expectations for each type of slot.

3.4.1 Inverses Relations

Producing inverses is an entirely deterministic process that simply generates Y *inverse* X in *Doc D* from an assertion of X *slot* Y in *Doc D*. For example, inverse relations like per:parent and per:children, or per:schools_attended and org:students. While straightforward, this is an important step, as relations are often extracted in only one direction during document-level analysis, yet we want both assertions to be explicitly present in our KB to aid with downstream reasoning.

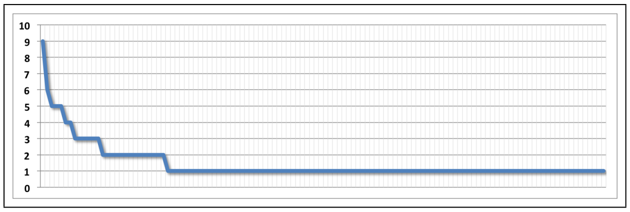


Figure 1: Kelvin initially extracted 121 distinct values for Barack Obama's employer from 26,000 Washington Post articles. The number of attesting documents for each followed a power law, with nine documents for the most popular value only one for the majority.

3.4.2 Predicate Constraints

Some extracted assertions can be quickly vetted for plausibility. For example, the object of a predicate expecting a country (*e.g.,* per:countries_of_residence) must match a small, enumerable list of country names; Massachusetts is not a reasonable response. Similarly, 250 is an unlikely value for a person's age. We have procedures to check certain slots to enforce that values are drawn from an accepted list of responses (*e.g.,* countries, religions), or cannot include responses from a list of known incorrect responses (*e.g.,* a girlfriend is not allowed as a slot fill for per:other_family).

3.4.3 Consolidating Slot Values

Extracting values for slots is a noisy process and errors are more likely for some slots than for others. The likelihood of finding incorrect values also depends on the popularity of both the entity and slot in the document collection. For example, in processing a collection of 26K articles from the Washington Post, we observed more than fifty entities who had 14 or more employers. One entity was reported as having had 122 employers (per:employee_of)!

Slot value consolidation involves selecting the best value in the case of a single valued slot (e.g., per:city_of_birth) and the best set of values for slots that can have more than one value (e.g, per:parents). In both cases, we use the number of attesting documents to rank candidate values, with greater weight given to values that were explicitly attested rather than implicitly attested via inference rules. See Figure 1 for the number of attesting documents for each of the values for the entity that had 122 distinct values for employer.

relation	many	maximum
per:children	8	10
per:countries_of_residence	5	7
per:employee_or_member_of	18	22
per:parents	5	5
per:religion	2	3
per:schools_attended	4	7
per:siblings	9	12
per:spouse	3	8

Table 1: The number of values for some multi-valued slots were limited by a heuristic process that involved the number of attesting documents for each value and two thresholds.

For slots that admit only a single value, we select the highest ranked candidate. However, for list-valued slots, it is difficult to know how many, and which values to allow for an entity. We made the pragmatic choice to limit list-values responses in a predicate-sensitive fashion, preferring frequently attested values. We associate two thresholds for selected list-valued predicates on the number of values that are reasonable – the first represents a number that is suspiciously large and the second is an absolute limit on the number of values reported. Table 1 shows the thresholds we used for some predicates. For predicates in our table, we accepted the nth value on the candidate list if n did not exceed the first threshold and rejected it if n exceeded the second. For n between the thresholds, a value is accepted only if it has more than one attesting document.

3.5 Inference

We apply a number of forward chaining inference rules to increase the number of assertions in our KB. To facilitate inference of assertions in the Cold Start schema, we introduce unofficial slots into our KB, which are subsequently removed prior to submission. For example, we add slots for the sex of a person, and geographical subsumption (*e.g.,* Gaithersburg is part-of Maryland). The most prolific inferred relations were based on rules for family relationships, corporate management, and geo-political containment.

Many of the rules are logically sound and follow directly from the meaning of the relations. For example, two people are siblings if they have a parent in common and two people have an "other_family" relation if one is a grandparent of the other. Our knowledge of geographic subsumption produced a large number of additional relations, e.g., knowing that a person's *city_of_birth* is Gaithersburg and that it is part of Maryland and that Maryland is a state supports the inference that the person's *stateorprovince_of_birth* is Maryland.

3.6 Linking to External Knowledge Bases

Entities are linked to one more external knowledge bases. Our current system uses just one external KB, the version of the Freebase KB described in Section 4. Our approach is relatively simple, only comparing an entity's type and mentions to the external KB's entity types, names and aliases.

In linking a collection entity to a KB entity, we start by producing a candidate set by selecting all KB entities whose names or aliases match any of the collection entity's canonical mentions.[5] The candidates are ranked by counting how often each matching mention was used and by the KB entity's *significance score* (see Section 4). We used experimentally derived thresholds to reject all candidates if there were too many or the top score was too low relative to the second highest score.

3.7 Knowledge-Level Clustering

After analyzing our previous Cold Start performance, we observed that KELVIN often undermerged entities. We added additional inference rules for merging entities that were applied at the knowledge-base level. One set of rules merges entities that are linked to the same Freebase entity. Another set merges entities that share the same canonical mention under several entity type specific conditions. For example, two ORG entities with subtype *Educational* are merged if they have the same canonical mention and the mention includes a token implying they are organizations of higher education (*e.g.,* college, university or institute).

A third set merges entities based on "discriminating relations." Our intuition is that it is likely that two people with similar names who have the same spouse or were born on the same date and in the same city should be merged. Similarly, organizations with similar names who share a top-level employee are good candidates for merging.

We maintain three categories of relations, those with high, medium and low discriminating power. Example of highly discriminating relations are per:children, org:date_founded, and gpe:part_of. Medium discriminating relations include per:city_of_birth, gpe:headquarters_in_city, and org:member_of. Examples of relations with low discriminating power include per:stateorprovince_-of_birth, org:students, and gpe:deaths_in_city. The decision to merge two entities with similar names is dependent on their type and the number of high, medium, and low discriminating relations they share.

3.8 Selecting Provenance Metadata

This step selects the provenance strings that will be used to support each relation for the final submission. The Cold Start evaluation rules allow for up to four provenance strings to support a relation, none of which can exceed 150 characters. For simple attested values, our initial provenance strings are spans selected from the sentence from which we extracted the relation, e.g., *"Homer is 37 years old"* for a per:age relation. Inferred relations can have more than one provenance string which can come from different documents, e.g., *"His daughter Lisa attends Springfield Elementary"* and *"Maggie's father is Homer Simpson"* for a per:siblings relation.

An initial step is to minimize the length of any overly-long provenance strings is to select a substring that spans both the subject and object. Candidate provenance strings whose length exceeds the maximum allowed after minimization are discarded.[6] If there are multiple provenance candidates, a simple greedy bin packing algorithm is used to include as many as possible into the four slots available. Preference is given for attested values over inferred values and provenance sources with higher certainty over a those with lower.

3.9 Cross-language Entity Linking

The overall processing has three stages: monolingual document processing, multilingual cross-

[5] Matching is done after normalizing strings by downcasing and removing punctuation.

[6] This could result in a relation being discarded if it has no legal provenance strings after minimization

entity	type	significance	inlinks	outlinks
United States	GPE	19.2	452006	162081
India	GPE	15.8	34273	23281
Harvard University	ORG	14.4	11163	11348
UMBC	ORG	7.4	172	192
Barack Obama	PER	11.4	744	1948
Alan Turing	PER	7.6	35	163
Ralph Sinatra	PER	2.8	0	7
Harvard Bridge	FAC	5.1	3	32
Mississippi River	LOC	8.9	242	245

Table 2: This table shows examples of entities, their estimated significance, and their number of incoming and outgoing links.

edl	initial	translate	final	final
run	entities	clusters	clusters	entities
1	1,258,324	62,025	58,674	532,515
2	1,258,324	65,973	63,646	565,521
3	1,258,324	62,055	58,686	532,886
4	1,258,324	65,930	63,654	565,899
5	1,258,324	62,060	58,663	532,513

Table 3: We combine clusters formed over all three languages by translating (where possible) non-English mentions into English (column 3) and clusters formed by considering each mono-lingual collection separately. The result produces a larger number of clusters (column 4) and a greater reduction in the total number of entities (column 5).

document coreference resolution, and multilingual knowledge-base processing.

The first stage applies Kelvin's standard pipeline to each of the three monolingual document collections using the appropriate SERIF language model.[7] For each language, we use just two outputs of the monolingual system: the serialized TAC KB produced by KELVIN's document level processing and the coreference relations produced by *Kripke*.

The second stage starts by creating a multilingual document level KB by concatenating the three monolingual KBs. If the mention-translation option is enabled, English translations of Chinese and Spanish mentions are added. We used the Bing translation service API. This combined, multilingual collection is then processed by Kripke to produce cross-document coreference relations.

The coreference relations from each of the monolingual collections and from the combined collection are integrated using a simple algorithm to combine equivalence relations, yielding a single coreference clustering file for the entire collection. The three monolingual document-level KBs are then combined (without any translated mentions) and the cross-document coreference relations used to generate the KB for subsequent KB-level processing by the rest of Kelvin's pipeline. Combining the results of using Kripke to cluster the mono-lingual runs and separately their combination with mentions translated into English achieves a greater reduction in the number of entities. Table 3 shows the effects of combining these equivalence sets on our five EDL runs.

The remaining processing, including linking, was performed by Kelvin's pipeline with a few small additions. We added a special module to find and extract authors' names of posts in Bolt documents. Our document-level processing typically produced longer nominal mentions than were allowed under track guidelines. In 2015 we reduced nominal mentions to head noun and any immediate nominal modifiers but for 2016 we further reduced them to their head noun. This was done with a simple procedure to find the the first sequence of consecutive tokens tagged as *NN*, *NNS*, *NNP* or *NNPS* and then select the last token. Examples of our adjusted nominal mentions for English include:

- a former two-term Florida governor ⟶ governor

[7] Our version of SERIF has models for English, Chinese, Spanish and Arabic

- the most formidable fundraiser in the Republican field ⟶ fundraiser

- Republican Congressman from New York ⟶ Congressman

- the Greek minister of Productive Reconstruction, Environment and Energy ⟶ minister

The 2016 task also required that nominal mentions be limited to references to singular entities and to those referring to specific, real-world entities. We used a combination of features to judge whether the nominal mention was singular, including information about its entity's other mentions (if any), its POS tag, and human judgments for a set of English words frequently observed as nominal mentions (e.g., congressman, rich, mankind). We assumed that a nominal mention was specific unless it appeared on a short list of English candidates we observed that we judged to be non-specific (e.g., someone, nobody and few).

3.10 Post-Processing

The final steps in our pipeline produce several outputs, including a submission file that complies with Cold Start task guidelines and an RDF version that is can be loaded into a triple store for inspection and querying.

We start by normalizing temporal expressions, ensuring that all entities have mentions, insisting that relations are consistent with the types of their subjects and objects, confirming that logical inverses are asserted, and checking that entities have mentions in the provenance documents.

We then translate the KB from TAC format to RDF using an OWL ontology that encodes knowledge about the concepts and relations, both explicit and implicit. For example, the Cold Start domain has an explicit type for geo-political entities (GPEs), but implicitly introduces disjoint GPE subtypes for cities, states or provinces, and countries through predicates like *city_of_birth*. Applying an OWL reasoner to this form of the KB detects various logical problems, e.g., an entity is being used as both a city and a country.

The RDF KB results are also loaded into a triple store, permitting access by an integrated set of standard RDF tools including Fuseki for SPARQL queries (Prud'Hommeaux and Seaborne, 2008), Pubby for browsing, and the Yasgui SPARQL GUI (Rietveld and Hoekstra, 2013).

4 External Knowledge Base

We created an external knowledge base derived from the BaseKB version of Freebase that was distributed by the LDC for use in the 2015 TAC KBP EDL tasks.[8] This external KB supported both our Cold Start and EDL submissions.

The full BaseKB dataset is quite large, containing more than a billion facts (counting each triple as a fact) about more than 40 million subjects. Much of this information is not relevant to the KBP tasks, such as information about musical groups, films or fictional characters.

We started by identifying entities that might be relevant to the TAC KBP tasks and removing any triples whose subjects were not in this set. An initial step was to identify those subjects that mapped to one of the five standard TAC types (PER, ORG, GPE, LOC and FAC) or represented what Freebase calls a Compound Value Type (CVT). The TAC ontology assumes that its five types are disjoint, but relevant Freebase entities can have types that map to several TAC types. For example, the Freebase entity with canonical name *Oval Office* (m.01hhz7) has subtypes associated with both a LOC and an ORG. We used various heuristics to assign such entities to only one TAC type.

We kept information about any CVTs that were linked to a TAC-relevant entity. CVTs are used in Freebase to represent reified relations, such as relations with associated units (for measurements), time or location.

Triples with literal values (i.e., strings) for objects are tagged with an XSD data type (e.g., integer or date) or a language tag (e.g., @EN for English or @ZH for Chinese). We discarded any string values whose language tag was not in the English, Chinese, or Spanish families.

We computed a measure of an entity's *significance* based on the number on triples in which it was the subject or object. The significance was set as the base-2 log of the total number of links, which pro-

[8] The dataset is available from the Linguistic Data Consortium as LDC2015E42

	0-hop				1-hop				All-hop			
Run	GT	R	W	D	GT	R	W	D	GT	R	W	D
E1	801	194	367	16	408	47	489	2	1209	241	856	18
E2	801	185	313	16	408	41	356	2	1209	226	669	18
E3	801	186	311	13	408	41	362	4	1209	227	673	17
E4	801	186	393	12	408	40	919	2	1209	226	1312	14
E5	801	195	366	15	408	47	489	2	1209	242	855	17
C1	751	141	125	16	230	57	134	10	981	198	259	26
C2	751	141	125	16	230	57	134	10	981	198	259	26
C3	751	141	125	16	230	57	134	10	981	198	259	26
C4	751	143	127	14	230	57	139	10	981	200	266	24
S1	332	4	9	2	213	0	0	0	545	4	9	2
S2	332	4	9	2	213	0	0	0	545	4	9	2
S3	332	3	3	1	213	0	0	0	545	3	3	1
S4	332	4	9	2	213	0	0	0	545	4	9	2
X1	4094	838	1177	96	1965	351	1956	52	6059	1189	3133	148
X2	4094	713	1107	110	1965	272	1196	62	6059	985	2303	172
X3	4094	697	936	116	1965	264	1021	65	6059	961	1957	181
X4	4094	321	385	33	1965	106	490	17	6059	427	875	50

Table 4: Ground-truth, right, wrong and duplicate answers for our submitted 2015 runs.

	0-hop			1-hop			All-hop		
Run	P	R	F1	P	R	F1	P	R	F1
E1	0.3458	0.2422	0.2849	0.0877	**0.1152**	0.0996	0.2197	0.1993	0.2090
E2	0.3715	0.2310	0.2848	**0.1033**	0.1005	**0.1019**	**0.2525**	0.1869	0.2148
E3	**0.3742**	0.2322	**0.2866**	0.1017	0.1005	0.1011	0.2522	0.1878	**0.2153**
E4	0.3212	0.2322	0.2696	0.0417	0.0980	0.0585	0.1469	0.1869	0.1645
E5	0.3476	**0.2434**	0.2863	0.0877	**0.1152**	0.0996	0.2206	**0.2002**	0.2099
C1	**0.5301**	0.1877	0.2773	**0.2984**	**0.2478**	**0.2708**	**0.4333**	0.2018	0.2754
C2	0.5301	0.1877	0.2773	0.2984	0.2478	0.2708	0.4333	0.2018	0.2754
C3	0.5301	0.1877	0.2773	0.2984	0.2478	0.2708	0.4333	0.2018	0.2754
C4	0.5296	**0.1904**	**0.2801**	0.2908	0.2478	0.2676	0.4292	**0.2039**	**0.2764**
S1	0.3077	**0.0120**	**0.0232**	0.0000	0.0000	0.0000	0.3077	**0.0073**	**0.0143**
S2	0.3077	0.0120	0.0232	0.0000	0.0000	0.0000	0.3077	0.0073	0.0143
S3	**0.5000**	0.0090	0.0178	0.0000	0.0000	0.0000	**0.5000**	0.0055	0.0109
S4	0.3077	0.0120	0.0232	0.0000	0.0000	0.0000	0.3077	0.0073	0.0143
X1	0.4159	**0.2047**	**0.2743**	0.1521	**0.1786**	**0.1643**	0.2751	**0.1962**	**0.2291**
X2	0.3918	0.1742	0.2411	0.1853	0.1384	0.1585	0.2996	0.1626	0.2108
X3	0.4268	0.1702	0.2434	**0.2054**	0.1344	0.1625	**0.3293**	0.1586	0.2141
X4	**0.4547**	0.0784	0.1337	0.1779	0.0539	0.0828	0.3280	0.0705	0.1160

Table 5: Micro precision, recall and F_1 scores for our submitted 2015 runs.

duced values between 1.0 and 20.0 for the reduced KB Table 2 shows data for a few example entities.

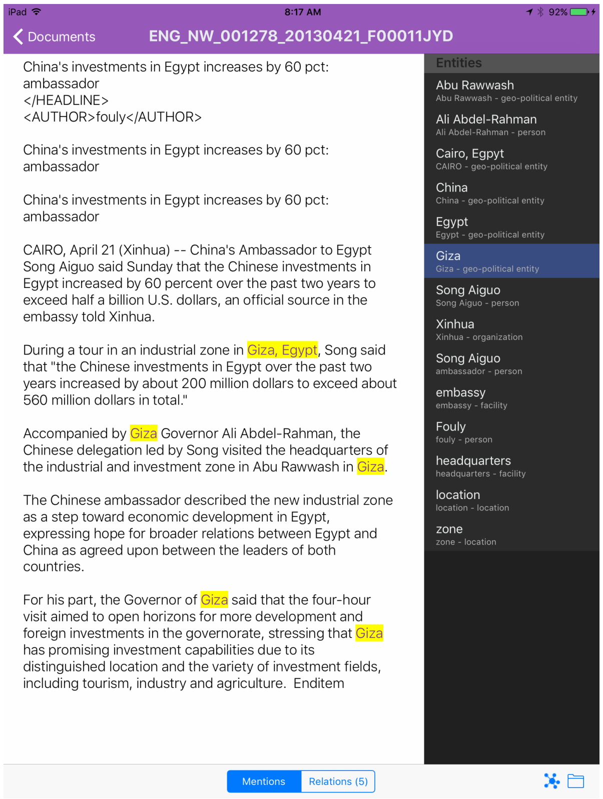
Finally, we added additional assertions to record an entity's TAC type and normalized versions of an entity's names and aliases by downcasing, removing punctuation, entity significance, number of in- and out-links, etc. The reduced KB has 146M triples over more than 4.5M TAC entities: 3074k PERs, 686k ORGs, 539k GPEs, 161k FACs and 85k LOCs. It was loaded into a triple store with SPARQL endpoint using the Apache Jena suite of RDF tools: Jena, Fuseki and TDB.

5 Manual Annotations

Building a knowledge base is not necessarily a process that should be fully automatic, as a person using such a system might want to remove inconsistencies and erroneous facts in the knowledge base. To incorporate human annotations an intuitive interface is essential. With this iteration of our Cold Start system we introduced *IrisXRef*.

IrisXRef is an iPad app built using Apple's Swift programming language. The app accesses a MySQL database, which stores a representation of the knowledge base in nine relational tables. Communication between the app and the database is facilitated by PHP scripts that send and receive JSON objects. Python scripts are used to load a knowledge base written in the TAC format into the database and to extract the knowledge base from the database and write it as a TAC file.

The app allows multiple projects built over the same or different document sets to be accessed simultaneously. Within each project, the user is presented with a list of documents referenced by document id. This list can be searched to identify a specific document. Within a document, the user views the full text in one view, and the list of its entities in an adjacent view; Figure 2 shows an example. A document entity can be selected or deleted from this view. When an entity is selected, the mentions of the entity are highlighted in yellow. Tapping on a highlighted mention allows the user to delete the mention. A second view, reveals asserted relations highlighted in green. Erroneous relations can be removed by tapping on the highlighted text, which displays the window shown in Figure 3, and then tapping the red button associated with the incorrect relation. New relations can be added by highlighting text and then selecting "Relation" from the pop-up menu, as is shown in Figure 4. A user specifies the relation that the string asserts and, if necessary, the secondary entity involved in the relation. The pop-up menu also facilitates adding new mentions with the "Highlight" option and adding new entities with the "Add New Entity" option.



China's investments in Egypt increases by 60 pct:
ambassador
</HEADLINE>
<AUTHOR>fouly</AUTHOR>

China's investments in Egypt increases by 60 pct:
ambassador

China's investments in Egypt increases by 60 pct:
ambassador

CAIRO, April 21 (Xinhua) -- China's Ambassador to Egypt Song Aiguo said Sunday that the Chinese investments in Egypt increased by 60 percent over the past two years to exceed half a billion U.S. dollars, an official source in the embassy told Xinhua.

During a tour in an industrial zone in Giza, Egypt, Song said that "the Chinese investments in Egypt over the past two years increased by about 200 million dollars to exceed about 560 million dollars in total."

Accompanied by Giza Governor Ali Abdel-Rahman, the Chinese delegation led by Song visited the headquarters of the industrial and investment zone in Abu Rawwash in Giza.

The Chinese ambassador described the new industrial zone as a step toward economic development in Egypt, expressing hope for broader relations between Egypt and China as agreed upon between the leaders of both countries.

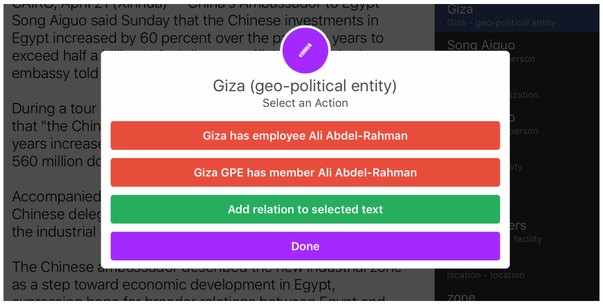
For his part, the Governor of Giza said that the four-hour visit aimed to open horizons for more development and foreign investments in the governorate, stressing that Giza has promising investment capabilities due to its distinguished location and the variety of investment fields, including tourism, industry and agriculture. Enditem

Abu Rawwash
Abu Rawwash - geo-political entity
Ali Abdel-Rahman
Ali Abdel-Rahman - person
Cairo, Egpyt
CAIRO - geo-political entity
China
China - geo-political entity
Egypt
Egypt - geo-political entity
Giza
Giza - geo-political entity
Song Aiguo
Song Aiguo - person
Xinhua
Xinhua - organization
Song Aiguo
ambassador - person
embassy
embassy - facility
Fouly
fouly - person
headquarters
headquarters - facility
location
location - location
zone
zone - location

Figure 2: *IrisXRef* display of a document with the mentions of "Giza" highlighted.

To facilitate speedier entry of new relations, we introduced a Naive Bayes Classifier that guesses relations and entities that are their arguments. Rather than using the automated system as training data, the system is continuously trained based on new relations that user has added. The classifier does a good job with small amounts of training data.

Another feature of the app is that it displays a network diagram of the entities and relationships in the document. This feature uses the online social network visualizer Cytoscape (Shannon et al., 2003). The visualization displays the interconnect-



Song Aiguo said Sunday that the Chinese investments in Egypt increased by 60 percent over the p years to exceed half a

embassy told

During a tour

that "the Chin

years increase

560 million do

Accompanied

Chinese deleg

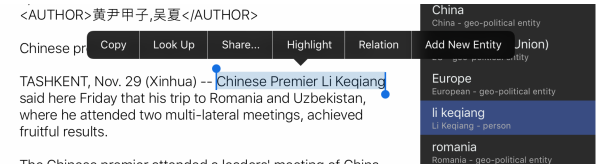
the industrial

The Chinese

as a step toward economic development in Egypt,

Giza (geo-political entity)
Select an Action
Giza has employee Ali Abdel-Rahman
Giza GPE has member Ali Abdel-Rahman
Add relation to selected text
Done
Giza
Giza - geo-political entity
Song Aiguo
erson
ization
person
ity
ers
facility
location - location
zone

Figure 3: Window of the relations asserted in a span. The buttons allow relations to be removed and added.



<AUTHOR>黄尹甲子,吴夏</AUTHOR>

Chinese pr

TASHKENT, Nov. 29 (Xinhua) -- Chinese Premier Li Keqiang said here Friday that his trip to Romania and Uzbekistan, where he attended two multi-lateral meetings, achieved fruitful results.

Figure 4: View of the pop-up menu that appears when text is highlighted to facilitate adding mentions, relations, and new entities.

edness found in most newswire documents given the TAC relations, and how many relations are being missed by the KELVIN system as is shown in the before and after images for one of the Cold Start documents in Figure 5.

6 Cold Start Submissions and Results

This year we submitted runs for the cross-language knowledge base population task in all three included languages and in the cross-lingual category. For the English task, we submitted the maximum of five experimental conditions. For the Chinese, Spanish, and cross-language tasks, we submitted four experimental conditions each.

For English, our most complex automated run was Run 1; it included link merging (see Section 3.6), knowledge base merging (see Section 3.7, and additional relation extraction techniques (see Section 3.2. Each subsequent run removed one of the conditions; thus, Run 2 does not include additional extraction, while Run 3 excluded additional and KB merging. Run 5 was a partially manual run where it started with Run 1. Then one of the authors spent approximately 4 hours fixing errors and introducing new mentions and relations using *IrisXRef* (see Section 5).

For Chinese, two different features were exercised in the experimental runs: KB merging from Section 3.7 and Super Kripke described in Section 3.3. Run 1 made use of neither of these features, Runs 2 and 3 both utilized KB merging, and Run 4 included both KB merging and Super Kripke.

For Spanish, link merging based described in 3.6 was added to the two features included in Chinese. Run 1 only included link merging. Run2 included link and KB merging. Run 3 had all the features turned off, and Run4 included all the features.

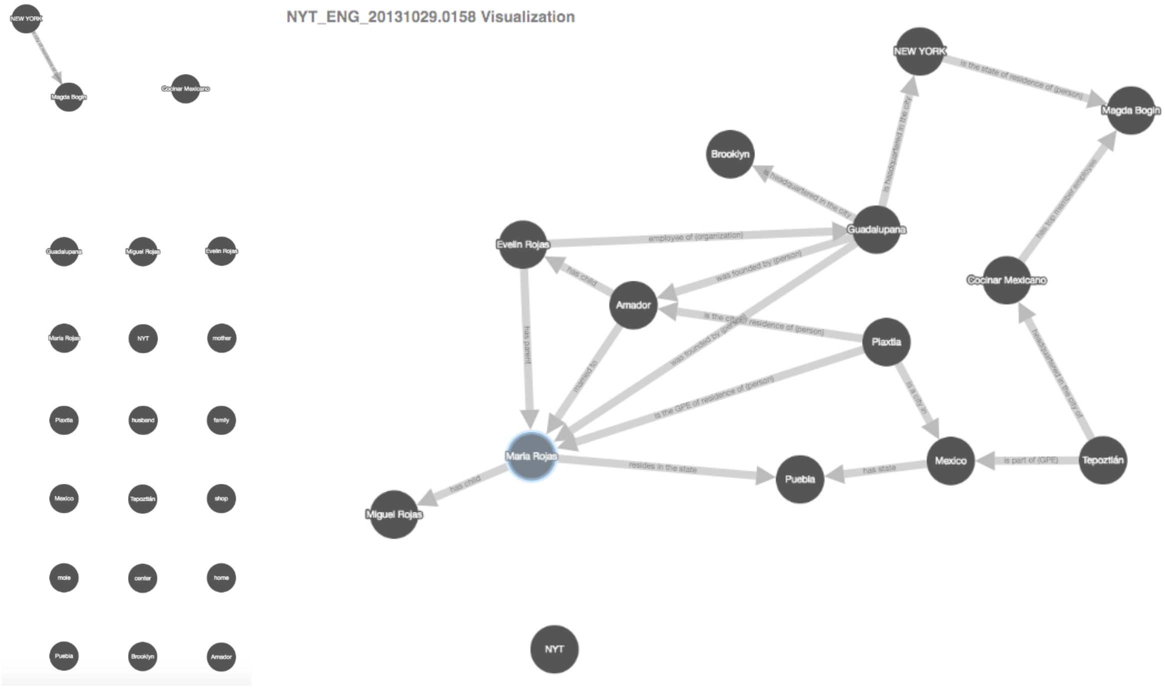
For the cross-lingual runs, the particular monolingual runs used was one of the experimental features and the use of translation was the other feature. In the first and second cross-lingual runs, the second monolingual runs were used as the source documents. In the third and fourth cross-lingual runs, the first monolingual runs were used as the source documents. In either case, the document level entities were used as the starting point, so that knowledge base entities formed by *Kripke* in the multilingual environment. The second condition determined whether translated entity names were added as mentions to the Chinese and Spanish document entities as is described in Section 3.9. In Runs 2 and 4, translations were added. They were not added in Runs 1 and 3.

Table 6 summarizes the various conditions, and Tables 4 and 5 give the key performance metrics. In Tables 4 and 5 run names are abbreviated to a single letter, where the first letter of the language is used for English, Chinese, and Spanish, and X is used to identify cross-lingual runs. Here we focus on the slot filling evaluation since the entity linking ability of Kelvin is evaluated in the discussion of EDL.

6.1 Discussion

The analysis in this section focuses on the monolingual runs before turning to the cross-lingual runs. Comparing our various experimental conditions, we make the following observations.

First in English, from Table 5 the additional relations that are added with the extra relations have a positive impact on recall but a negative on precision. This impact is present for both 0 and 1 hop queries. Given that the precision is more negatively impacted than the recall is positively impacted, the overall effect as measured by F1 indicates that these added



(a) before

(b) after

Figure 5: Entity network (a) before and (b) after human annotation of the document

relations are not an improvement. Although only a small percent of the document were impacted by the manual annotations in Run 5, there were slight improvements at the 0-hop level.

In Chinese, using KB merging has no impact on performance. The application of super Kripke improves recall slightly with an even smaller negative impact precision.

In Spanish, linking to an external knowledge base appears to be as effective as it is in English for precision; however, unlike English, it seems to be accompanied with a negative impact on recall. Looking at the overall performance in Spanish, the system greatly under preforms which makes it challenging to draw any strong conclusions about the experimental conditions.

For the multi-lingual KB, translating the entity names suppresses recall independent of the monolingual runs used as a starting place. In addition, starting with the second runs also leads to higher recall. The precision results are less clear. Here is appears that using the first monolingual runs yields higher precision. This is counter intuitive since for Runs 1 and 2 on both Chinese and Spanish there was no observable difference in performance, and for English the first run had a lower precision than the second run. There are, however, differences in the runs as is evident from Table 7. This table also reveals that there are more entities and more facts for Run 4. This may indicate that translation is not helping to merge more entities. This larger number of entities is favorably impacting precision.

7 EDL submissions and results

We submitted five EDL runs, all of which used KELVIN for the processing and then converted the knowledge base to the output required for the EDL task. The knowledge base was built over the entire Cold Start KBP data set. When generating the EDL output, the mentions from documents not part of the EDL dataset were excluded. None of the runs used links to Wikipedia in the reference, used relations encoded in the reference KB, or attempted to generate meaningful confidence values.

The experimental conditions investigated the usefulness of translation as a way to create more commonalities between names when performing cross-

Name	Link Merging	KB Merging	Extra Relations	Manual Fixes
hltcoe_ENG_1	Yes	Yes	Yes	
hltcoe_ENG_2	Yes	Yes		
hltcoe_ENG_3	Yes			
hltcoe_ENG_4				
hltcoe_ENG_5	Yes	Yes	Yes	Yes

Name	KB Merging	Super Kripke
hltcoe_CMN_1		
hltcoe_CMN_2	Yes	
hltcoe_CMN_3	Yes	
hltcoe_CMN_4	Yes	Yes

Name	Link Merging	KB Merging	Super Kripke
hltcoe_SPA_1	Yes		
hltcoe_SPA_2	Yes	Yes	
hltcoe_SPA_3			
hltcoe_SPA_4	Yes	Yes	Yes

Name	Source Runs	Translation
hltcoe_XLING_1	E2/C2/S2	
hltcoe_XLING_2	E2/C2/S2	Yes
hltcoe_XLING_3	E1/C1/S1	
hltcoe_XLING_4	E1/C1/S1	Yes

Table 6: Experimental variables for submitted Cold Start KBP runs.

document coreference as is described in Section 3.9; the usefulness of including knowledge base merging of entities as is described in Section 3.7; and the usefulness of searching the document for more mentions by using exact matches of names discovered by SERIF in the documents. While the first two conditions are applied to the multi-lingual knowledge base, the third condition applies to the document processing step which occurs in a monolingual environment. Therefore, two monolingual KBs were created for each language to support the cross-language EDL task. Table 9 shows the effect of adding exact mention matches to the KB where the second run in each language includes the added mentions. Surprisingly, the number of mentions decreases when for Spanish and Chinese when this approach is used. It does, however, have the expected outcome of adding mentions in English.

Turning to the particular configurations of the experimental runs, Run 1 used the Bing translation service to translate mentions from Chinese and Spanish to English, KB merging, and no extra exact match mentions. Run2 was the same as Run1 except it did not utilized translation. Run 3 used the Bing translation service, but it did not utilize KB merging. Run 4 was the same as Run 3 expect it did not utilize translation. Finally, Run 5 was just like Run 1 except it added mentions that were exact matches of names already known in the document to that mention chain. Table 8 summarizes the experimental conditions of the runs.

Table 10 shows the precision, recall and F_1 scores for each multilingual run for three key metrics: *strong typed mention match* (a measure of NER effectiveness) , *strong all match* (a measure of linking performance) and *mention ceaf* (a measure for clustering).

7.1 Discussion

When examining the scores, the scores do reveal many differences among the experimental conditions. Adding more mentions had the biggest impact of the scores. Strangely, it improved the precision as the expense of recall for all reported metrics. This likely is because the number of mentions for Spanish and Chinese actually decreased for this condition.

One of the highlights of the approach is the ability to do NIL clustering, where the system gets its highest recall values. This is because NIL linking is not a special case of linking but rather the default scenario, since cross-document coreference is performed prior to any clustering of the entities with an external knowledge base.

8 Achieving Portability with Docker

Kelvin is a large system with a number of components written in different programming languages, several services (e.g., our Freebase KB server), and datasets of various sizes. Together they represent hundreds of files spread over the shared Unix file

run	entities	PER	ORG	GPE	LOC	FAC	facts
E1	209871	107344	51255	15036	14164	22070	165946
E2	209952	107353	51318	15041	14168	22070	150981
E3	211891	136957	52460	15277	14170	22073	180402
E4	136159	76858	37245	12538	3480	6036	151868
E5	209903	107359	51260	15051	14162	22069	165939
C1	215901	106349	50755	24544	17773	16478	106940
C2	215959	106382	50773	24551	17773	16478	106954
C3	215900	106347	50760	24540	17773	16478	106942
C4	141094	76310	33292	17277	5732	8481	104372
S1	130958	56450	48655	9455	16396	0	29792
S2	133363	57056	50142	9755	16408	0	30288
S3	157709	61286	55420	23992	17009	0	32726
S4	124538	55016	47421	6235	15864	0	30302
X1	596042	341385	162082	69570	48919	38632	297323
X2	546906	264342	148811	47182	48036	38533	285419
X3	546880	264324	148788	47195	48037	38534	300419
X4	529357	257228	142696	44031	47195	38205	294101

Table 7: Number of entity mentions and facts identified in the evaluation corpus for each run.

Name	Translation	KB Merging	Exact Mention Match
hltcoe1	Yes	Yes	
hltcoe2		Yes	
hltcoe3	Yes		
hltcoe4			
hltcoe5	Yes	Yes	Yes

Table 8: Experimental variables for submitted Entity Linking and Discovery runs.

system at the HLTCOE. This has made it difficult, at best, to port to a different Unix installation and very challenging to run it on Windows or a Macintosh.

In 2016 we used Docker (Merkel, 2014) containers to package our system as several Docker images. We make use of Concrete as a common, open sourced representation for information about a single communication (e.g., a document, email message or social media post) using a well defined and published schema (Ferraro et al., 2014; Human Language Technology Center of Excellence, 2015). The Concrete schema is realized as a set of Apache Thrift (Thrift, 2016) schema files, which allows us to generate schema-specific classes for popular programming languages, including Java, Python, C++ and JavaScript.

The simplest instance of a Concrete communication might only hold the communication's text. A more elaborate example will also hold a language identifier, tokenizations, segmentations, syntactic information, mentions, entities, relations, and other information supported by Concrete's schema.

Our two basic Docker containers are Serif and Kelvin. Serif takes in a compressed tar file of raw text files or Concrete objects, creates Concrete objects if necessary, adds information by running Serif and Facets and mapping their output into Concrete's schemas and produces a compressed tar file of Concrete objects as output. Our use of Concrete objects as a standard representation for information extracted from a document enables a host of other tools to take these object as input and add additional features to them, for example, running additional relation extraction systems.

The Kelvin Docker container takes a compressed tar file of Concrete objects and a configuration file and processes the documents with the Kelvin pipeline, producing one or more output files (e.g., TAC, EDL or RDF). By using Docker, we are now

run	entities	PER	ORG	GPE	LOC	FAC	facts	mentions
E1	430306	133809	129363	124326	18608	24199	1058638	780475
E2	430306	133809	129363	124326	18608	24199	1058638	789502
C1	471651	147191	126559	150887	28062	18951	1071980	791435
C2	471651	147191	126559	150887	28062	18951	1071980	787239
S1	356370	91373	124517	107828	32651	0	731922	688343
S2	356370	91373	124517	107828	32651	0	731922	654788
X1	521903	257088	137401	43503	45891	38017	4124418	2601553
X2	543254	265431	145567	46881	47061	38311	4237003	2601553
X3	526337	258367	139576	43932	46336	38123	4147872	2601217
X4	548258	266736	148150	47356	47583	38430	4262444	2601217
X5	522228	257303	137493	43519	45891	38019	4125865	2572829

Table 9: Number of entity mentions and facts identified in the evaluation corpus for each run.

	NER			Linking			Nil Linking			Clustering		
#	pre	rec	F1	pre	rec	F1	pre	rec	F1	pre	rec	F1
1	0.656	0.573	0.612	0.489	0.427	0.456	0.368	0.614	0.460	0.470	0.411	0.438
2	0.656	0.573	0.612	0.488	0.426	0.455	0.335	0.624	0.436	0.469	0.410	0.437
3	0.656	0.573	0.612	0.476	0.416	0.444	0.346	0.615	0.443	0.457	0.399	0.426
4	0.656	0.573	0.612	0.489	0.427	0.456	0.335	0.625	0.436	0.469	0.410	0.438
5	0.661	0.563	0.608	0.494	0.420	0.454	0.374	0.612	0.464	0.474	0.404	0.436

Table 10: This table shows the precision, recall and F_1 measures over all three languages for each run for four key metrics: strong typed mention match, strong all match, strong nil match, and mention ceaf plus.

able to easily port Kelvin to any computer that has Docker installed.

9 Conclusion

The JHU Human Language Technology Center of Excellence has participated in the TAC Knowledge Base Population exercise since its inception in 2009, in Cold Start task since 2012, and in Entity Discovery and Linking the last two years. We modified the KELVIN system used in the 2015 Cold Start and EDL tasks by refining our Freebase based linking system and our knowledge based approach to entity coreference resolution. To support the cross-lingual task in Cold Start with relied primarily on the modifications made to KELVIN for Entity Discovery and Linking in 2015. These were further enhanced with a greater focus on translation, entity linking, and handling nominal mentions.

References

Adrian Benton, Jay Deyoung, Adam Teichert, Mark Dredze, Benjamin Van Durme, Stephen Mayhew, and Max Thomas. 2014. Faster (and better) entity linking with cascades. In *NIPS Workshop on Automated Knowledge Base Construction*.

E. Boschee, R. Weischedel, and A. Zamanian. 2005. Automatic information extraction. In *Int. Conf. on Intelligence Analysis*, pages 2–4.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity disambiguation for knowledge base population. In *Conf. on Computational Linguistics*.

Francis Ferraro, Max Thomas, Matt Gormley, Travis Wolfe, Craig Harman, and Benjamin Van Durme. 2014. Concretely annotated corpora. In *4th Workshop on Automated Knowledge Base Construction*.

Tim Finin, Paul McNamee, Dawn Lawrie, James Mayfield, and Craig Harman. 2014. Hot stuff at cold start: HLTCOE participation at TAC 2014. In *7th Text Analysis Conf.*, Nov.

Tim Finin, Dawn Lawrie, Paul McNamee, James Mayfield, Doug Oard, Nanyun Peng, Ning Gao, Yiu-Chang

Lin, Josh MacKin, and Tim Dowd. 2015. HLTCOE participation at TAC KBP 2015: Cold start and TEDL. In *8th Text Analysis Conf.*, Nov.

Human Language Technology Center of Excellence. 2015. Concrete repository. http://hltcoe.github.io.

Dawn Lawrie, Tim Finin, James Mayfield, and Paul McNamee. 2013. Comparing and Evaluating Semantic Data Automatically Extracted from Text. In *AAAI 2013 Fall Symposium on Semantics for Big Data*. AAAI Press, Nov.

James Mayfield and Tim Finin. 2012. Evaluating the quality of a knowledge base populated from text. In *Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*. ACL.

James Mayfield, Paul McNamee, Craig Harmon, Tim Finin, and Dawn Lawrie. 2014. KELVIN: Extracting Knowledge from Large Text Collections. In *AAAI Fall Symposium on Natural Language Access to Big Data*. AAAI Press, November.

Paul McNamee, Veselin Stoyanov, James Mayfield, Tim Finin, Tim Oates, Tan Xu, Douglas W. Oard, and Dawn Lawrie. 2012. HLTCOE participation at TAC 2012: Entity linking and cold start knowledge base construction. In *5th Text Analysis Conf.*, Gaithersburg, MD, Nov.

Paul McNamee, James Mayfield, Tim Finin, Tim Oates, Baltimore County, Dawn Lawrie, Tan Xu, and Douglas W Oard. 2013. KELVIN: a tool for automated knowledge base construction. In *NAACL-HLT*, volume 10, page 32.

Dirk Merkel. 2014. Docker: lightweight linux containers for consistent development and deployment. *Linux Journal*, 2014(239):2.

E Prud'Hommeaux and A. Seaborne. 2008. SPARQL query language for RDF. Technical report, World Wide Web Consortium, January.

Lance Ramshaw, Elizabeth Boschee, Marjorie Freedman, Jessica MacBride, Ralph Weischedel, and Alex Zamanian. 2011. Serif language processing effective trainable language understanding. *Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation*, pages 636–644.

Laurens Rietveld and Rinke Hoekstra. 2013. Yasgui: Not just another sparql client. In Philipp Cimiano, Miriam Fernández, Vanessa Lopez, Stefan Schlobach, and Johanna Völker, editors, *The Semantic Web: ESWC 2013 Satellite Events*, volume 7955 of *Lecture Notes in Computer Science*, pages 78–86. Springer Berlin Heidelberg.

Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. 2003. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11):2498–2504.

V. Stoyanov, J. Mayfield, T. Xu, D.W. Oard, D. Lawrie, T. Oates, T. Finin, and B. County. 2012. A context-aware approach to entity linking. *Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction, NAACL-HLT*.

Apache Thrift. 2016. Apache thrift website. https://thrift.apache.org/.