Overview of the TAC2013 Knowledge Base Population Evaluation: English Slot Filling and Temporal Slot Filling

Mihai Surdeanu

School of Information: Science, Technology, and Arts
University of Arizona, Tucson, AZ, USA
msurdeanu@email.arizona.edu

Abstract

We overview two tracks of the TAC2013 Knowledge Base Population (KBP) evaluation: English slot filling (SF) and temporal slot filling (TSF). The goal of these two KBP tracks is to promote research in the extraction of relations between named entities from free text (SF), and identify the temporal intervals when these relations were valid (TSF). The main changes this year include the requirement for a stricter textual justification of the extracted relations for SF, and a simplification of the TSF task, where the relation to be temporally grounded is given as input. These two tracks attracted 43 registered teams, out of which 20 teams submitted a run in at least one of the tracks.

1 Introduction

The main goal of the Knowledge Base Population track at the Text Analysis Conference (TAC) is to promote research on systems that gather information on entities from large document collections, and use this extracted information to populate a structured knowledge base (KB) (Ji et al., 2011). This effort can be seen as a natural continuation of previous conferences and evaluations, such as the Message Understanding Conference (MUC) (Grishman and Sundheim, 1996) and the Automatic Content Extraction (ACE) evaluations[1]. Within this larger effort, the slot filling (SF) subtask must extract the values of specified attributes (or *slots*) for a given entity from large collections of natural language texts. Examples of slots include age, birthplace, and spouse for a person or founder, top members, and website for organizations. The temporal slot filling (TSF) subtask grounds this extracted values temporally by finding the start and end dates when these slot fillers were valid.

This is the fifth year a SF evaluation takes place, and the second for TSF, if the 2011 TSF pilot is counted (Ji et al., 2011). This year, 43 teams registered for at least one of these tasks. 18 teams submitted results for SF and five submitted results for TSF. This approximately 50% retention rate is in line with previous KBP evaluations, and highlights the difficulty of the task.

Both the SF and TSF evaluations followed definitions close to the ones from previous years. However, several important changes were implemented this year:

1. This year, systems had to provide provenance information for the entity and the filler and justification texts for the relation that are, at the same time, concise and informative. For example, justification is limited to at most two sentences, unlike previous years when entire documents were provided.

2. To lower the barrier of entry for TSF, this year's input for the TSF task included *both* the entity and the slot filler to be analyzed, similar to the diagnostic task in the 2011 pilot.

3. The document collections were extended

[1] http://www.itl.nist.gov/iad/mig/tests/ace/

with data from discussion forums, in the hope that this promotes research on information extraction from less formal texts.

We detail these distinctions in Section 2, which defines the two tasks together with their evaluation metrics. We overview the participants in these two tracks in Section 3, and analyze the systems and results in both tracks in Section 4 and 5, respectively. We conclude with a discussion of remaining challenges in Section 6.

2 Task Definitions

The overall goal of KBP is to automatically identify entities in natural language texts written in multiple languages; disambiguate them by linking them to entries in an existing KB; discover attributes about these entities (together with temporal validity spans); and, finally, expand the KB with any novel attributes. We refer the reader interested in such an overall KBP architecture to the description in (Ji et al., 2011). In this section, we focus mainly on the SF and TSF components.

2.1 Slot Filling Definition

The goal of the SF is to collect information on certain attributes (or *slots*) of entities, which may be either persons or organizations. Table 1 lists the slots for this year's SF evaluation.

2.1.1 Task Changes since 2012

This task is close to last year's task, with a few differences, discussed below.

Annotation guidelines

The slot annotation guidelines are close to last year's, with a few significant changes:

1. The definition of the per:title slot changed significantly. This year, titles that report positions at different organizations are reported separately. For example, Mitt Romney has held three different CEO positions: CEO at Bain Capital (1984–2002), CEO at Bain & Company (1991–92), and CEO at the 2002 Winter Olympics Organizing Committee (1999–2002). These positions must be reported as distinct titles by the systems.

2. The per:employee_of and per:member_of slots were merged into a single slot, per:employee_or_member_of due to their similarity.

3. This year, entities mentioned in document meta data can be used as input for the slot fillings tasks or fillers to be extracted by systems. For example, systems should consider as slot filler candidates the post authors, which are recorded in the meta data of discussion forum documents.

Please see the task definition document (Surdeanu, 2013), or the slot description and assessment documents for more details (Ellis, 2013b; Ellis, 2013a).

Query format

Similar to previous years, each query in the SF task consists of the name of the entity, its type (person or organization), a document (from the corpus) in which the name appears (to disambiguate the query in case there are multiple entities with the same name), its node ID (if the entity appears in the knowledge base), and the attributes which need not be filled. Additionally, to facilitate the disambiguation of the entity name, this year's queries include the start and end offsets of the name as it appears in the document. An example query is:

```
<query id="SF_002">
    <name>PhillyInquirer</name>
    <docid>eng-NG-31-141808-9966244</docid>
    <beg>757</beg>
    <end>770</end>
    <enttype>ORG</enttype>
    <nodeid>E0312533</nodeid>
    <ignore>
        org:city_of_headquarters
        org:country_of_headquarters
    </ignore>
</query>
```

Provenance of entity and filler

New this year is the fact that systems must provide provenance information for both entity and filler. Provenance is to be reported as start/end character offsets for the span of text which yielded the entity or filler. To account for the fact that systems may use coreference resolution and date normalization to extract or match the slot filler and entity, the provenance output must contain at least one mention and may contain the offsets of up to two relevant mentions, i.e., up to two pairs of start/end offsets.

For example, for a filler date that is normalized from the document date and the string "yesterday" should provide the offsets for both "yesterday" and the document date in its provenance. A

Person Slots			Organization Slots		
Name	**Type**	**List?**	**Name**	**Type**	**List?**
per:alternate_names	Name	Yes	org:alternate_names	Name	Yes
per:date_of_birth	Value		org:political_religious_affiliation	Name	Yes
per:age	Value		org:top_members_employees	Name	Yes
per:country_of_birth	Name		org:number_of_employees_members	Value	
per:stateorprovince_of_birth	Name		org:members	Name	Yes
per:city_of_birth	Name		org:member_of	Name	Yes
per:origin	Name	Yes	org:subsidiaries	Name	Yes
per:date_of_death	Value		org:parents	Name	Yes
per:country_of_death	Name		org:founded_by	Name	Yes
per:stateorprovince_of_death	Name		org:date_founded	Value	
per:city_of_death	Name		org:date_dissolved	Value	
per:cause_of_death	String		org:country_of_headquarters	Name	
per:countries_of_residence	Name	Yes	org:stateorprovince_of_headquarters	Name	
per:statesorprovinces_of_residence	Name	Yes	org:city_of_headquarters	Name	
per:cities_of_residence	Name	Yes	org:shareholders	Name	Yes
per:schools_attended	Name	Yes	org:website	String	
per:title	String	Yes			
per:employee_or_member_of	Name	Yes			
per:religion	String	Yes			
per:spouse	Name	Yes			
per:children	Name	Yes			
per:parents	Name	Yes			
per:siblings	Name	Yes			
per:other_family	Name	Yes			
per:charges	String	Yes			

Table 1: List of slots for TAC KBP 2013 slot filling. The slot in bold is new this year. The slot types can be: Name, i.e., named entities such as person, organizations, or locations; Value, i.e., numeric entities such as dates or other numbers; and String, which do not fall in any of the previous two categories. The list column indicates if the slot accepts multiple values for a given entity.

more complicated example involves coreference resolution for both slot filler and entity. For example, consider the query per:spouse of "Michelle Obama" and the text:

> *Michelle Obama started her career as a corporate lawyer specializing in marketing and intellectual property. Michelle met Barack Obama when she was employed as a corporate attorney with the law firm Sidley Austin. She married him in 1992.*

If a system extracts the filler "him" from the last sentence, and normalizes it to "Barack Obama" using coreference resolution, it must report offsets for both these strings in the filler provenance.

Justification of relation

Unlike previous SF evaluations, where document ids were required as justification, this year's task requires the justification to be a minimal number of clauses or sentences that provides justification for the extraction. The justification must contain at least one clause and at most two sentences. If two sentences are reported, they may be discontiguous. For example, in the above text, the last sentence is a valid justification for a system that performs extraction from individual sentences (but the provenances for entity and filler must include the necessary information to disambiguate them). As a more extreme example, a system that does not use coreference resolution but is capable of performing cross-sentence extractions may report the last two sentences as a valid justification.

One exception from this requirement is the per:alternate_names slot. This slot needs separate treatment because systems may extract it without any contextual information (other than occurrence in the same document). While textual patterns may sometimes provide useful context for this slot (e.g., "Dr. Jekyll, *also known as* Mr. Hyde"), it is possible to extract instances of this slot without such information. For example, a system may decide that "IBM" is an alternate name for "International Business Machines" solely based on the fact that the former is an acronym for the latter and they appear in the same document. To allow for these situations, we will accept empty justifications for this slot.

Document collections

We have extended the document collections for the English SF task with data from discussion forums, in the hope that it will promote research on information extraction from less formal texts. This year's source document collection contains: (a) one million documents from Gigaword; (b) one million web documents (similar to last year), and (c) approximately 100,000 documents from web discussion fora. To simplify training, this collection is released as a single LDC corpus, entitled "TAC 2013 KBP Source Corpus", with Catalog ID LDC2013E45.

2.1.2 Scoring Metric

The scoring process is similar with the previous year, with a small extension to handle the new provenance and justification texts. For completeness, we summarize it below. The main difficulty with scoring SF systems is that, just as in information retrieval (IR) evaluations, it is not feasible to prepare a comprehensive slot-filling answer key in advance. Because of the difficulty of finding information in such a large corpus, any manually-prepared key is likely to be quite incomplete. For this task, we approximate this comprehensive strategy by pooling the responses from all the systems and have human assessors judge the responses. To increase the chance of including answers which may be particularly difficult for a computer to find, LDC also prepares a manual key which was included in the pooled responses.

The slot filler in each non-Nil response is assessed as Correct, ineXact, Redundant, or Wrong, as follows:

1. A response that contains more than two sentences in the justification will be assessed as Wrong.

2. Otherwise, if the text spans defined by the provenance and justification offsets in (+/- a few sentences on either side of each span) do not contain sufficient information to justify that the slot filler is correct, then the slot filler will also be assessed as Wrong.

3. Otherwise, if the text spans justify the slot filler but the slot filler either includes only part of the correct answer or includes the correct answer plus extraneous material, the slot filler will be assessed as ineXact. No credit is given for ineXact slot fillers.

4. Otherwise, if the text spans justify the slot filler and the slot filler string in Column 5 is exact, the slot filler will be judged as Correct (if it is not in the reference KB) or Redundant (if it is in the reference KB). Note that, unlike previous years, this year's task removes the non-redundancy requirement with the KB for filler. That is, slot fillers that are already filled in the reference database must be reported as well. We hoped that this simplifies system development, as developers do not have to implement a redundancy component.

Two types of redundant slot fillers are flagged for list-valued slots. First, two or more system responses for the same query entity and slot may have equivalent slot fillers; in this case, the system is given credit for only one response, and is penalized for all additional equivalent slot fillers. (This is implemented by assigning each correct response to an equivalence class, and giving credit for only one member of each class.) Second, a system response will be assessed as Redundant with the reference knowledge base if it is equivalent to a slot filler in the reference knowledge base; in KBP 2013, these Redundant responses are counted as Correct, but NIST will also report an additional score in which such Redundant responses are neither rewarded nor penalized (i.e., they do not contribute to the total counts of Correct, System, and Reference below).

Given these judgments, we can count:

- Correct = total number of correct equivalence classes in system responses;

- System = total number of non-NIL system responses; and

- Reference = number of single-valued slots with a correct non-NIL response + number of equivalence classes for all list-valued slots.

The official evaluation scoring metrics are:

- Recall (R) = Correct / Reference

- Precision (P) = Correct / System

- $F1 = \frac{2PR}{P+R}$

2.2 Temporal Slot Filling Definition

This task is based on the TSF pilot at KBP 2011 (Ji et al., 2011). Its goal is to add temporal validity information to selected slots in the regular SF output. The task uses the following seven slots:

- per:spouse
- per:title
- per:employee_or_member_of
- per:cities_of_residence
- per:statesorprovinces_of_residence
- per:countries_of_residence
- org:top_employees/members

2.2.1 Changes since 2011

The task is close to the diagnostic subtask at the 2011 pilot, with two differences, discussed below.

Input

To allow participants to focus on the temporal aspect of the task, the TSF queries will include both the entity and the filler (unlike in the 2011 pilot when only the entity was given). The query format is very close to the *output* of the regular SF task. That is, both entity and filler are given, together with their provenance and justification for the corresponding relation. One example for a TSF query (ignoring the offsets in Columns 6 through 8) is:

```
Column 1: TEMP70711
Column 2: per:spouse
Column 3: Barack Obama
Column 4: AFP_ENG_20081208.0592.LDC2009T13
Column 5: Michelle Obama
Column 6: XXX-YYY
Column 7: ZZZ-WWW
Column 8: SSS-TTT
Column 9: 1.0
Column 10: E0566375
Column 11: E0082980
```

where Column 3 contains the entity, Column 5 the filler, Columns 6 and 7 their provenances, and Column 8 the justification for the relation in Column 2. Please see the task definition document (Surdeanu, 2013) for a full description of the query format.

Provenance of slot fillers

Similar to the regular slot filling task, the TSF output should include the offsets for at least one mention, and up to two mentions used for the extraction and normalization of temporal information. For example, if a system extracts the relative date "Wednesday" and normalizes it to "2008-12-31" using the document date from the document below:

```
<DOC>
<DOCID> AFP_ENG_20081231.0121.LDC2009T13 </DOCID>
<DOCTYPE SOURCE="newswire"> NEWS STORY </DOCTYPE>
<DATETIME> 2008-12-31 </DATETIME>
<BODY>
<HEADLINE>
Thousands protest in Brussels against
Israeli action in Gaza
</HEADLINE>
<TEXT>
<P>
Thousands took the streets in Brussels on Wednesday
calling for an end to Israeli bombing of the
Palestinian Gaza Strip ...
</DOC>
```

the system should report the offsets for both "Wednesday" and "2008-12-31" (from the <DATETIME> block) in the provenance.

2.2.2 Scoring Metric

This year's TSF task uses the same representation of temporal information as the 2011 pilot (Ji et al., 2011). For each relation provided in the input, TSF systems must produce a 4-tuple of dates: [T1 T2 T3 T4], indicating that the relation is true for a period beginning at some time between T1 and T2 and ending at some time between T3 and T4. A hyphen in one of the positions implies a lack of a constraint. Thus [– 20110101 20110101 –] implies that the relation was true starting on or before January 1, 2011 and ending on or after January 1, 2011. As discussed in Ji et al. (2011), there are situations (e.g., recurring events) that cannot be covered by this representation, but the most common situations for the relations covered in this task are addressed by this 4-tuple representation. For this reason, and in order to be able to reuse annotations generated in the 2011 pilot, we adopted the same representation this year. The training data this year included: (a) the training data used during the 2011 pilot, and (b) outputs of the systems that participated in the 2011 pilot, which were re-annotated by LDC for correctness.

The scoring metric this year is a simplified version of the 2011 scorer. Similar to the previous evaluation, we define a metric $Q(S)$ that compares a system's output $S =< t_1, t_2, t_3, t_4 >$ against a gold standard tuple $S_g =< g_1, g_2, g_3, g_4 >$, based on the absolute distances between t_i and g_i:

$$Q(S) = \frac{1}{4} \sum_i \frac{1}{1 + |t_i - g_i|}$$

The absence of a constraint in t_1 or t_3 is treated as a value of -∞; the absence of a constraint in t_2

or t_4 is treated as a value of $+\infty$. The unit of each tuple element is counted based on years.

The overall system score is simply an average of the $Q(S)$ metrics for each relation in the system output:

$$Accuracy = \frac{\sum_{S^i \in S} Q(S^i)}{N}$$

where S is the set of N system output tuples: $\{S^1, S^2, ..., S^N\}$, with one tuple for each query in the input. Note that the 2011 precision and recall scores are no longer necessary, because this year the queries include the correct fillers, whereas in the 2011 pilot the systems were responsible for extracting the fillers as well.

3 Participants Overview

Table 2 summarizes the participants that submitted at least one run in at least one of these two tasks. A larger number of teams (43) registered for at least one the tasks, but only 20 teams submitted results. Table 3 compares the number of participants and submissions for the two tasks with previous years. The last table shows a continuous increase in participation (both teams and runs) for SF, with a small dip in 2012. As discussed in the next section, this increase in participation is correlated with an increase in technology maturity, which is highly encouraging. For TSF, we also a see small increase in participation (five teams vs. four) and a considerable increase in number of runs submitted (from seven to 16).

4 English Slot Filling

4.1 Overall Results

Table 4 lists the results of the best run for each team that participated in SF. It is important to note that these scores are not directly comparable with last year's scores for two main reasons. First, the official score this year considers fillers that are redundant with the KB, which were ignored in previous years. However, as the diagnostic scores (which ignore these redundant fillers) indicate, this difference does not drastically affect performance. Generally, diagnostic scores are approximately 1 percentage point lower than the corresponding official scores. Our conjecture is that fillers that exist in the KB are the "easier" ones, e.g., with higher redundancy in the dataset, which makes it slightly easier for systems to extract them. Second and more importantly, this year's task definition requires that that the relation justification be the exact text (defined as one or two sentences) that supports the extracted filler. This is stricter than previous years, when the justification consisted simply of the id of the document that supported the extracted relation. Because of this, and of the fact that this year's evaluation dataset is more complex (see §4.3), one would expect this year's scores to be lower than the scores of previous SF evaluations. This is highlighted in the scores of the output manually generated by the LDC annotators: as seen in the table, this year they obtained an F1 score of 68.5, whereas last year they obtained a much higher score of 81.4 F1.

Considering these observations, it is clear that the results this year show increased performance. For example, last year's median score was 9.9 F1, whereas this year it is 15.7 F1. Last year only two systems obtained an F1 score over 30 points; this year the top six systems did. This strongly suggests that information extraction (IE) technology has improved, but it takes considerable time to reach this maturity: six out of the seven top systems have previously participated in at least one SF evaluation, and the majority participated in several. While this performance boost is very positive, it is important to put it in perspective: the top system this year is at approximately 54% of human performance (i.e., of the LDC annotators), and the median score is at only 23% of human performance. This is much lower than other NLP tasks, such as part-of-speech tagging or named entity recognition, where machines approach human performance.

With respect to technology, several observations can be made:

- Similar to previous years, the most successful approaches combine distant supervision (DS) with rules. This can be implemented in at least two different ways: (a) running these approaches as different systems and combining their output (NYU, Stanford, BIT), or (b) using rules to generate additional training data for DS (lsv). To the best of our knowledge, the latter approach is novel for KBP.

- For the first time, we see more complex DS models that have built-in noise reduction (Surdeanu et al., 2012) participate

Team Id	Organization(s)	SF?	TSF?
ARPANI	Bhilai Institute of Technology	√	
CMUML	Carnegie Mellon University	√	√
PRIS2013	Beijing University of Posts and Telecommunications	√	
TALP_UPC	TALP Research Center of Technical University of Catalonia (UPC)	√	
UWashington	Department of Computer Science and Engineering, University of Washington	√	
utaustin	University of Texas at Austin – AI Lab	√	
SINDI	Korea Institute of Science and Technology Information	√	
CohenCMU	Carnegie Mellon University	√	
UMass_IESL	University of Massachusetts Amherst, Information Extraction and Synthesis Lab	√	
BIT	Beijing Institute of Technology	√	
SAFT_KRes	University of Southern California Information Sciences Institute	√	
UNED	Universidad Nacional de Educación a Distancia	√	√
IIRG	University College Dublin	√	
NYU	New York University	√	
Stanford	Stanford University	√	
lsv	Saarland University	√	
Compreno	ABBYY	√	√
RPI-BLENDER	Rensselaer Polytechnic Institute	√	√
MS_MLI	Microsoft Research		√

Table 2: Overview of the SF and TSF participants at KBP 2013.

	English Slot Filling		Temporal Slot Filling	
	Teams	Submissions	Teams	Submissions
2009	8	16	–	–
2010	15	31	–	–
2011	14	31	4	7
2012	11	27	–	–
2013	18	53	5	16

Table 3: Number of participants and submissions in the past five years of KBP. For the 2011 TSF task we included the statistics of the diagnostic subtask, which is closer to this year's TSF.

in the evaluation, and perform well (Stanford). Previously, reducing the noise of data generated through DS was handled as a separate process, preceding model training (Min et al., 2012).

- For the first time, one group built their SF system around open-domain information extraction (OpenIE) technology (UWashington). They first extracted tuples in the format (Arg1, Rel, Arg2) from the KBP corpus using Open IE 4.0 (Mausam et al., 2012). Then they used a set of manually written rules to map these tuples to KBP-specific relations. This approach scored higher than the median and had the highest precision of all submissions, which is highly encouraging for a first-time participant.

- Another successful approach focused on the bootstrapping of patterns based on dependency tree paths, using tuples of entities and fillers from the KB as seeds (PRIS2013).

- Other notable approaches used unsupervised learning. For example, TALP_UPC's approach relies on an unsupervised clustering of patterns using the ensemble weak minority clustering algorithm (Gonzalez and Turmo, 2012). UMass_IESL's universal schema model combines observed and unlabeled data by performing a joint optimization over the train and test data together to factorize a matrix consisting of observed relations between entities (Riedel et al., 2012). Another exciting direction is utaustin's approach: they augment relations that are explicitly stated in the text by the system of Ji and Grishman (2011b) with ones that are inferred from the stated relations using probabilistic rules that encode commonsense world knowledge. These probabilistic first-order logic rules were learned using

	Diagnostic Scores			Official Scores		
	Recall	**Precision**	**F1**	**Recall**	**Precision**	**F1**
lsv	**32.93**	38.50	35.50	**33.17**	42.53	**37.28**
ARPANI*	29.10	47.83	**36.18**	27.45	50.38	35.54
RPI-BLENDER	30.62	38.19	33.98	29.02	40.73	33.89
PRIS2013	27.82	35.33	31.13	27.59	38.87	32.27
BIT	22.06	57.86	31.94	21.73	61.35	32.09
Stanford	28.46	32.30	30.26	28.41	35.86	31.70
NYU	17.35	50.70	25.85	16.76	53.83	25.56
UWashington	10.31	**59.72**	17.59	10.29	**63.45**	17.70
CMUML	10.63	28.79	15.53	10.69	32.30	16.07
SAFT_KRes	13.43	12.43	12.91	14.99	15.67	15.32
UMass_IESL	18.47	9.43	12.48	18.46	10.88	13.69
utaustin	7.91	21.85	11.62	8.11	25.16	12.26
UNED	9.11	15.08	11.36	9.33	17.59	12.19
Compreno	13.19	8.69	10.48	12.74	9.74	11.04
TALP_UPC	9.67	6.54	7.81	9.81	7.69	8.62
IIRG	3.20	7.38	4.46	2.86	7.72	4.17
SINDI	2.80	7.26	4.04	2.59	7.84	3.89
CohenCMU	3.68	1.69	2.32	3.68	1.98	2.57
LDC	58.35	83.81	68.80	57.08	85.60	68.49

Table 4: Overall results for SF, for the 100 entities in the evaluation dataset. The diagnostic score ignores fillers that are redundant with the reference KB (similar to previous years). The official score considers these redundant fillers during scoring. If multiple runs were submitted, we report the best run for each group. Results are listed in descending order of the official F1 score. The system marked with asterisk submitted their output after the deadline. The LDC score corresponds to the output created by the LDC experts.

Bayesian Logic Programs (BLP) (Raghavan et al., 2012). Unfortunately, all these systems performed below the median, but we suspect that most of these groups suffered a penalty from being first time participants.

4.2 Results without Justification

Table 5 lists system results when we relax the constraints on the justification. The left block of the table includes results when the scorer has the parameter `ignoreoffsets` set to true, which means that the justification is considered correct when the reported document id is correct (i.e., all offsets are ignored). The right block in the table shows results when the scorer has the parameter `anydoc` set to true, in which case the entire justification is ignored and fillers are considered correct if they match a gold filler. Note that these lenient scoring strategies have an important side effect: they collapse per:title fillers with the same value but applied to different organizations (e.g., "**CEO** of Apple" is different than "**CEO** of Next") because, without document ids and in-document offsets, we can no longer differentiate between them. Empirically, we observed that this collapsing of per:title fillers impacts mostly the `anydoc` configuration. For this reason, these lenient scores are not immediately comparable with the official scores in Table 4.

Despite the above limitation, several observations can be made based on the results in Table 5:

- One system (IIRG) had a significant bug in offset generation, which led to a considerable penalty in their official score. With the lenient scorer, this system's score increases by 16.80 F1 points with the `anydoc` configuration, and by 10.86 F1 points with the `ignoreoffsets` configuration.

- Ignoring the above system, the results suggest that the additional requirement imposed this year to provide in-document offsets for provenance and justification does

	Official Score with `ignoreoffsets`			Official Score with `anydoc`			
	Recall	**Precision**	**F1**	**Recall**	**Precision**	**F1**	**F1 Increase**
lsv	**33.56**	42.97	**37.69**	**35.84**	45.67	**40.17**	+2.89
RPI-BLENDER	29.13	40.82	34.00	31.87	44.46	37.13	+3.24
ARPANI*	27.49	50.36	35.57	28.72	52.38	37.10	+1.56
Stanford	29.20	36.80	32.56	32.49	40.76	36.16	+4.46
PRIS2013	28.03	39.44	32.78	29.34	41.07	34.23	+1.86
BIT	21.90	61.73	32.33	22.55	63.27	33.25	+1.16
NYU	16.98	54.49	25.90	18.16	57.99	27.66	+2.10
IIRG	10.50	28.31	15.32	14.39	38.60	20.97	**+16.80**
UWashington	10.44	**64.29**	17.96	11.38	**69.75**	19.56	+1.86
CMUML	10.71	32.30	16.09	11.72	35.19	17.58	+1.51
SAFT_KRes	15.55	16.24	15.89	17.20	17.88	17.53	+2.21
utaustin	8.46	26.22	12.79	10.76	33.19	16.25	+3.99
Compreno	13.48	10.26	11.64	17.82	13.54	15.39	+4.35
UNED	9.69	18.23	12.65	11.65	21.82	15.19	+3.00
UMass_IESL	18.49	10.88	13.70	20.49	12.01	15.14	+1.45
TALP_UPC	10.16	7.96	8.93	13.02	10.15	11.41	+2.79
SINDI	2.66	8.04	4.00	3.43	10.31	5.14	+1.25
CohenCMU	3.89	2.09	2.72	5.55	2.97	3.87	+1.30
LDC	57.36	85.90	68.79	59.01	87.95	70.63	+2.14

Table 5: Results for SF ignoring justification. In the `ignoreoffsets` configuration justifications are considered correct if the correct document is reported (similar to past years' evaluations). In the `anydoc` configuration justifications are completely ignored, and fillers are marked as correct solely based on string matching with gold fillers. If multiple runs were submitted, we report the best run for each group. Results are listed in descending order of the F1 score with `anydoc`. The system marked with asterisk submitted their output after the deadline. The LDC score corresponds to the output created by the LDC experts.

not impact the overall score in a considerable way. For example, the official score for the top system this year (lsv) is 37.28 F1 points (see Table 4), and the corresponding score with `ignoreoffsets` is 37.69. Similar small differences (under 1 F1 point) between these scores are observed for most participating systems. This observation indicates that, as long as systems manage to retrieve a correct supporting document, they generally extract justifications and provenances that are considered correct by LDC evaluators.

- On the other hand, identifying a valid supporting document for the extracted relation remains a challenge for some systems. Note that the `anydoc` scores are further removed from the official scores because ignoring the document id causes more collapsing for the per:title slots than the `ignoreoffsets` option. For example, because of this, the LDC score, which indicates the performance of the human expert, is boosted by slightly more than two points. However, even when accounting for this discrepancy, it is clear that some systems were penalized for not reporting a correct supporting document. This is considerable for two types of systems: (a) systems that extracted fillers from documents outside of the KBP source corpus, such as Stanford's best run, whose score improves by more than 4 F1 points under the `anydoc` scorer configuration; and (b) systems that inferred relations not explicitly stated in text, such as utaustin, whose score improves by 4 F1 points.

	Entity Count	Value Count (Pct)
per:title	33	142 (10.8%)
org:top_members_employees	41	116 (8.8%)
org:alternate_names	45	82 (6.2%)
per:employee_or_member_of	28	72 (5.5%)
per:children	23	52 (3.9%)
per:cities_of_residence	30	51 (3.9%)
per:age	31	51 (3.9%)
per:date_of_death	36	48 (3.6%)
per:cause_of_death	33	47 (3.5%)
per:charges	13	45 (3.4%)
per:alternate_names	24	45 (3.4%)
per:countries_of_residence	25	36 (2.7%)
per:city_of_death	32	35 (2.6%)
org:country_of_headquarters	34	34 (2.6%)
org:website	32	32 (2.4%)
per:origin	28	32 (2.4%)
per:spouse	23	28 (2.1%)
per:statesorprovinces_of_residence	23	28 (2.1%)
per:schools_attended	16	27 (2.0%)
org:subsidiaries	13	25 (1.9%)
per:parents	18	25 (1.9%)
org:city_of_headquarters	23	24 (1.8%)
org:members	4	22 (1.6%)
org:founded_by	11	21 (1.6%)
org:stateorprovince_of_headquarters	20	20 (1.5%)
per:stateorprovince_of_death	18	18 (1.3%)
org:shareholders	12	17 (1.3%)
per:date_of_birth	13	16 (1.2%)
per:other_family	10	15 (1.1%)
org:parents	11	13 (0.9%)
org:date_founded	13	13 (0.9%)
per:city_of_birth	12	12 (0.9%)
org:number_of_employees_members	11	12 (0.9%)
per:siblings	11	12 (0.9%)
per:stateorprovince_of_birth	10	10 (0.7%)
per:country_of_death	10	10 (0.7%)
per:religion	7	9 (0.6%)
per:country_of_birth	5	5 (0.3%)
org:member_of	4	4 (0.3%)
org:political_religious_affiliation	1	1 (0.0%)

Table 6: Distribution of correct slots in the 2013 SF gold dataset. The entity count column indicates how many of the 100 entities in the evaluation dataset contain at least one non-NIL correct fill for this slot. The value count column indicates how many equivalence classes were found for this slot across all entities. The slots are listed in descending order of the value count.

4.3 Distribution of Slot Types

Table 6 shows the distribution of slots in this year's evaluation data. As the table indicates, the distribution of slots in this dataset is not as skewed as in the previous years. For example, at KBP 2011 seven slots accounted for slightly more than 60% of data. These slots were: per:title, org:top_members_employees, org:alternate_names, per:employee_of, per:member_of, per:alternate_names, and org:subsidiaries. Some of these tend to be very local (e.g., per_title, per:employee_of, org:top_members_employees), which means that systems could perform well by focusing on few slots with local extraction patterns. This year this is no longer the case. For example, to reach 60% coverage of the evaluation data, a system would have to model 13 slots, and these include more complex relations such as per:charges. This suggests that the evaluation dataset this year was more difficult that in the past.

5 Temporal Slot Filling

Table 4 lists the results for TSF, overall and for each individual slot. Note that, even though the evaluation dataset contained 273 queries, only 201 were actually scored: 5 queries were dropped because neither the systems' nor LDC's output contained correct slot fillers; and 67 queries were eliminated because the gold temporal annotations

	S1	S2	S3	S4	S5	S6	S7	All
Baseline	24.70	17.40	15.18	17.83	14.75	21.08	23.20	19.10
MS_MLI	**31.94**	**36.06**	**32.85**	**40.12**	**33.04**	**31.85**	**27.35**	**33.15**
RPI-BLENDER	31.19	13.07	14.93	26.71	29.04	17.24	34.68	23.42
UNED	26.20	6.88	8.16	15.24	14.47	14.41	19.34	14.79
CMUML	19.95	7.46	8.47	16.52	13.43	5.65	11.95	11.53
Compreno	0.0	2.42	8.56	0.0	13.50	7.91	0.0	5.14
LDC	69.87	60.22	58.26	72.27	81.10	54.07	91.18	68.84

Table 7: Results for TSF, for the 201 evaluation queries that were scored. If multiple runs were submitted, we report the best run for each group. Results are listed in descending order of the accuracy for all slots (the official score). We also include scores for the individual slots as follows: S1: org:top_members_employees, S2: per:cities_of_residence, S3: per:countries_of_residence, S4: per:employee_or_member_of, S5: per:spouse, S6: per:statesorprovinces_of_residence, S7: per:title. The Baseline is the DCT-WITHIN baseline of (Ji et al., 2011). The LDC score corresponds to the output created by the LDC experts.

produced a gold standard tuple that had an invalid temporal interval (a temporal interval is valid only if $T1 \leq T2$, $T3 \leq T4$ and $T1 \leq T4$). Also note that the official scores currently ignore the textual justification generated by the systems, i.e., we score TSF outputs solely based on the formula introduced in Section 2.2.2. Implementing a policy for evaluating temporal justifications is left as future work.

The results in the table are compared against the DCT-WITHIN baseline of (Ji et al., 2011). This baseline makes the simple assumption that the corresponding relation is valid at the document date. That is, it creates a "within" tuple as follows: $< -\infty$, doc date, doc date, $\infty >$. The table shows that only two out of the five systems outperform this baseline: MS_MLI and RPI-BLENDER. This is worse than the equivalent evaluation in 2011, when three out of four systems outperformed this baseline (Ji et al., 2011). We also compare the system outputs against an output generated by human experts (LDC). This comparison indicates that the top performer this year achieves approximately 48% of human performance, and the median systems is at 21% of human performance. These numbers are slightly lower than the corresponding numbers measured for SF (53% for top system, and 23% for median), which suggests that TSF is a more difficult task.

The results for individual slot types indicate that the slots that address locations of residence (per:cities_of_residence, per:countries_of_residence, per:statesorprovinces_of_residence) perform generally worse than average, whereas the slots that address employment (org:top_members_employees, per:employee_or_member_of) tend to perform better than average. This suggests that, at least in this dataset, extracting temporal information for residence relations is harder than for employment relations. Our conjecture is that residence is more mobile than employment (e.g., a person can change residence but continue to work for the same employer), which increases the ambiguity of the corresponding relations.

With respect to technology, two trends are clear:

- Most groups used DS to assign temporal labels to tuples of <entity, filler, time> extracted from text. Several approaches (RPI-BLENDER, UNED) used training tuples from Freebase, whereas the top performing system (MS_MLI) used the Wikipedia infoboxes for this purpose. It is unclear if this made a difference in overall performance. With respect to temporal labels, most groups used at least `Start`, `End`, and `In` labels, with RPI-BLENDER adding an additional one (`Start-And-End`). Notably, one of the top systems (RPI-BLENDER) used an ensemble of classifiers combining flat features (surface text, dependency paths) with tree kernels (Ji et al., 2013).

- The top system (MS_MLI) used a language model to clean up the noise introduced by

DS before the actual temporal classification step. For example, this language model learned that n-grams such as "FILLER and ENTITY were married" are indicative of the per:spouse relation. These n-grams are then used as features in a boosted decision tree classifier that decides if the extracted <entity, filler, time> tuples belong to the relation under consideration or not. Considering that this noise removal step appears to be the most significant difference between the top and the second system in Table 7, these results suggest that noise removal is crucial for TSF as well.

6 Concluding Remarks

With respect to the SF task, this year's evaluation had some clear positive trends. First, this was the most popular SF evaluation to date, with 18 teams submitting results in 53 different runs. Second, this year's results show increased performance, on average. The median score this year was 15.7 F1, which is approximately 60% higher than the median score last year. This year, six teams obtained F1 scores over 30 points, whereas last year only two did. This is despite the fact that the task this year was more complex than past years' evaluations (with stricter scoring and more complex queries).

While this improvement is very positive, it is important to note that SF systems are still far from human performance on this task. The top system this year barely achieves 50% of human performance, and the median system is at only 23% of human performance. We are still far from solving the SF problem. Furthermore, retention of SF participants has not improved over the years. This year, slightly more than 50% of registered participants have dropped out of the evaluation. Similar drop-out rates have been observed in previous years. This strongly suggests that the SF task has a high barrier of entry, which can be detrimental for the success of the task. A possible way to lower participation effort is to offer more preprocessed data that allows participants to focus on IE models rather than on the engineering necessary to support them. For example, organizers could provide sentences (generated through distant supervision) that contain co-occurring entities and fillers (for training) or entities (for testing queries), which would allows participants to skip the information retrieval part of a SF system. System development effort can be further reduced by processing these sentences with named entity recognizers and syntactic parsers.

The considerable performance increase measured for SF this year is not replicated in the TSF results. This year, only two out of the five participating systems outperformed a simple baseline, whereas in 2011, when an equivalent evaluation was organized, three out of four did. While this is to a certain extent disappointing, it is important to note that TSF is a complex task that has started receiving attention from the research community only recently. We suspect it will take a few more years until TSF has the same research mass behind it as SF. We are already seeing that TSF can benefit considerably from more complex models inspired by the word in SF, e.g., this year's top TSF system included a noise reduction algorithm in their distant-supervision architecture.

Acknowledgments

We gratefully thank Hoa Dang, Joe Ellis, Heng Ji, and Ralph Grishman for babysitting the transition to the new organizational team for SF and TSF. We especially thank Hoa Dang for implementing the SF and TSF scorers, Heng Ji for providing us with older materials from the previous evaluations, and Heng Ji and Taylor Cassidy for running the TSF baseline.

References

Joe Ellis. 2013a. TAC KBP 2013 assessment. http://surdeanu.info/kbp2013/TAC_KBP_2013_Assessment_Guidelines_V1.3.pdf.

Joe Ellis. 2013b. TAC KBP 2013 slot descriptions. http://surdeanu.info/kbp2013/TAC_2013_KBP_Slot_Descriptions_1.0.pdf.

Edgar Gonzalez and Jordi Turmo. 2012. Unsupervised ensemble minority clustering. In *Research Report, Technical University of Catalonia*.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference – 6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*.

Heng Ji and Ralph Grishman. 2011b. Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*.

Heng Ji, Ralph Grishman, and Hoa Trang Dang. 2011. Overview of the TAC2011 knowledge base population track. In *Proceedings of the Text Analytics Conference (TAC2011)*.

Heng Ji, Taylor Cassidy, Qi Li, and Suzanne Tamang. 2013. Tackling representation, annotation and classification challenges for temporal knowledge base population. *Journal of Knowledge and Information Systems*.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland, and Oren Etzioni. 2012. Open language learning for information extraction. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL)*.

Bonan Min, Xiang Li, Ralph Grishman, and Ang Sun. 2012. New york university 2012 system for kbp slot filling. In *Proceedings of TAC-KBP*.

Sindhu Raghavan, Raymond J. Mooney, and Hyeonseo Ku. 2012. Learning to "read between the lines" using Bayesian Logic Programs. In *Proceedings of 50th Annual Meeting of the Association for Computational Linguistics*.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin Marlin. 2012. Relation extraction with matrix factorization and universal schemas. In *Proceedings of North American Chapter of the Association for Computational Linguistics*.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing and Natural Language Learning (EMNLP-CoNLL)*.

Mihai Surdeanu. 2013. Proposed task description for knowledge-base population at TAC 2013: English slot filling – regular and temporal. http://surdeanu.info/kbp2013/KBP2013_TaskDefinition_EnglishSlotFilling_1.1.pdf.