Overview of the TAC 2010 Knowledge Base Population Track

Heng Ji[1], Ralph Grishman[2], Hoa Trang Dang[3], Kira Griffitt[4], Joe Ellis[4]

[1]Computer Science Department, Queens College and the Graduate Center,
City University of New York, New York, NY, USA
hengji@cs.qc.cuny.edu

[2]Computer Science Department, New York University
New York, NY, USA
grishman@cs.nyu.edu

[3]National Institute of Standards and Technology
Gaithersburg, MD, USA
hoa.dang@nist.gov

[4]Linguistic Data Consortium, University of Pennsylvania
Philadelphia, PA, USA
{kiragrif, joellis}@ldc.upenn.edu

Abstract

In this paper we give an overview of the Knowledge Base Population (KBP) track at TAC 2010. The main goal of KBP is to promote research in discovering facts about entities and expanding a structured knowledge base with this information. A large source collection of newswire and web documents is provided for systems to discover information. Attributes (a.k.a. "slots") derived from Wikipedia infoboxes are used to create the reference knowledge base (KB). KBP2010 includes the following four tasks: (1) Regular Entity Linking, where names must be aligned to entities in the KB; (2) Optional Entity linking, without using Wikipedia texts; (3) Regular Slot Filling, which requires a system to automatically discover the attributes of specified entities from the source document collection and use them to expand the KB; (4) Surprise Slot Filling, which requires a system to return answers regarding new slot types within a short time period. KBP2010 has attracted many participants (over 45 teams registered for KBP 2010 (not including the RTE-KBP Validation Pilot task), among which 23 teams submitted results). In this paper we provide an overview of the task definition and annotation challenges associated with KBP2010. Then we summarize the evaluation results and discuss the lessons that we have learned based on detailed analysis.

1 Introduction

Traditional information extraction evaluations, such as the Message Understanding Conferences (MUC) and Automatic Content Extraction (ACE), assess the ability to extract information from individual documents in isolation. In practice, however, we may need to gather information about a person or organization that is scattered among the documents of a large collection. This requires the ability to identify the relevant documents and to integrate facts, possibly redundant, possibly complementary, possibly in conflict, coming from these documents. Furthermore, we may want to use the extracted information to *augment* an existing data base. This requires the ability to link individuals mentioned in a document, and information about these individuals, to entries in the data base.

The goal of the Knowledge Base Population (KBP) shared task is to address and evaluate these capabilities. This is done through two separate subtasks, Entity Linking and Slot Filling. For both tasks, the system is given a name and a document in which this name appears. For Entity Linking, it must decide whether this name corresponds to an entry in a data base and, if so, which one. For Slot Filling, the system must determine from a large corpus the values of specified attributes of the en-

tity, such as the age and birthplace of a person or the top employees of a corporation.

This is the second year that we are conducting a KBP evaluation. Although the essence of the evaluation is the same as KBP2009 (McNamee and Dang, 2009, McNamee et al., 2010) – with Entity Linking and Slot Filling sub-tasks – several changes were made based on what we learned from the initial evaluation:

- The corpus was enlarged to include a large number of weblogs; this introduced a wider variety of formats, less edited language, and a wider range of topics.
- Much more training data was available. In addition to the assessments from 2009, we commissioned the Linguistic Data Consortium to prepare manual responses to the 2009 queries and 50 additional training queries, and induced the participants to prepare manual responses to 50 further queries.
- Scoring for Slot Filling was changed to an F measure over filled slots. Last year's primary measure counted the accuracy over all slots, filled and empty. Because information to fill slots is relatively sparse in the corpus (most slots are correctly left blank), this favored (for the primary measure) systems which filled few slots. The baseline system, which left all slots blank, outperformed all participant systems.
- Some task simplifications were made: Slot Filling for locations (cities, states, countries) was dropped because we found that relatively little information about locations could be gleaned from the corpus. Entity Linking of slot values as an optional component of Slot Filling was dropped, having attracted little interest last year; the separate Entity Linking task was felt a sufficient test of this capability.
- A Surprise Slot Filling task was introduced – akin to the regular Slot Filling task, but allowing a maximum of 4 days for training and running.
- An optional entity-linking task was introduced, in which systems were not allowed to use the free text ('wiki_text') associated with each KB node; this variant more closely mimics the common setting where only structured data is available in the KB.

In addition to all these novel challenges of KBP, the traditional challenges of document-based information extraction remain. KBP Slot Filling is similar to ACE Relation Extraction[1], which has been extensively studied for the past 7 years. However, the amount of training data is much smaller, forcing sites to adjust their training strategies. Also, some of the constraints of ACE relation mention extraction – notably, that both arguments are present in the same sentence – are not present, requiring further innovations in extraction strategy.

2 Task Definition

This section will summarize the tasks conducted at KBP 2010. More details regarding data format can be found in the KBP 2010 task definition document[2]. The overall goal of KBP is to automatically identify salient and novel entities, link them to corresponding Knowledge Base (KB) entries (if the linkage exists), then discover attributes about the entities, and finally expand the KB with any new attributes. For example, the actor "James Parsons" became famous after he got an Emmy Award on August 29, 2010. A user may be interested in reading an accurate and concise profile (facts) about him. An Entity Linking system can link any document including "James Parsons" to the corresponding KBP entry, or determine that no corresponding entry exists (for the real application this means a new Wikipedia entry needs to be constructed about this person). This process involves both name disambiguation (e.g. the actor "James Parsons" should not be linked to the lawyer "James A. Parsons" or the judge "James B. Parsons") and name variant clustering ("James Parsons" = "Jim Parsons" for the actor). Furthermore, a slot filling system is required to discover the values of predefined attributes about "James Parsons". For example, if "University of Houston" is extracted as his "school_attended" attribute, and this fact does not exist in the KB yet, the system should add this information to expand the KB. For the evaluation an initial (or reference) KB derived from Wikipe-

[1] http://projects.ldc.upenn.edu/ace/
[2] http://nlp.cs.qc.cuny.edu/kbp/2010/

dia Infoboxes is provided to the systems, along with a large collection of source documents.

2.1 Regular and Optional Entity Linking

In the Entity Linking task, given a query that consists of a name string and a background document ID, the system is required to provide the ID of the KB entry to which the name refers; or NIL if there is no such KB entry. For example, one of the evaluation queries is

```
<query id="EL000304">
  <name>Barnhill</name>
  <docid>eng-NG-31-100578-11879229</docid>
</query>
```

For the regular entity linking task, the system may consult the text from the Wikipedia pages associated with the KB nodes. However, in a more realistic setting, when a salient and novel entity appears in news or web data, there may not be many Wikipedia texts to utilize. Therefore in KBP 2010, an optional entity linking task was conducted, in which the systems can only use the attributes in the KB; this corresponds to the task of updating a KB with no 'backing' text.

2.2 Regular Slot Filling

The goal of Slot Filling is to collect from the corpus information regarding certain attributes of an entity, which may be a person or some type of organization. Each query in the Slot Filling 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. Attributes are excluded if they are already filled in the reference data base and can only take on a single value. An example query is

```
<query id="SF114">
  <name>Masi Oka</name>
  <docid>eng-WL-11-174592-12943233</docid>
  <enttype>PER</enttype>
  <nodeid>E0300113</nodeid>
  <ignore>per:date_of_birth per:age
per:country_of_birth per:city_of_birth</ignore>
</query>
```

Along with each slot fill, the system must provide the ID of a document which supports the correctness of this fill. If the corpus does not provide any information for a given attribute, the system should generate a NIL response (and no document ID).

The sets of attributes are listed in Tables 1 and 2.

per:alternate_names	Name	List
per:date_of_birth	Value	Single
per:age	Value	Single
per:country_of_birth	Name	Single
per:stateorprovince_of_birth	Name	Single
per:city_of_birth	Name	Single
per:origin	Name	List
per:date_of_death	Value	Single
per:country_of_death	Name	Single
per:stateorprovince_of_death	Name	Single
per:city_of_death	Name	Single
per:cause_of_death	String	Single
per:countries_of_residence	Name	List
per:stateorprovinces_of_residence	Name	List
per:cities_of_residence	Name	List
per:schools_attended	Name	List
per:title	String	List
per:member_of	Name	List
per:employee_of	Name	List
per:religion	String	Single
per:spouse	Name	List
per:children	Name	List
per:parents	Name	List
per:siblings	Name	List
per:other_family	Name	List
per:charges	Name	List

Table 1. Person Attributes for Slot Filling

For each attribute we indicate the type of fill and whether the fill must be (at most) a single value or can be a list of values. Since the overall goal is to augment an existing KB, two types of redundancy in list-valued slots must be detected and avoided. First, two fills for the same entity and slot must refer to distinct individuals. Second, if the knowledge base already has one or more values for a slot, items in the system output must be distinct from those already in the knowledge base. In both cases, it is not sufficient that the strings be distinct; the fills must refer to distinct individuals. For example, if the knowledge base already has a slot fill

"William Jefferson Clinton", the system should not generate a fill "Bill Clinton" for the same slot.

org:alternate_names	Name	List
org:political/religious_affiliation	Name	List
org:top_members/employees	Name	List
org:number_of_employees/members	Value	Single
org:members	Name	List
org:member_of	Name	List
org:subsidiaries	Name	List
org:parents	Name	List
org:founded_by	Name	List
org:founded	Value	Single
org:dissolved	Value	Single
org:country_of_headquarters	Name	Single
org:stateorprovince_of_headquarters	Name	Single
org:city_of_headquarters	Name	Single
org:shareholders	Name	List
org:website	String	Single

Table 2. Organization Attributes for Slot Filling

2.3 Surprise Slot Filling

One longstanding concern is how easily and rapidly an information extraction system can be adapted to new types of relations and events. To assess this, we introduced for 2010 a 'surprise slot filling task', where participants were given four new slot types ("diseases", "awards-won" and "charity-supported" for persons, "products" for organizations) along with annotation guidelines and training data (correct fills) for 24 person entities and 8 organization entities. A total of 83 slot fills were provided for the 32 entities. Sites were given a maximum of 4 days to develop their systems and run them on the KBP corpus, and were encouraged to complete the task in a shorter time if possible.

Only a few sites ended up participating in the surprise task, possibly because it was scheduled immediately after the regular task, which many sites were still struggling to complete.

2.4 Changes Compared to KBP2009

The KBP2010 task is largely similar to 2009 Slot Filling, but several small changes were made. The answer linking task was dropped because the required techniques largely overlap with regular entity linking. Location slots in the 2009 task (place of birth, place of residence, place of death, place of headquarters) were each divided into three slots (city, state/province, country). Origin was changed from a single to a list value. And, as mentioned earlier, locations ("Geo-Political Entities", in ACE terminology) were dropped as a query type from the Slot Filling task because there was little information about such entities in the corpus.

For 2009 the entities for Slot Filling were selected from a larger set used to evaluate Entity Linking. In consequence, these entities included a larger number of ambiguous names and acronyms. For 2010 the entities were chosen independently of Entity Linking and were more representative of the names in the corpus. In particular, they were less ambiguous and so name disambiguation played a less critical role in this year's Slot Filling.

System developers in 2009 found that external sources of information could substantially improve performance by hypothesizing slot fills which would then be validated through a search of the corpus. Some sites used Web search in this way as a part of their evaluation system. To assess the impact of such external sources and have a more level playing field, we required each site to submit at least one run as a 'closed system' which did not access the Web at evaluation time. In addition, we encouraged sites which use a large off-line source of facts such as a copy of the Wikipedia text or Freebase to submit an additional run without this data – to see how well a system could perform using the provided text corpus as its sole large-scale source of facts. The detailed results and analysis about using external KB and Web access are presented in sections 8.4 and 12 below.

3 Participants Overview

Table 3 summarizes the participants for each task. Over 45 teams registered for KBP 2010 (not including the RTE-KBP Pilot task), among which 23 teams submitted results. Each team was allowed to submit up to 3 submissions for each task. Table 4 shows the number of participants and submissions compared to KBP2009.

Team Name	Organization	Regular Entity Linking	Optional Entity Linking	Regular Slot Filling	Surprise Slot Filling
ARPANI	Bhilai Institute Of Technology	√			
BUDAPEST-ACAD	Computer & Automation Research Institute, Hungarian Academy of Science	√	√	√	
BUPTPRIS	Beijing University of Posts and Telecommunications	√		√	
CMCRC	Capital Markets Cooperative Research Centre	√			
CORTEX	Cortex Intelligence			√[3]	√[4]
CUNY	City University of New York	√		√	√
HLTCOE	Johns Hopkins University Center of Excellence	√	√	√	
IBM	IBM T.J. Watson Research Center			√	
ICL	Peking University Institute of Computational Linguistics	√	√	√	
IIRG	University College Dublin			√	√
LCC	Language Computer Corp.	√		√	√
LSV	Saarland University			√	
NUSchime	National Univ. of Singapore	√			
NYU	New York University			√	
SIEL	Int'l Institute of Information Technology, Hyderabad	√	√	√	
SMU	School of Information Systems, Singapore Management University	√	√		
STANFORD	Stanford University			√	
STANFORD_UBC	Stanford University and University of the Basque Country	√			
TCAR	Text Content Analytics Research	√	√		
UBC	University of the Basque Country			√	√
UC3M	University Carlos III de Madrid	√	√		
USFD	University of Sheffield	√		√	
WebTLab	Web Technologies Laboratory, University Carlos III de Madrid	√			

Table 3. Overview of KBP2010 Participants

Tasks	2009		2010	
	#Teams	**#Submissions**	**#Teams**	**#Submissions**
Regular Entity Linking	13	35	16	46
Optional Entity Linking	-	-	7	20
Regular Slot Filling	8	16	15	31
Surprise Slot Filling	-	-	5	6

Table 4. Number of KBP Participants and Submissions

[3] This system used 2 million additional documents with extensive, partially-manual, annotation.
[4] This system performed semi-automatic slot filling, involving manual review.

Genre	#documents
Broadcast Conversation	17
Broadcast News	665
Conversational Telephone Speech	1
Newswire	1,286,609
Web Text	490,596

Table 5. # Documents in Source Collection

Corpus	Genre/Source	Size (entity mentions)		
		Person	Organization	GPE
Training	2009 Training	627	2710	567
	2010 Web data	500	500	500
Evaluation	Newswire	500	500	500
	Web data	250	250	250

Table 6. Entity Linking Corpora

Corpus	Task	Source	Size (entities)	
			Person	Organization
Training	Regular Task	2009 Evaluation	17	31
		2010 Participants	25	25
		2010 LDC	25	25
	Surprise Task	LDC	24	8
Evaluation	Regular Task	LDC	50	50
	Surprise Task	LDC	30	10

Table 7. Slot Filling Corpora

4 Data Annotation

4.1 Overview

A source collection of documents is provided, with the detailed statistics shown in Table 5.

The reference knowledge base includes hundreds of thousands of entities based on articles from an October 2008 dump of English Wikipedia which includes 818,741 nodes. Table 6 and Table 7 summarize the KBP2010 training and evaluation data provided for participants. For both tasks we try to achieve a balance between different genres, and between the queries with and without KB entry linkages.

For the entity linking system, a substantial amount of web training data was added this year in order to evaluate the impact of noisy genres on the task.

Slot Filling System developers had access to four sets of keys for use in system training and tuning:

1) Slot fills marked 'correct' in the adjudication of the 2009 evaluation data. These used a slightly different set of slots, so a small adjustment was required, primarily for location slots.

2) A key prepared by Linguistic Data Consortium (LDC) annotators for the queries of the 2009 evaluation.

3) A key prepared by LDC annotators for an additional 50 queries (25 person and 25 organization entities).

4) A key prepared manually by the participants for another 50 queries (25 person and 25 organization entities), with more details shown in the next subsection.

In 2) and 3) the manual search was limited to 2 hours per entity.

4.2 Data Selection

We ran an English name tagger (Grishman et al., 2005) on the entire source collection. For any name string *n*, we counted the number of documents containing *n*, which was then used to count the number of KB entries that match *n*.

For entity linking, we selected confusable queries which have no or many (7 or more) possible KB entry matches. In addition, we kept a balance between entity disambiguation and alias detection for the query selection. Selection preferences were given to entities which have an incorrect spelling (such as 'Bil Clinton' vs. 'Bill Clinton' or an alternative spelling, such as ('Jon' vs. 'John'), abbreviated forms (e.g. 'CDC') and ambiguous or more common names (e.g. 'John Smith'). 20 documents were selected from the source data for each name string. If more than 20 documents were found, the selected set should equally represent all unique entities.

For slot filling, we selected informative queries which were non-confusable. A candidate query is considered as informative if its name string is contained in 1-6 KB entries, and there are relatively many (at least 2-3) slot answers to be potentially filled from the source collection. The candidate query should also have reference documents existing in the source collection.

We excluded name strings which did not meet the above confusability requirements for the Entity Linking or Slot Filling tasks, were incorrectly tagged by the name tagger, nonsensical, or included objectionable content.

4.3 Slot Filling Annotation

The KB entries for all SF entities were reviewed to prevent redundant annotations. Those already filled slots were made visible to annotators but could not be altered. The KBP2010 slot filling annotation guideline was developed based on the KBP2009 assessment guideline by addressing all known issues from the 2009 assessment process. Besides the guideline, the task definition and slot type category descriptions with examples are also provided to the annotators.

LDC conducted in-house annotator training so that potential annotators could complete a practice kit with supervision. The training procedure includes two sessions: (1) In-person training session including explanation of the task, demonstration of annotation tool and distribution of annotation guidelines; (2) In-person supervised annotation session including completion of an in-person supervised practice annotation kit, and then assignment of production kits.

Each annotator used an annotation tool that can provide the following functions: (1) search function including searching over all source documents, editing search string and syntax (rank results by relevance, search for exact strings, etc.); (2) annotation function including directly selecting and annotating fillers from text and the ability to edit the selection (e.g. 'Christians' selected and edited to 'Christian' for PER:Religion). The annotators were presented with an entity and all available slots for that entity, and required to perform within a two-hour time limit for each entity.

4.4 Slot Filling Assessment

We refined the annotation guideline before the assessment by clarifying annotations of the following cases: (1) specific GPEs (Emirates, Capital Districts); (2) actors as employees, headquarters of Universities, headquarters of sports teams, implication of origin on residence; and (3). non-recognized states and areas of contention; alternate names used in connection with other slots.

The rigorous pre-testing process removed a number of annotators. Assessors were required to complete a full-size assessment test kit, with 12 slots filled with slot-filling answers for an actual entity. Assessors were required to assess every slot with 90% accuracy or greater on a standard test kit, across all slots. This process also caused clarifications to be made in the guidelines. For example, former/past relationships are acceptable fillers while future relationships are not acceptable filler. In addition, further clarifications were made to specific slots, including PER:Age, PER:Alternate Names, PER:Country of Birth, PER:Employee Of, PER:Origin, ORG:Alternate Names and

ORG:Number of Employees/Members. Those who passed went on to assess the validity of slot-filling answers from both humans and systems. After assessment was completed, systematic and spot-checking quality control was performed on 50% of the assessment data. These quality control passes revealed only incidental errors, which were corrected prior to release of assessment results. We will address the errors identified in assessment results in the KBP2011 slot-filling annotation and assessment guidelines.

4.5 Participant Annotation – A Community Effort

Slot filling annotation is a costly task because it requires the annotators to conduct an exhaustive search. In addition, LDC was only able to provide single annotation for each query, which led to limited recall, as we will discuss in the next section.

Therefore this year we organized a community effort to augment the slot filling training corpora. This work was distributed among the participants, with most queries annotated independently by two sites but some annotated by as many as six sites. 26 KBP teams volunteered to annotate 25 persons and 25 organizations together. Table 8 illustrates an example of the task distribution to avoid cross-site bias.

	Team 1	Team 2	Team 3	Team 4
Query1	√			√
Query2	√	√		
Query3	√	√		
Query4		√	√	
Query5			√	√
Query6			√	√

Table 8. Example of Task Distribution for Participant Slot Filling Annotation

4.6 Entity Linking Inter-Annotator Agreement

LDC conducted an Entity Linking Inter-Annotator Agreement Study on 200 selected queries from the evaluation set. The selection procedure was based on the following criteria:

- 100 queries each were selected from web text and non-web text documents;
- The proportions of person (PER), organization (ORG), and geo-political GPE entities were kept as close as possible to the evaluation query set (68 PERs, 66 GPEs and 66 ORGs)
- The proportion of KB to NIL entities was kept as close as possible to the evaluation query set

LDC recruited 3 annotators who had not previously worked on KBP-related annotation. One or more of the annotators used Web search when annotating 20 of the queries. In order to conduct a fair comparison between human and systems, we removed these queries. Figure 1 presents the results for the remaining 180 queries. 90.56% of the queries received the same annotations from all three annotators.

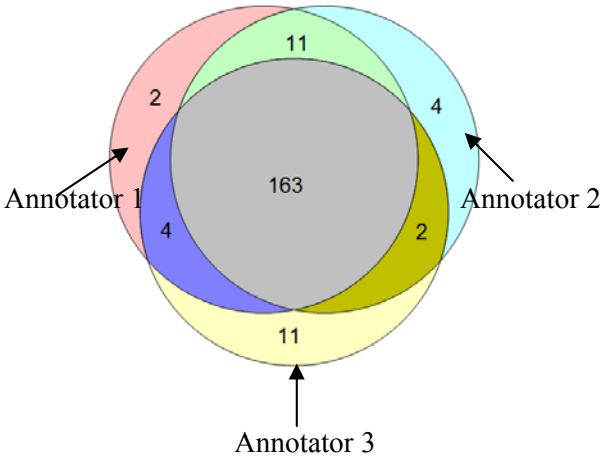

Figure 1. Inter-Annotator Agreement Number for Entity Linking

Table 9 shows the distribution of the disagreement cases. The agreement rate represents the percentage of queries on which all three annotators agreed. Geo-political names are relatively more difficult than persons and organizations.

Entity Type	#Total Queries	Agreement Rate	Genre	#Disagreed Queries
Person	59	91.53%	Newswire	4
			Web Text	1
Geo-political	64	87.5%	Newswire	3
			Web Text	5
Organization	57	92.98%	Newswire	3
			Web Text	1

Table 9. Distribution of Disagreement Cases in Entity Linking

For those 20 queries that relied on Web search, only 15 of them (75%) received the same annotations from all three annotators. This indicates that when human annotators have to rely on Web access to make the decision on a query, that query is probably highly ambiguous. Most of these cases are abbreviation names such as "NCIC" and "CAA".

4.7 Slot Filling Annotation Bottleneck

Figure 2 shows the number of correct answers for 3 persons ("*Dario Franchitti*", "*Anne Flaherty*" and "*Francois Mitterrand*") and 3 organizations ("*San Antonio Spurs*", "*Al-Arabiya*" and "*Viacom Inc.*") as we merge the results from more participant annotators.

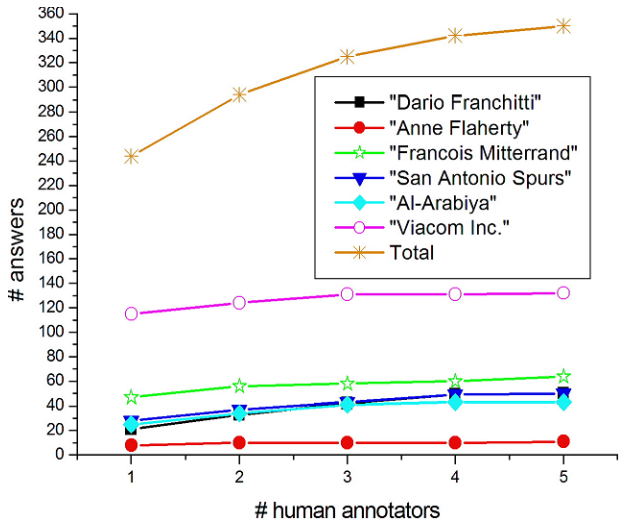

Figure 2. Human Annotation Combination

We can see that a single human annotator can only find 70% of the answers merged from five annotators. On the other hand, the annotation set tends to converge by the time the fifth annotator is added (the fifth annotator only found 8 new answers compared to the set of 342 answers merged previously). No obvious spurious answers were observed from these annotations.

During the 2010 evaluation data annotation/adjudication process, an initial answer key annotation was created by a manual search of the corpus (resulting in 797 instances), and then an independent adjudication pass was applied to assess these annotations together with pooled system responses. Table 10 shows the Precision, Recall and F-measure for the initial human annotation.

Performance	Precision	Recall	F-Measure
All slots	70.14	54.06	61.06
All slots except org:top-employee, per:member_of, per:title	71.63	57.6	63.86

Table 10. LDC Human Annotator Performance for the KBP2010 Evaluation Data (%)

The results show that only 70.14% of the 559 answers from the initial manual annotation were confirmed as correct in the subsequent adjudication. If we remove the relatively confusable slot types (org:top-employee, per:member-of and per:title), the performance was slightly better. For the remaining 238 answers, 32 were judged as inexact and 200 as wrong answers. This indicates that even for the manually discovered answers, a large portion are not exactly correct. We asked another annotator to examine 20 answers marked as incorrect; for 65% of them the third annotator agrees that the answers are indeed incorrect; for 15% the third annotator believes the answers should be correct

and the assessor overlooked the context sentences; while for the remaining 20% the third annotator cannot decide because of lack of specificity in the annotation guideline.

The human annotation errors include:

(1). The answers are not explicitly stated in the document, for example, based on "*Staff at the state-owned Paris Opera and Comedie Francaise joined in a strike last week over plans by President Nicolas Sarkozy's government to reform the special pension benefits of 500,000 public sector workers*." the annotator mistakenly extracted "*France*" as the answer for "org:country_of_headquarters" slot of the query "*Opera National de Paris*" based on world knowledge and reasoning between words "*Paris*" and "*Francaise*";

(2). The answers are explicitly stated in the document but they require a little world knowledge and reasoning. For example, from "*According to the National Republican Congressional Committee website: www.nrcc.org/issues/default.asp?ID=47*" the annotator extracted "www.nrcc.org" as the answer for "org:website" slot for the query "*National Republican Campaign Committee*", but it needs the knowledge of linking "*National Republican Campaign Committee*" and "*National Republican Congressional Committee*";

(3). Many wrong answers were produced because of the ambiguities and underspecification in the annotation guideline, in particular about slot types such as "org:top_members/employees", "org:members" and "per:title". Many of the slot filing answers that were judged incorrect came from these three difficult slot types;

(4). The KBP 2010 slot filling annotation guidelines were developed from the KBP2009 assessment guidelines. The assessors identified additional challenges and questions in the data during the 2010 assessment task, which resulted in revisions to the slot definitions in the assessment guidelines and a stricter evaluation of the Slot Filling annotations;

(5). A few mistakes were due to confusion about acceptable answer representation, for example, whether "70"/"30,000" is a good answer to extract from "about 70 employees"/"more than 30,000 members" for the "org:number_of_employees/members" slot.

5 Evaluation Metrics

In this section we will present the evaluation metrics used in KBP 2010.

5.1 Entity Linking Metric

For each query, we check whether the KB node ID (or NIL) returned by a system is correct or not. Then we compute the Micro-averaged Accuracy, computed across all queries. Because KBP2010 doesn't require a system to cluster NIL answers, the Macro-averaged Accuracy (computed across all KB entries) becomes less meaningful. So the official scores are based on the Micro-average Accuracy.

5.2 Slot Filling Metric

As is the case with IR (document retrieval) 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. Instead (as for IR) we pool 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 did prepare a manual key which was included in the pooled responses.

Each response is rated as correct, inexact, redundant, or wrong. A response is inexact if it either includes part of the correct answer or includes the correct answer plus extraneous material. No credit is given for inexact answers. Two types of redundant answers are flagged for list-valued slots. First, a system response may be equivalent to an answer in the reference knowledge base; this is considered incorrect. Second, two system responses for the same attribute may be equivalent; in the latter case, only the first of a set of equivalent answers is marked correct. (This is implemented by assigning each correct answer to an *equivalence class*, and only giving credit for one member of each class.)

Given these judgments, we can count

Correct = total number of non-NIL system output slots judged correct
System = total number of non-NIL system output slots
Reference = number of single-valued slots with a correct non-NIL response + number of equivalence classes for all list-valued slots
Recall = Correct / Reference
Precision = Correct / System
F-measure = (2 x Recall x Precision) / (Recall + Precision)

The F score is the primary metric for system evaluation.

6 Evaluation Results

This section will summarize the evaluation results, and following sections will provide detailed analysis and discussion. From each participant, we only select the best submission without Web access for comparison.

6.1 Entity Linking Results

The results of regular entity linking and optional entity linking are summarized in Figure 3, Figure 4 and Figure 5.

From Figure 3 we can see that systems generally performed the best on person entities, and the worst on geo-political entities (which is consistent with human performance).

Figure 4 shows the comparison between the averaged human annotators and the top 5 systems on the subset of 200 queries used for inter-annotator agreement study. We can see that Siel and CMCRC systems achieved higher performance than human annotators on person entities, and LCC achieved similar performance with human annotators.

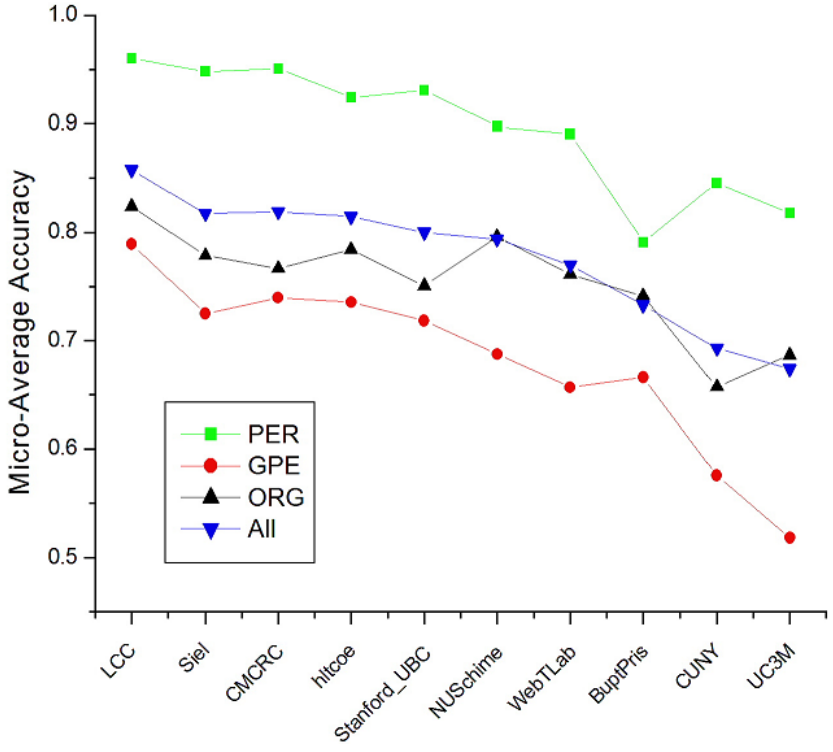

Figure 3. Top-10 Regular Entity Linking System Performance (All Queries)

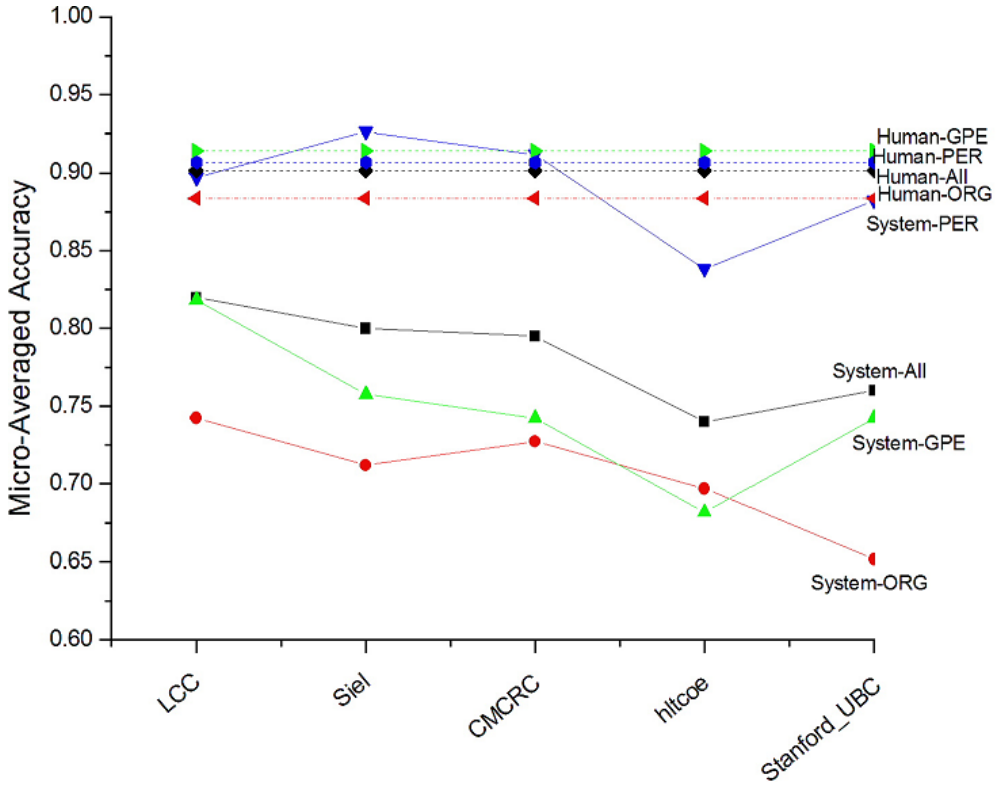

Figure 4. Comparison between System and Human Entity Linking Performance (Subset of 200 queries)

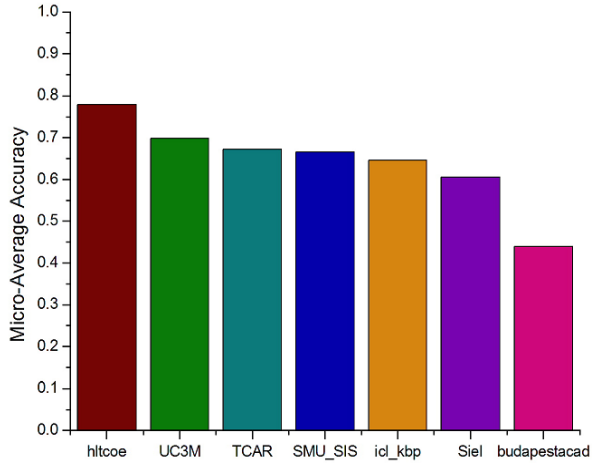

Figure 5. Overall Optional Entity Linking Performance

In addition, we have found that there is a low correlation between the micro-averaged accuracy for overall entities and for Non-NIL entities – 0.777 for the regular task and 0.723 for the optional task. This indicates that because the task doesn't require systems to cluster NIL entities, it actually provided a lot of flexibility for systems to tune parameters in order to optimize the F-measure of overall queries. But the optimized parameters cannot directly lead to the best performance on Non-NIL entities. Therefore, in KBP2011 we will add a task of clustering NIL entities so that the systems can be optimized and evaluated with clustering. Several teams such as UC3M (Pablo-Sanchez et al., 2010) and LCC (Lehmann et al., 2010) have already developed NIL classifiers as post-processing this year, so we hope that NIL clustering task will be a natural addition to entity linking.

Finally, among the 7 systems that participated in both the regular and optional tasks, 5 got worse results without using Wikipedia texts, while the other 2 systems actually got some slight improvement. The Wilcoxon Matched-Pairs Signed-Ranks Test across these systems showed that the improvement using Wikipedia texts is not statistically significant (only at 89% confidence level).

Figure 6 presents the regular entity linking progress for the teams that participated in both KBP2009 and KBP2010. Although the query sets are different, they followed similar selection crite-

ria, so we can clearly see that systems, especially the UC3M and TCAR teams, have made consistent improvement since 2009.

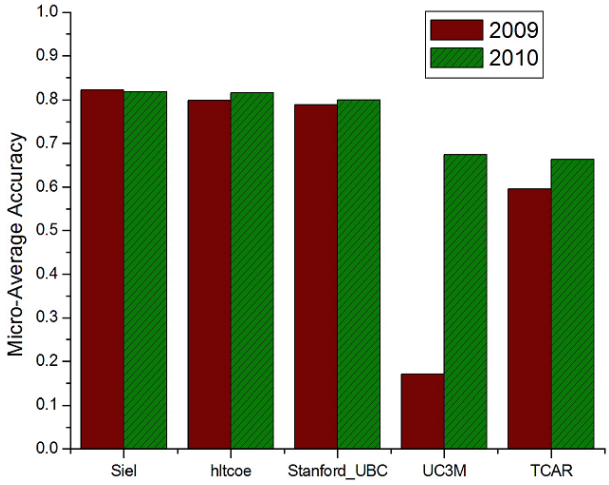

Figure 6. Regular Entity Linking Progress from 2009 to 2010

6.2 Regular Slot Filling Results

Figure 7 presents the performance of the top 10 regular slot filling systems.

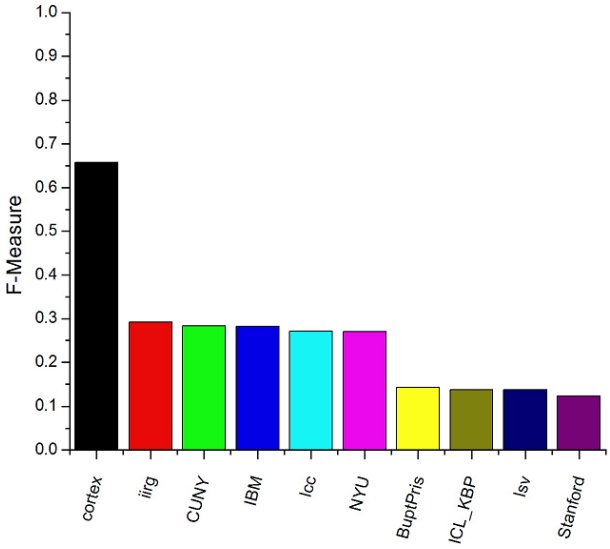

Figure 7. Top 10 Regular Slot Filling System Performance

It's noteworthy that the Cortex system, which achieved much higher performance than all other systems, used an additional 2 million documents, which were extensively annotated in a semi-automatic way. We didn't compare with KBP2009 results because the queries in 2009 were selected differently (using the same criteria as for entity linking) and the evaluation metric was quite different (as discussed above), and thus the results from the two years are not comparable.

6.3 Surprise Slot Filling Results

Four sites fielded automatic systems for filling the surprise slots. Given the small amount of training data and limited time, this was a difficult task and only one site, LCC, exceeded 10% F score on the evaluation metric. They did an evaluation after 11 hours of development and a second evaluation after 34 hours. Figure 8 presents their performance. We can see that as more time is spent on system development and adaptation, the precision was slightly enhanced while significant improvement was obtained on recall.

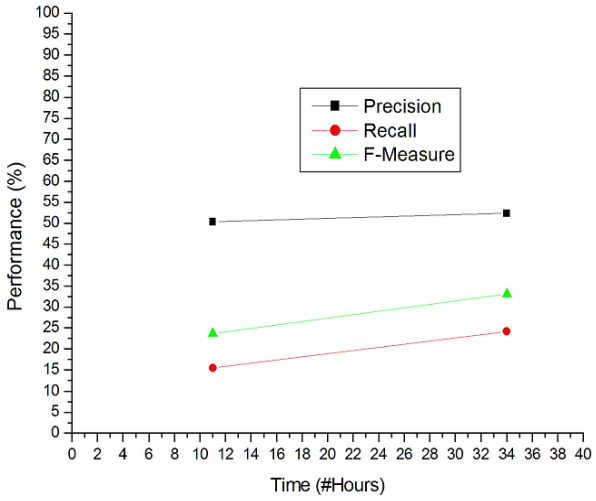

Figure 8. LCC Surprise Slot Filling System Performance

7 Discussion of Entity Linking

There are two principal challenges in entity linking: the same entity can be referred to by more than one name string and the same name string can refer to more than one entity. The research on these problems can be traced back to the Web People Search task (Artiles et al., 2007) and NIST Automatic Content Extraction (ACE2008), and the interest in these problems has rapidly grown among different NLP communities. A variety of approaches have been proposed to solve the task with considerable success; nevertheless, there are many

aspects of the task that remain unclear. What is the impact of the features used so far? What kind of problems are represented in the current KBP entity linking testbeds? In which way have the current testbeds and evaluation methodology affected our perception of the task difficulty? Have we reached a performance ceiling with current state of the art techniques? In this section we aim to answer some of these questions based on our analysis of evaluation results.

7.1 Unsupervised/Weakly-Supervised vs. Supervised Learning

The approaches exploited in the KBP2010 entity linking systems can be generally categorized into two types: (1) Unsupervised or weakly-supervised learning, in which annotated data is minimally used to tune thresholds and parameters; the similarity measure is largely based on the unlabeled contexts; (2) Supervised learning, in which a pair of entity and KB node is modeled as an instance for classification. Such a classifier can be learned from the annotated training data based on many various features. Methods in type (2) involve a lot of feature engineering and are therefore much more difficult to duplicate. So the first question we will investigate is how much higher performance can be achieved by using supervised learning? Among the 16 regular entity linking systems, CMCRC (Radford et al., 2010), hltcoe, Stanford_UBC (Chang et al., 2010), NUSchime (Zhang et al., 2010) and UC3M have explicitly used supervised classification based on many lexical level and name tagging features, and they are ranked the 3rd, 4th, 5th, 6th and 10th as shown in Figure 3. Therefore we can conclude that supervised learning normally leads to a reasonably good performance. However, a high-performing entity linking system can also be implemented in an unsupervised fashion by exploiting effective characteristics and algorithms, as we will discuss in the next sections.

7.2 Impact of Semantic Features

Almost all entity linking systems have used semantic annotations as features, including name tagging, Wikipedia infoboxes and semantic categories. Two systems (BuptPris and CUNY) submitted alternative runs in order to evaluate the impact of using these features.

The semantic features used in the BuptPris system include name tagging, infoboxes, synonyms, variants and abbreviations. In the CUNY system, the semantic features are automatically extracted from their slot filling system. The results are summarized in Table 11. As we can see, except for person entities in the BuptPris system, all of the other types of entities have obtained significant improvement by using semantic features in linking.

System	Using Semantic Features	PER	ORG	GPE	Overall
BuptPris	No	83.89	59.47	33.38	58.93
	Yes	79.09	74.13	66.62	73.29
CUNY	No	84.55	63.07	57.54	59.91
	Yes	92.81	65.73	84.10	69.29

Table 11. Impact of Semantic Features on Entity Linking (Micro-Averaged Accuracy %)

7.3 Impact of Wikipedia Structure Mining

Wikipedia articles are peppered with structured information and hyperlinks to other (on average 25) articles (Medelyan et al., 2009). Such information provides an additional source for entity linking. For example, WebTLab team (Fernandez et al., 2010) used Wikipedia link structure (source, anchors, redirects and disambiguation) to compute instance co-occurrence estimates. Stanford-UBC team (Chang et al., 2010) used Wikipedia hyperlinks (clarification, disambiguation, title) for query re-mapping, and encoded features to train a supervised classifier; they reported a significant improvement on micro-averaged accuracy from 74.85% to 82.15%. Many other teams including CUNY (Chen et al., 2010) and IIIT (Bysani et al., 2010) used redirect pages and disambiguation pages for query expansion. IIIT team also exploited bold texts from first paragraphs because they often contain nick names, alias names and full names.

In fact, when the mined attributes becomes rich enough, they can be used as an expanded query and sent into an information retrieval engine in order to obtain the relevant source documents. Budapestacad team (Nemeskey et al., 2010) adopted this strategy.

8 Discussion of Slot Filling

Regular slot filling remains a very challenging task. The goal of this section is to lay out the current status and potential challenges of slot filling, and suggest some possible research directions.

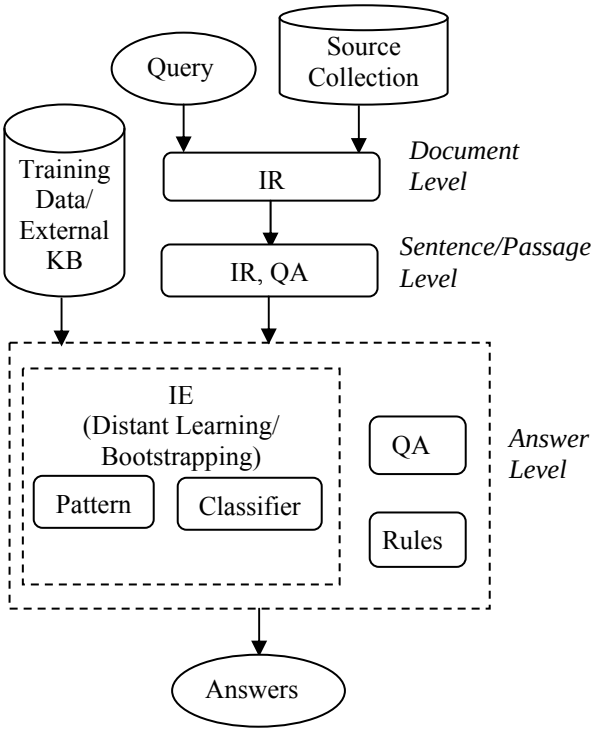

Figure 9. General Slot Filling System Architecture

8.1 A General Architecture

Various techniques are exploited in different slot filling systems this year, but they generally follow a common architecture summarized in Figure 9.

Almost all systems used IR techniques to retrieve documents and sentences, except that iirg team (Byrne and Dunnion, 2010) used QA methods to obtain sentences. To extract answers, many diverse methods have been proposed, as presented in Table 12. Most systems used one main pipeline, while CUNY system adopted a hybrid approach of combining three pipelines.

8.2 Slot-Specific Analysis

Table 13 and 14 give some basic statistics for the various slots for the evaluation corpus.

We can observe first of all that a few slots account for a large fraction of the answers, whereas others appear rarely. In particular, four slots associated with employment and personal membership (per:title, per:employee_of, per:member_of, and org:top_members/employees) together account for 388 of the 1057 correct responses (37%), counted in terms of equivalence classes.

Methods			System Examples	Characteristics
IE	Pattern Learning	Distant Learning (large seeds, one iteration)	CUNY (Chen et al., 2010)	High Recall, Low Precision
		Bootstrapping (small seeds, multiple iterations)	NYU (Grishman and Min, 2010)	High Precision, Low Recall
	Supervised Classifier	Distant Supervision	UBC (Intxaurrondo et al., 2010), lsv (Chrupala et al., 2010), Stanford (Surdeanu et al., 2010)	
		Training Data	CUNY (Chen et al., 2010), IIIT (Bysani et al., 2010), LCC (Lehmann et al., 2010), IBM (Castelli et al., 2010)	
QA			CUNY (Chen et al., 2010), iirg (Byrne and Dunnion, 2010)	
Rules			USFD (Yu et al., 2010)	

Table 12. Answer Extraction Method Comparison

SLOT	Entities	Equivs	Correct	Inexact
per:age	34	49	61	18
per:alternate_names	27	38	39	14
per:cause_of_death	1	3	3	0
per:charges	5	11	13	12
per:children	10	13	20	7
per:cities_of_residence	30	43	46	6
per:city_of_birth	9	9	9	3
per:city_of_death	1	1	1	0
per:countries_of_residence	11	14	19	4
per:country_of_birth	9	9	12	1
per:country_of_death	0	0	0	0
per:date_of_birth	11	11	23	1
per:date_of_death	1	1	2	0
per:employee_of	37	71	86	29
per:member_of	17	44	66	17
per:origin	19	28	31	7
per:other_family	12	28	35	2
per:parents	14	23	35	8
per:religion	3	4	4	0
per:schools_attended	13	16	18	1
per:siblings	12	22	38	8
per:spouse	14	15	20	10
per:stateorprovince_of_birth	7	8	8	1
per:stateorprovince_of_death	1	1	1	0
per:stateorprovinces_of_residence	21	24	28	1
per:title	45	149	191	102

Table 13. Statistics on Person Slots in 2010 Evaluation Data
Entities = number of entities (out of 50) with some value for this slot.
Equivs = number of equivalence classes of correct responses.
Correct = number of distinct correct string fills (ignoring case)
Inexact = number of distinct inexact string fills (ignoring case)

SLOT	Entities	Equivs	Correct	Inexact
org:alternate_names	37	58	69	20
org:city_of_headquarters	29	30	34	5
org:country_of_headquarters	23	23	31	7
org:dissolved	3	3	4	0
org:founded	14	14	17	0
org:founded_by	12	13	20	6
org:member_of	2	2	2	0
org:members	4	17	17	0
org:number_of_employees/members	13	15	17	9
org:parents	12	15	24	3
org:political/religious affiliation	5	8	8	0
org:shareholders	3	5	5	0
org:stateorprovince_of_headquarters	25	26	35	4
org:subsidiaries	17	53	55	13
org:top_members/employees	44	124	148	30
org:website	16	16	25	2

Table 14. Statistics on Organization Slots in 2010 Evaluation Data (see legend for Table 13)

Inexact responses are typically those where a system found the general location of a correct answer, but was unable to delimit the answer: too many or too few words were included. Slots where inexact was a substantial fraction of correct are those that were particularly difficult to delimit. Notable here are per:charges, where the correct response may be one of several types of linguistic constituents, and per:title, where it may be difficult to determine which modifiers are part of the title For example, "rookie" is not part of the title for someone described as a "rookie driver"; "record" is an essential part of the title for a "record producer".

A list-valued slot where the number of correct answer strings is appreciably greater than the number of equivalence classes indicates a slot for which equivalent-answer detection is important. per:title again accounts for the largest number of cases. For example "defense minister" and "defense chief" should be treated as equivalent.

8.3 How Much Inference is Needed?

The KBP slot filling task has proven to be a much more challenging task than ACE relation mention and event mention detection, because it needs cross-sentence and cross-slot inference. In Figure 10 we present the distribution of various cases in the KBP2010 training corpora which need different techniques.

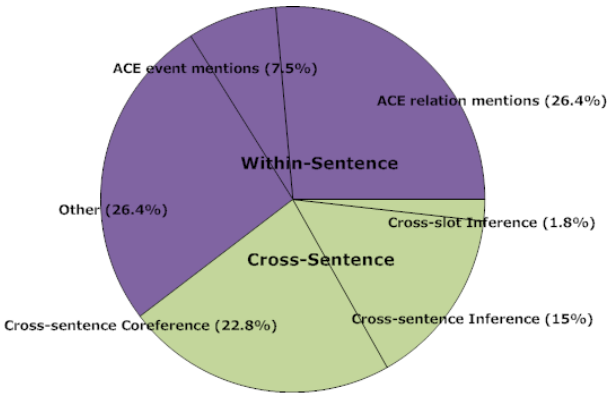

Figure 10. Distribution of Slot Filling 2010 Training Data

Among the 492 answers analyzed, in only 60.4% do the query and answer pairs appear in the same sentence. The remaining 39.6% require a system to go beyond single-sentence extraction. As in ACE, cross-sentence coreference resolution plays a crucial role; it is essential here for 22.8% of the cases.

Table 15 presents examples for each case. In the fourth example, a sentence-by-sentence approach may identify "he" as the *org:founder* of the query "*International Steel Group*", but coreference resolution is needed to link "he" back to "*Wilbur L. Ross Jr.*" in order to generate "*Wilbur L. Ross Jr.*" as the final answer; The remaining, most challenging cases require bridging anaphora resolution or other forms of cross-sentence or cross-slot reasoning, especially for the following slot types: *org:subsidiaries, org:top_members/employees, org:alternate_names, per:cities_of_residence, per:employee_of, per:title and per:stateorprovinces_of_residence*.

The inferences include:

- Non-identity coreference: in row 5 of the table, the semantic relation between "*children*" and "*son*" needs to be exploited in order to generate "*Emile Emile Lahoud*" as the *per:children* of the query entity "*Lahoud*";
- Cross-slot inference based on revertible queries, propagation links or even world knowledge (see more details in the CUNY system description) to capture some of the most challenging cases. In the KBP slot filling task, slots are often dependent on each other, so we can improve the results by improving the "coherence" of the story (i.e. consistency among all generated answers (query profiles)). In example 6 in the table, the following reasoning rules are needed to generate the answer "Henry Daniel Moder" as *per:children* of "Danny Moder":
  ChildOf ("Henry Daniel Moder", "Julia Roberts*") ∧ Coreferential (*"Julia Roberts", "Roberts*") ∧ SpouseOf (*"Roberts", "Danny Moder*") → ChildOf (*"Henry Daniel Moder", "Danny Moder*")*

For another example, after we determine the value of *per:age* for the query "Sean Preston" to be "2", we can immediately remove all of the candidate answers for *per:employee_of* because by world knowledge a two-year-old child doesn't work for any employers.

Such inferences (and bridging anaphora) play a smaller role in ACE relations and events because of the requirement that the arguments of an ACE

relation mention or event mention appear within a single sentence.

A simple approach to constructing a KBP slot-filling system is to build on top of an existing information extraction system, and in particular one for ACE relations and events. This has the benefit that a large amount of training data is available for ACE. However, even when the query and answer appear in the same sentence, a large portion (26.4% of all slots) cannot be extracted directly from ACE relation/event mentions. Some slots are not covered at all by ACE types (e.g. *per:age*, *org:website*, *org:number_of_employees/members* and all the surprise slot types), while others require some non-NLP types of shallow reasoning in addition to the information provided by the ACE entities, relations, and events. For example,

- *alternative_name* slots need some further filtering beyond traditional coreference resolution;
- *per:places_of_residence* and *org:headquarters* slots are not directly mapped to the ACE "located_in" relation;
- *per:origin* is not the same as the "GPE-affiliation" relation in ACE;
- *per:title* has different definitions in KBP and ACE;
- "place" slots must be divided into country/province-or-state/city;
- *org:top_members/employees* slots have a vague boundary.

This partial match naturally leads to a hybrid approach combining existing information extraction capabilities with distant learning and reasoning methods. Participants such as IBM, iirg, and CUNY explored this direction. We hope some of the non-NLP reasoning burden can be alleviated through improved task specification and additional resources in KBP2011. For example, we aim to provide participants with lists of country and state/province names and top_employee titles, so that the participants don't need to debate the definitions of specific slot types and rather focus more research effort on extraction itself.

<query, answer>	characteristics/ Techniques	Example	
		Slot type	Context Sentence
Within-Sentence	ACE relation mentions	org: subsidiary	So his half brother arranged a 15-year mortgage from ***WMC Mortgage Co.***, a subprime division of ***General Electric***
	ACE event mentions	org: founder	…said ***William Dallas***, the founder of ***Ownit Mortgage Solutions***, a lending business in which Merrill bought a stake a few years ago.
	Other	per: orgin	***Michael Johns***, a 29-year-old rocker who moved from ***Australia*** to the U.S
Cross-Sentence	cross-sentence coreference	org: founder	The failures also allowed for the renegotiation of labor contracts, something ***Wilbur L. Ross Jr.***, a specialist in distressed assets, realized when he began looking at the moribund industry. The only *bidder* for the bankrupt LTV Steel, *he* proceeded to buy Bethlehem and other old-line companies, putting them together as ***International Steel Group***.
	cross-sentence inference	per: children	***Lahoud*** is married to an Armenian and the couple have *three children*. Eldest *son* ***Emile Emile Lahoud*** was a member of parliament between 2000 and 2005.
	cross-slot inference	per: children	People Magazine has confirmed that actress *Julia Roberts* has given birth to her third child a boy named ***Henry Daniel Moder***. Henry was born Monday in Los Angeles and weighed 8? lbs. *Roberts*, 39, and husband ***Danny Moder***, 38, are already parents to twins Hazel and Phinnaeus who were born in November 2006.

Table 15. Non-Inference and Inference Examples in Slot Filling 2010 Training Data

8.4 Impact of Using External Knowledge Base

Many instance-centered knowledge bases that have harvested Wikipedia are proliferating on the semantic web. The most well known are probably the Wikipedia derived ontologies, including DBpedia (Auer2007), which serves as a hub in the Linked Data Web, Freebase (Bollacker2008), which harvests information from many open data sets (for instance Wikipedia and MusicBrainz), as well as from user contributions, and YAGO (Suchanek2007), which adds an ontological structure on top of Wikipedia's entities. The main motivation of the KBP program is to automatically distill information from news and web unstructured data instead of manually constructed knowledge bases, but these existing knowledge bases can provide a large number of seed tuples to bootstrap slot filling or distant learning. For example, the UBC team used Freebase for distant learning.

Such resources can also be used in a more direct way. For example, CUNY exploited Freebase and LCC exploited DBPedia as fact validation in slot filling. However, most of these resources are manually created from single data modalities and only cover well-known entities. For example, while Freebase contains 116 million instances of 7,300 relations for 9 million entities, it only covers 48% of the slot types and 5% of the slot answers in KBP2010 evaluation data. Therefore, both CUNY and LCC observed limited gains from the answer validation approach from Freebase (as shown in Table 16). Both systems gained about 1% improvement in recall with a slight loss in precision.

System	Use Answer Validation from Freebase?	Precision	Recall	F-measure
LCC	No	45.33	18.76	26.54
	Yes	44.87	19.44	27.13
CUNY	No	27.99	26.02	26.97
	Yes	28.74	27.85	28.29

Table 16. Impact of Using External Knowledge Base on Slot Filling (%)

8.5 Efficient Generation of Annotated Data

The Slot Filling evaluation results clearly indicate that further advances requires us to pursue an effective way of creating answer-keys. It has been demonstrated in many settings that the combination of several algorithms for a prediction problem can result in better performance than any one system alone. In this section we will discuss the possibilities of various combinations across systems and human annotators.

Recently Amazon Mechanical Turk (AMT) has become a popular method to obtain annotations for NLP tasks at low cost. We have also attempted to use AMT to assess system output. Given a query, answer and supporting context sentence, a Turk should judge whether the answer is correct (Y), incorrect (N) or uncertain (U).

Table 17 shows the distribution of assessment results on 1690 instances. We can see that only 41.8% of the voted results seem acceptable. However, even for the cases agreed upon by all five assessors, many results are incorrect. For example, Table 18 shows two answers which were mistakenly judged as correct. Automatic quality control methods and presenting the Turk with training and test material for a single slot type may help improve the results. We suspect, however, that the current KBP slot filling task may be too challenging for non-expert annotators.

Acceptable Voting (41.8%)		Unacceptable Voting (58.2%)	
Cases	**Number**	**Cases**	**Number**
Y Y Y Y Y	230	Y Y Y N N	164
N N N N N	16	Y Y Y N U	165
Y Y Y Y U	151	N N N Y Y	158
N N N N U	24	Y Y N N U	171
Y Y Y Y N	227	Y Y N U U	77
N N N N Y	46	Y Y U U U	17
Y Y Y U U	13	N N N Y U	72
N N N U U	59	N N Y U U	57
		Y N U U U	22
		Y U U U U	8
		N N U U U	11
		N U U U U	1
		U U U U U	1

Table 17. AMT Assessment Results

Query	Slot	Answer	Context
Citibank	org:top_members /employees	**Tim Sullivan**	He and **Tim Sullivan**, **Citibank**'s Boston area manager, said they still to plan seek advice from activists going forward.
International Monetary Fund	org: subsidiaries	**World Bank**	President George W. Bush said Saturday that a summit of world leaders agreed to make reforms to the **World Bank** and **International Monetary Fund**.

Table 18. Incorrect AMT Assessment Examples

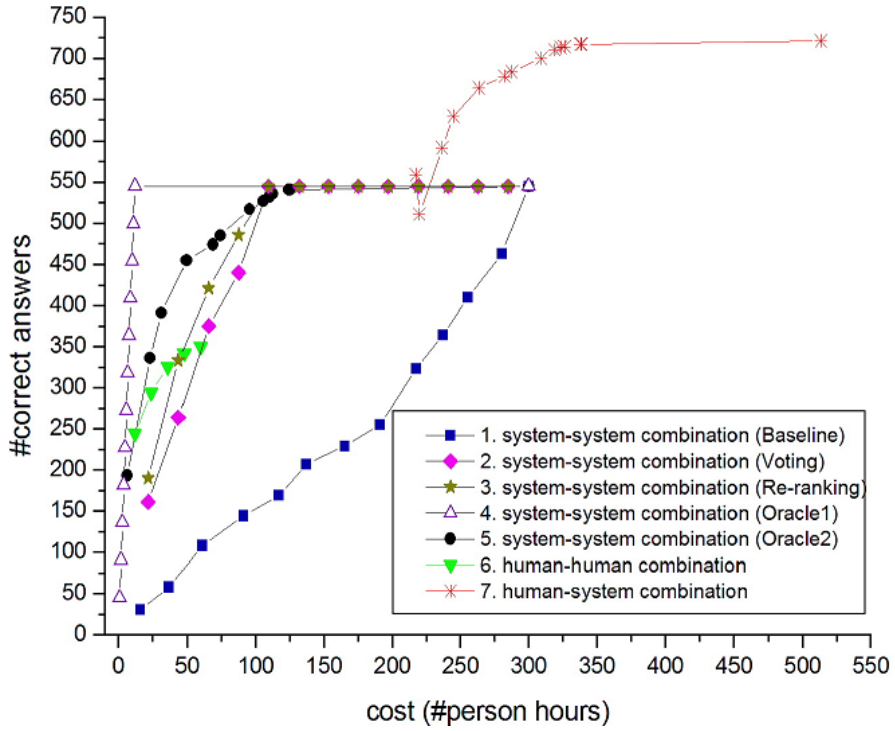

Figure 11. Answer-Key Learning Method Comparison

Web Data Percentage	Slot Types (#)	
0	Per	(0)
	Org	dissolved, members, number_of_employees/members, shareholders **(4)**
(0, 40]	Per	date_of_birth, diseases (2)
	Org	founded_by, founded, city_of_headquarters, top_members,employees, subsidiaries, stateorprovince_of_headquarters **(6)**
[40, 60]	Per	schools_attended, member_of, awards_won, country_of_birth, countries_of_residence, parents (6)
	Org	country_of_headquarters, political/religious_affiliation, website (3)
(60, 80]	Per	origin, city_of_birth, alternate_names, religion, siblings, children, other_family, age, charities_supported, title, employee_of(11)
	Org	products **(1)**
(80, 100]	Per	stateorprovinces_of_residence, cities_of_residence, spouse, stateorprovince_of_birth, cause_of_death, charges (6)
	Org	**(0)**

Table 20. Percentages of Correct Answers from Web Texts in Slot Filling

Next we compared the efficiency of various approaches to producing annotated data, either using direct manual annotation or by the assessment of system output. For the comparison, we used various combinations across human annotators, 13 systems (iirg1, IBM3, lcc2, CUNY3, NYU3, BuptPris1, Stanford2, ICL_KBP2, lsv1, hltcoe1, siel101, budapestacad2, ubc3, usfd1) as applied to the 2010 evaluation data, and LDC assessors as follows.

- Automatically combine and assess the pooled system responses based on:
  1. Alphabetical order (by site name);
  2. Majority voting; the responses which get more votes across systems are assessed first;
  3. Re-ranked by a statistical re-ranking model (Chen et al., 2010);
- Combine and assess the pooled system responses using two oracles (for upper-bound analysis):
  4. Assess all correct responses first;
  5. Ranked by the oracle system performance (the best to the worst);
- 6. Combine the human annotations from five independent annotators for six queries in the community effort;
- 7. Combine LDC human annotation with 5.

Figure 11 summarizes the results from the above 7 approaches.

We assume for this figure a labor cost for assessment proportional to the number of non-NIL items assessed. This is only approximately correct; it may be faster (per response) to assess more responses to the same slot. The common end point of curves 1-5 represents the cost and benefit of assessing all system responses. The starting point of curve 7 is the cost and yield of manual annotation; the end point of curve 7 combines manual annotation with assessment of all system responses.

Comparing the starting point of curve 7 (pure manual annotation) with points along curves 1 to 5 allows us to compare manual annotation with various strategies for assessing pooled system responses. The baseline (curve 1, assessing systems in alphabetical order by site) is less efficient than direct manual annotation. This reflects the fact that some of the systems have very low precision. However, if we employ automatic statistical voting (curve 2) or a statistical re-ranking approach (curve 3) and apply some cut-off, the process can be dramatically more efficient than manual annotation at nearly the same recall. In particular, the performance of the statistical re-ranking approach is very close to Oracle 2. This suggests an alternative way to annotate answer keys for slot filling – re-ranking and assessing the pooled system responses as opposed to identifying correct answers from scratch.

While pooling system responses (without human annotation in the pool) and then manually reviewing them may be more efficient in terms of the rate of gathering correct responses, this must be done with some caution. If no system incorporates a particular type of processing, answers which require this type of processing will not be picked up. If further system development is driven by shortfalls in recall and precision relative to an answer key based on system pooling, this type of processing may not get added. For example, based on analysis of slots successfully filled by LDC annotators but none of the systems, we suspect that all of the systems were limited in their ability to handle

- texting-style alternate names ("K8" for "Kate")
- implicit tables in weblogs
- reasoning about family relations (my father's brother is my uncle, a *per:other_family*)
- disjoint antecedents of plural nouns or pronouns (X married Y and *they* went on a honeymoon*)*

9 Impact of Text Coherence

In the current setting of KBP, a set of target entities is provided to each system in order to simplify the task and its evaluation, because it's not feasible to ask any system to generate answers for all possible entities in the entire source collection. However, ideally a fully-automatic KBP system should be able to automatically discover novel entities ("queries") which have no KB entries or rare information in KB, extract their attributes, and conduct global reasoning on top of these attributes in order to generate the final output. At the very least, due to the semantic coherence principle (McNamara, 2001), the information of an entity depends on the information of other entities. Several teams in KBP2010 attempted capturing this intuition by making up new queries in the context of any given

query, and conduct global ranking or reasoning to refine the results.

For example, in the entity linking task, WebTLab team (Fernandez et al., 2010) and CMCRC team (Radford et al., 2010) extracted all entities in the context of a given query, and disambiguated all entities at the same time using PageRank-like algorithm (Pages et al., 1998) or Graph-based Re-ranking algorithm. The SMU-SIS team (Gottipati and Jiang, 2010) re-formulated queries using contexts, for example, convert a query "Cambridge" into "Cambridge, Massachusetts, the United States" or "Cambridge, Ontario, Canada" based on the contexts in the source document and Wikipedia text. They did not report the impact of using this approach but stated that it played an important role in entity disambiguation. The LCC team (Lehmann et al., 2010) modeled contexts using Wikipedia page concepts, and computed linkability scores iteratively.

In the slot filling task, a slot is often dependent on other slots. In particular, the family slots include such dependency relationships (e.g. *X is per:children of Y → Y is per:parents of X; X is per:spouse of Y → Y is not likely to be per:siblings of X*). Therefore CUNY (Chen et al., 2010) and IBM (Castelli et al., 2010) teams developed recursive reasoning components to refine extraction results. For a candidate <*q, a, slot-type*$_i$>, if there are no other related answer candidates available, they built "revertible" queries in the contexts, similar to (Prager et al., 2006), to enrich the inference process iteratively.

Table 19 summarized the impact of using this idea for two tasks by different systems. We can see that consistent improvements were achieved across systems, especially the IBM slot filling system significantly benefited from recursive reasoning.

Using Text Coherence	Web TLab (EL)	CMCRC (EL)	CUNY (SF)	IBM (SF)
No	63.64	86.8	33.57	26
Yes	66.58	87.1	35.29	34.83

Table 19. Impact of Text Coherence (%, EL: Micro-Averaged Accuracy; SF: F-Measure)

10 Impact of Supervised System Combination

The increasing number of diverse approaches provide new opportunities for both entity linking and slot filling to benefit from system combination.

NUSchime entity linking system (Zhang et al., 2010) trained a SVM based re-scoring model to combine two individual pipelines. Only one feature based on confidence values from pipelines was used for re-scoring. The micro-averaged accuracy was enhanced from 79.29%/79.07% to 79.38% after combination.

The CUNY (Chen et al., 2010) slot filling system trained a MaxEnt based re-ranking model to combine three individual pipelines, based on various global features including voting and dependency relations. Significant gain in F-measure was achieved: from 17.9%, 27.7%, 21.0% to 34.3% after combination.

11 Impact of Data Genre

Finally, one of the major new aspects for entity linking this year is the addition of a substantial amount of web data. Figure 12 presents the detailed top-10 system performance breakdown for the two genres for all queries.

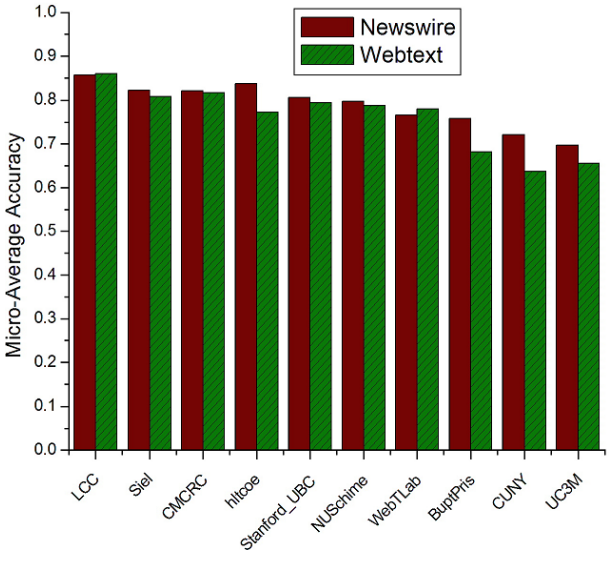

Figure 12. Top-10 Regular Entity Linking Performance on Different Genres (All Queries)

Overall entity linking performance was generally better on the newswire data. The Wilcoxon Matched-Pairs Signed-Ranks Test across all 16 systems showed that the difference between mixed

genre and newswire is significant at 99.83% confidence level; the difference between newswire and web text is significant at 99.58% confidence level. However we still observed that 3 systems achieved higher performance on web texts than newswire data, and 4 systems obtained similar performance on these two genres.

Furthermore, Figure 13 presents the comparison on non-NIL queries, surprisingly most systems performed better on web data than newswire, possibly because the non-NIL queries selected from web data were relatively common with less ambiguous source documents.

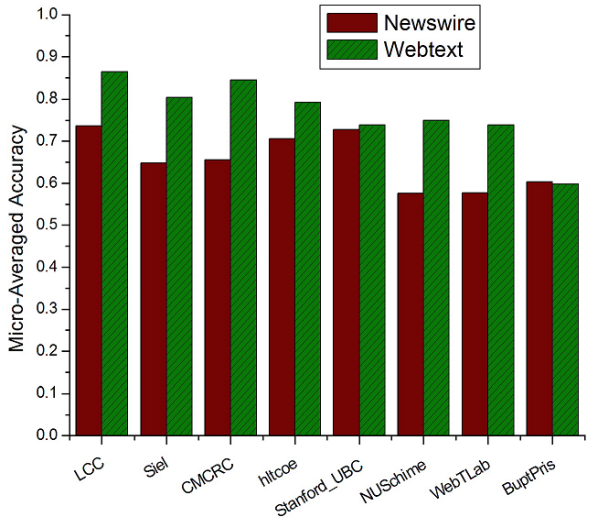

Figure 13. Top-10 Regular Entity Linking Performance on Different Genres (Non-NIL Queries)

Table 20 summarizes the percentages of correct instances of <query, slot-type, answer, docid> extracted from web texts in the slot filling task. We only included the slot types which involve more than 2 instances in both genres. Because the amount of web text (WT) is only 30% of that of newswire documents (NW), the percentage is counted as (#WT/(#WT+#NW*30%))

As we can see, most organization slots appear primarily in newswire documents, except for information related to web pages such as "products" and "website". However, a large percentage of person slots appear in the web texts, especially for personal information such as family members and employment. These results may be helpful for data selection in distant learning.

12 Impact of Web Access

KBP2010 doesn't allow official runs to use web access. However it's still interesting to measure the impact of using web access. Figure 14 presents the results for the systems that have submitted comparable runs with and without web access.

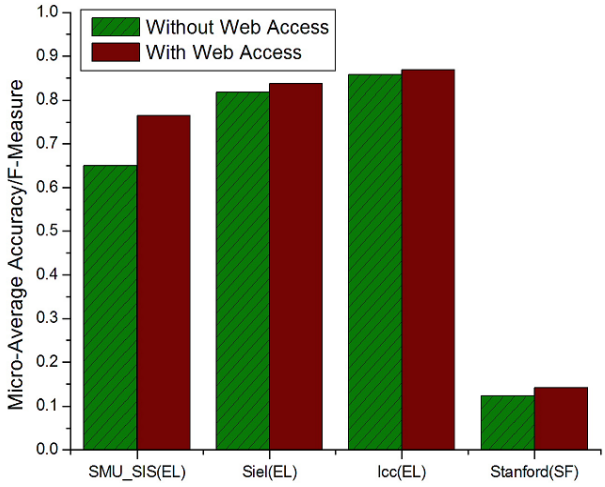

Figure 14. Impact of Web Access on Entity Linking (EL) and Slot Filling (SF)

As we can see, using the information available through web access (e.g. popularity ranks for entity linking and short answer snippets for slot filling) can enhance both tasks significantly. Previous literature has also shown that the web popularity rank is an important feature in state-of-the-art entity linking systems. Some preliminary results using the CUNY system have also shown that using the web popularity rank feature alone can beat a more sophisticated feature driven system without web access. Since the web information is used as a black box which changes over time, it's more difficult for duplicating research results. Therefore we expect to continue disallowing web access for official runs in the future; however, this kind of high baseline may be useful to generate entity linking training data in a semi-automatic way.

13 Conclusion

The Knowledge Base Population task for 2010 was marked by a growing variety of tasks and a growing pool of participants.

In Entity Linking, we saw a general improvement in performance over last year's results. When

measured against a benchmark based on 3-way inter-annotator agreement, system performance approached and in some cases (and for some entity types) exceeded the benchmark. Performance on new weblog material was close to that on the more traditional newswire corpus. An optional entity linking task required systems to make linking decisions based on the knowledge base alone, and not on the Wikipedia text associated with individual KB entries. Performance on the optional task was slightly worse, but the differences were not statistically significant.

In Slot Filling, the selection of query entities and the target evaluation metric both changed, making comparison with prior results more difficult. One system (using a very large supplementary corpus with substantial hand annotation) obtained results on a par with human annotation, and five other systems were within a factor of two. A wide variety of approaches were represented and the substantial training corpora produced as part of this year's effort should lead to a better understanding of the shortcoming of these approaches and continued progress in this area.

References

S. Auer and C. Bizer and G. Kobilarov and J. Lehmann and Z. Ives. 2007. DBpedia: A nucleus for a web of open data. *Proc. 6th International Semantic Web Conference*.

K. Bollacker, R. Cook, and P. Tufts. 2007. Freebase: A Shared Database of Structured General Human Knowledge. *Proc. National Conference on Artificial Intelligence* (Volume 2).

Lorna Byrne and John Dunnion. 2010. UCD IIRG at TAC 2010. *Proc. TAC 2010 Workshop.*

Praveen Bysani, Kranthi Reddy, Vijay Bharath Reddy, Sudheer Kovelamudi, Prasad Pingali and Vasudeva Varma. 2010. IIIT Hyderabad in Guided Summarization and Knowledge Base Population. *Proc. TAC 2010 Workshop*.

Vittorio Castelli, Radu Florian and Ding-jung Han. 2010. Slot Filling through Statistical Processing and Inference Rules. *Proc. TAC 2010 Workshop*.

Angel X. Chang, Valentin I. Spitkovsky, Eric Yeh, Eneko Agirre and Christopher D. Manning. 2010. Stanford-UBC Entity Linking at TAC-KBP. *Proc. TAC 2010 Workshop*.

Zheng Chen, Suzanne Tamang, Adam Lee, Xiang Li, Wen-Pin Lin, Matthew Snover, Javier Artiles, Marissa Passantino and Heng Ji. 2010. CUNY-BLENDER TAC-KBP2010 Entity Linking and Slot Filling System Description. *Proc. TAC 2010 Workshop*.

Grzegorz Chrupala, Saeedeh Momtazi, Michael Wiegand, Stefan Kazalski, Fang Xu, Benjamin Roth, Alexandra Balahur, Dietrick Klakow. Saarland University Spoken Language Systems at the Slot Filling Task of TAC KBP 2010. *Proc. TAC 2010 Workshop*.

Norberto Fernandez, Jesus A. Fisteus, Luis Sanchez and Eduardo Martin. 2010. WebTLab: A Cooccurence-based Approach to KBP 2010 Entity-Linking Task. *Proc. TAC 2010 Workshop*.

Swapna Gottipati and Jing Jiang. 2010. SMU-SIS at TAC 2010 – KBP Track Entity Linking. *Proc. TAC 2010 Workshop*.

Ralph Grishman and Bonan Min. 2010. New York University KBP 2010 Slot-Filling System. *Proc. TAC 2010 Workshop*.

Ander Intxaurrondo, Oier Lopez de Lacalle and Eneko Agirre. 2010. UBC at Slot Filling TAC-KBP2010. *Proc. TAC 2010 Workshop*.

John Lehmann, Sean Monahan, Luke Nezda, Arnold Jung and Ying Shi. 2010. LCC Approaches to Knowledge Base Population at TAC 2010. *Proc. TAC 2010 Workshop*.

Paul McNamee and Hoa Dang. 2009. Overview of the TAC 2009 Knowledge Base Population Track. *Proc. TAC 2009 Workshop*.

Paul McNamee, Hoa Trang Dang, Heather Simpson, Patrick Schone and Stephanie M. Strassel. 2010. An Evaluation of Technologies for Knowledge Base Population. *Proc. LREC2010.*

Danielle S McNamara. 2001. Reading both High-coherence and Low-coherence Texts: Effects of Text Sequence and Prior Knowledge. *Canadian Journal of Experimental Psychology*.

Olena Medelyan, Catherine Legg, David Milne and Ian H. Witten. 2009. Mining Meaning from Wikipedia. *International Journal of Human-Computer Studies archive.* Volume 67 , Issue 9.

David Nemeskey, Gabor Recski, Attila Zseder and Andras Kornai. 2010. BUDAPESTACAD at TAC 2010. *Proc. TAC 2010 Workshop*.

Cesar de Pablo-Sanchez, Juan Perea and Paloma Martinez. 2010. Combining Similarities with Regression based Classifiers for Entity Linking at TAC 2010. *Proc. TAC 2010 Workshop*.

Lawrence Page, Sergey Brin, Rajeev Motwani and Terry Winograd. 1998. The PageRank Citation Ranking: Bringing Order to the Web. *Proc. the 7th International World Wide Web Conference.*

J. Prager, P. Duboue, J. Chu-Carroll. 2006. Improving QA Accuracy by Question Inversion. *Proc. ACL-COLING 2006.*

Will Radford, Ben Hachey, Joel Nothman, Matthew Honnibal and James R. Curran. 2010. CMCRC at TAC10: Document-level Entity Linking with Graph-based Re-ranking. *Proc. TAC 2010 Workshop*.

Mihai Surdeanu, David McClosky, Julie Tibshirani, John Bauer, Angel X. Chang, Valentin I. Spitkovsky, Christopher D. Manning. 2010. A Simple Distant Supervision Approach for the TAC-KBP Slot Filling Task. *Proc. TAC 2010 Workshop*.

F. M. Suchanek, G. Kasneci, and G. Weikum. 2007. Yago: A Core of Semantic Knowledge. *Proc. 16th International World Wide Web Conference.*

Jingtao Yu, Omkar Mujgond and Rob Gaizauskas. 2010. The University of Sheffield System at TAC KBP 2010. *Proc. TAC 2010 Workshop*.

Wei Zhang, Yan Chuan Sim, Jian Su and Chew Lim Tan. 2010. NUS-I2R: Learning a Combined System for Entity Linking. *Proc. TAC 2010 Workshop*.