Knowledge Base Population: Successful Approaches and Challenges

Heng Ji
Computer Science Department
Queens College and Graduate Center
City University of New York
New York, NY 11367, USA
hengji@cs.qc.cuny.edu

Ralph Grishman
Computer Science Department
New York University

New York, NY 10003, USA
grishman@cs.nyu.edu

Abstract

In this paper we give an overview of the Knowledge Base Population (KBP) track at the 2010 Text Analysis Conference. The main goal of KBP is to promote research in discovering facts about entities and augmenting a knowledge base (KB) with these facts. This is done through two tasks, *Entity Linking* – linking names in context to entities in the KB – and *Slot Filling* – adding information about an entity to the KB. A large source collection of newswire and web documents is provided from which systems are to discover information. Attributes ("slots") derived from Wikipedia infoboxes are used to create the reference KB. In this paper we provide an overview of the techniques which can serve as a basis for a good KBP system, lay out the remaining challenges by comparison with traditional Information Extraction (IE) and Question Answering (QA) tasks, and provide some suggestions to address these challenges.

1 Introduction

Traditional information extraction (IE) 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. On the other hand, traditional Question Answering (QA) evaluations made limited efforts at disambiguating entities in queries (e.g. Pizzato et al., 2006), and limited use of relation/event extraction in answer search (e.g. McNamee et al., 2008).

The Knowledge Base Population (KBP) shared task, conducted as part of the NIST Text Analysis Conference, aims to address and evaluate these capabilities, and bridge the IE and QA communities to promote research in discovering facts about entities and expanding a knowledge base with these facts. KBP is done through two separate sub-tasks, Entity Linking and Slot Filling; in 2010, 23 teams submitted results for one or both sub-tasks. A variety of approaches have been proposed to address both tasks with considerable success; nevertheless, there are many aspects of the task that remain unclear. What are the fundamental techniques used to achieve reasonable performance? What is the impact of each novel method? What types of problems are represented in the current KBP paradigm compared to traditional IE and QA? 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? What are the remaining challenges and what are the possible ways to address these challenges? In this paper we aim to answer some of these questions based on our detailed analysis of evaluation results.

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 1148–1158,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

2 Task Definition and Evaluation Metrics

This section will summarize the tasks conducted at KBP 2010. 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.

In the Entity Linking task, given a person (PER), organization (ORG) or geo-political entity (GPE, a location with a government) query that consists of a name string and a background document containing that name string, 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. The background document, drawn from the KBP corpus, serves to disambiguate ambiguous name strings.

In selecting among the KB entries, a system could make use of the Wikipedia text associated with each entry as well as the structured fields of each entry. In addition, there was an optional task where the system could only make use of the structured fields; this was intended to be representative of applications where no backing text was available. Each site could submit up to three runs with different parameters.

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 background document containing the name (again, 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. 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). KBP2010 defined 26 types of attributes for persons (such as the age, birthplace, spouse, children, job title, and employing organization) and 16 types of attributes for organizations (such as the top employees, the founder, the year founded, the headquarters location, and subsidiaries). Some of these attributes are specified as only taking a single value (e.g., birthplace), while some can take multiple values (e.g., top employees).

The reference KB includes hundreds of thousands of entities based on articles from an October 2008 dump of English Wikipedia which includes 818,741 nodes. The source collection includes 1,286,609 newswire documents, 490,596 web documents and hundreds of transcribed spoken documents.

To score Entity Linking, we take each query and 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.

To score Slot Filling, we first pool all the system responses (as is done for information retrieval evaluations) together with a set of manually-prepared slot fills. These responses are then assessed by hand. Equivalent answers (such as “Bill Clinton” and “William Jefferson Clinton”) are grouped into equivalence classes. Each system response is rated as correct, wrong, or redundant (a response which is equivalent to another response for the same slot or an entry already in the knowledge base). Given these judgments, we 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 × Recall × Precision) / (Recall + Precision)

3 Entity Linking: What Works

In Entity Linking, we saw a general improvement in performance over last year’s results – the top system achieved 85.78% micro-averaged accuracy. When measured against a benchmark based on inter-annotator agreement, two systems’ performance approached and one system exceeded the benchmark on person entities.

3.1 A General Architecture

A typical entity linking system architecture is depicted in Figure 1.

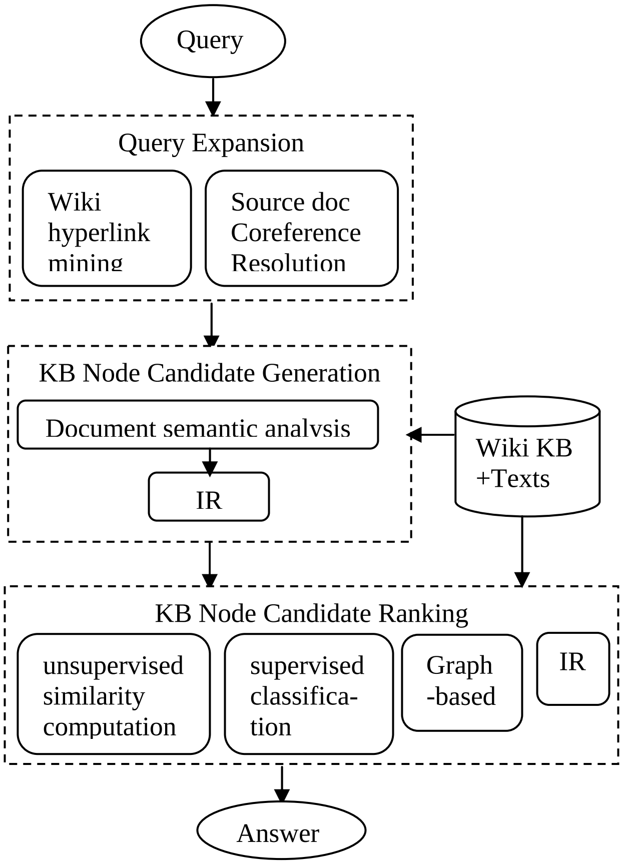
Query
Query Expansion
Wiki hyperlink mining
Source doc Coreference Resolution
KB Node Candidate Generation
Document semantic analysis
IR
Wiki KB +Texts
KB Node Candidate Ranking
unsupervised similarity computation
supervised classifica-tion
Graph -based
IR
Answer

Figure 1. General Entity Linking System Architecture

It includes three steps: (1) query expansion – expand the query into a richer set of forms using Wikipedia structure mining or coreference resolution in the background document. (2) candidate generation – finding all possible KB entries that a query might link to; (3) candidate ranking – rank the probabilities of all candidates and NIL answer.

Table 1 summarizes the systems which exploited different approaches at each step. In the following subsections we will highlight the new and effective techniques used in entity linking.

3.2 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 additional sources for entity linking: (1). Query Expansion: For example, WebTLab (Fernandez et al., 2010) used Wikipedia link structure (source, anchors, redirects and disambiguation) to extend the KB and compute entity co-occurrence estimates. Many other teams including CUNY and Siel used redirect pages and disambiguation pages for query expansion. The Siel team also exploited bold texts from first paragraphs because they often contain nicknames, alias names and full names.

Methods		System Examples	System Ranking Range
Query Expansion	Wikipedia Hyperlink Mining	CUNY (Chen et al., 2010), NUSchime (Zhang et al., 2010), Siel (Bysani et al., 2010), SMU-SIS (Gottipati et al., 2010), USFD (Yu et al., 2010), WebTLab team (Fernandez et al., 2010)	[2, 15]
	Source document coreference resolution	CUNY (Chen et al., 2010)	9
Candidate Generation	Document semantic analysis and context modeling	ARPANI (Thomas et al., 2010), CUNY (Chen et al., 2010), LCC (Lehmann et al., 2010)	[1,14]
	IR	CUNY (Chen et al., 2010), Budapestacad (Nemeskey et al., 2010), USFD (Yu et al., 2010)	[9, 16]
Candidate Ranking	Unsupervised Similarity Computation (e.g. VSM)	CUNY (Chen et al., 2010), SMU-SIS (Gottipati et al., 2010), USFD (Yu et al., 2010)	[9, 14]
	Supervised Classification	LCC (Lehmann et al., 2010), NUSchime (Zhang et al., 2010), Stanford-UBC (Chang et al., 2010), HLTCOE (McNamee, 2010), UC3M (Pablo-Sanchez et al., 2010)	[1, 10]
	Rule-based	LCC (Lehmann et al., 2010), BuptPris (Gao et al., 2010)	[1, 8]
	Global Graph-based Ranking	CMCRC (Radford et al., 2010)	3
	IR	Budapestacad (Nemeskey et al., 2010)	16

Table 1. Entity Linking Method Comparison

(2). Candidate Ranking: Stanford-UBC used Wikipedia hyperlinks (clarification, disambiguation, title) for query re-mapping, and encoded lexical and part-of-speech features from Wikipedia articles containing hyperlinks to the queries to train a supervised classifier; they reported a significant improvement on micro-averaged accuracy, from 74.85% to 82.15%. In fact, when the mined attributes become 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.

3.3 Ranking Approach Comparison

The ranking approaches exploited in the KBP2010 entity linking systems can be generally categorized into four 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 different features.

(3). Graph-based ranking, in which context entities are taken into account in order to reach a global optimized solution together with the query entity.

(4). IR (Information Retrieval) approach, in which the entire background source document is considered as a single query to retrieve the most relevant Wikipedia article.

The first question we will investigate is how much higher performance can be achieved by using supervised learning? Among the 16 entity linking systems which participated in the regular evaluation, LCC (Lehmann et al., 2010), HLTCOE (McNamee, 2010), Stanford-UBC (Chang et al., 2010), NUSchime (Zhang et al., 2010) and UC3M (Pablo-Sanchez et al., 2010) have explicitly used supervised classification based on many lexical and name tagging features, and most of them are ranked in top 6 in the evaluation. 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.

3.4 Semantic Relation Features

Almost all entity linking systems have used semantic relations as features (e.g. BuptPris (Gao et al., 2010), CUNY (Chen et al., 2010) and HLTCOE). 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 2, showing the gains over a baseline system (using only Wikipedia title features in the case of BuptPris, using tf-idf weighted word features for CUNY). As we can see, except for person entities in the BuptPris system, all types of entities have obtained significant improvement by using semantic features in entity 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 2. Impact of Semantic Features on Entity Linking (Micro-Averaged Accuracy %)

3.5 Context Inference

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 require a 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 entry or few slot fills in the KB, extract their attributes, and conduct global reasoning over 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. For example, the WebTLab team and the CMCRC team extracted all entities in the context of a given query, and disambiguated all entities at the same time using a PageRank-like algorithm (Page et al., 1998) or a Graph-based Re-ranking algorithm. The SMU-SIS team (Gottipati and Jiang, 2010) re-formulated queries using contexts. The LCC team modeled contexts using Wikipedia page concepts, and computed linkability scores iteratively. Consistent improvements were reported by the WebTLab system (from 63.64% to 66.58%).

4 Entity Linking: Remaining Challenges

4.1 Comparison with Traditional Cross-document Coreference Resolution

Part of the entity linking task can be modeled as a cross-document entity resolution problem which includes two principal challenges: 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 cross-document entity coreference resolution can be traced back to the Web People Search task (Artiles et al., 2007) and ACE2008 (e.g. Baron and Freedman, 2008). Compared to WePS and ACE, KBP requires linking an entity mention in a source document to a knowledge base with or without Wikipedia articles. Therefore sometimes the linking decisions heavily rely on entity profile comparison with Wikipedia infoboxes. In addition, KBP introduced GPE entity disambiguation. In source documents, especially in web data, usually few explicit attributes about GPE entities are provided, so an entity linking system also needs to conduct external knowledge discovery from background related documents or hyperlink mining.

4.2 Analysis of Difficult Queries

There are 2250 queries in the Entity Linking evaluation; for 58 of them at most 5 (out of the 46) system runs produced correct answers. Most of these queries have corresponding KB entries. For 19 queries all 46 systems produced different results from the answer key. Interestingly, the systems which perform well on the difficult queries are not necessarily those achieved top overall performance – they were ranked 13rd, 6th, 5th, 12nd, 10th, and 16th respectively for overall queries. 11 queries are highly ambiguous city names which can exist in many states or countries (e.g. "Chester"), or refer to person or organization entities. From these most difficult queries we observed the following challenges and possible solutions.

- **Require deep understanding of context entities for GPE queries**

In a document where the query entity is not a central topic, the author often assumes that the readers have enough background knowledge ('anchor' location from the news release information, world knowledge or related documents) about these entities. For 6 queries, a system would need to interpret or extract attributes for their context entities. For example, in the following passage:

> *…There are also photos of **Jake** on **IHJ** in **Brentwood**, still looking somber…*

in order to identify that the query "*Brentwood*" is located in California, a system will need to understand that "*IHJ*" is "*I heart Jake community*" and that the "*Jake*" referred to lives in Los Angeles, of which *Brentwood* is a part.

In the following example, a system is required to capture the knowledge that "*Chinese Christian man*" normally appears in "*China*" or there is a "*Mission School*" in "*Canton, China*" in order to link the query "*Canton*" to the correct KB entry. This is a very difficult query also because the more common way of spelling "*Canton*" in China is "*Guangdong*".

> *…and was from a **Mission School** in **Canton**, … but for the energetic efforts of this **Chinese Christian man** and the Refuge Matron…*

- **Require external hyperlink analysis**

Some queries require a system to conduct detailed analysis on the hyperlinks in the source document or the Wikipedia document. For example, in the source document "*…Filed under: Falcons <http://sports.aol.com/fanhouse/category/atlanta-falcons/>*", a system will need to analyze the document which this hyperlink refers to. Such cases might require new query reformulation and cross-document aggregation techniques, which are both beyond traditional entity disambiguation paradigms.

- **Require Entity Salience Ranking**

Some of these queries represent salient entities and so using web popularity rank (e.g. ranking/hit counts of Wikipedia pages from search engine) can yield correct answers in most cases (Bysani et al., 2010; Dredze et al., 2010). In fact we found that a naïve candidate ranking approach based on web popularity alone can achieve 71% micro-averaged accuracy, which is better than 24 system runs in KBP2010.

Since the web information is used as a black box (including query expansion and query log analysis) which changes over time, it's more difficult to duplicate research results. However, gazetteers with entities ranked by salience or major entities marked are worth encoding as additional features. For example, in the following passages:

> *... Tritschler brothers competed in gymnastics at the 1904 Games in* ***St Louis*** *104 years ago*" and "*A chartered airliner carrying Democratic White House hopeful Barack Obama was forced to make an unscheduled landing on Monday in* ***St. Louis*** *after its flight crew detected mechanical problems…*

although there is little background information to decide where the query "*St Louis*" is located, a system can rely on such a major city list to generate the correct linking. Similarly, if a system knows that "*Georgia Institute of Technology*" has higher salience than "*Georgian Technical University*", it can correctly link a query "*Georgia Tech*" in most cases.

5 Slot Filling: What Works

5.1 A General Architecture

The slot-filling task is a hybrid of traditional IE (a fixed set of relations) and QA (responding to a query, generating a unified response from a large collection). Most participants met this challenge through a hybrid system which combined aspects of QA (passage retrieval) and IE (answer extraction). A few used off-the-shelf QA, either bypassing question analysis or (if QA was used as a "black box") creating a set of questions corresponding to each slot.

The basic system structure (Figure 2) involved three phases: document/passage retrieval (retrieving passages involving the queried entity), answer extraction (getting specific answers from the retrieved passages), and answer combination (merging and selecting among the answers extracted).

The solutions adopted for answer extraction reflected the range of current IE methods as well as QA answer extraction techniques (see Table 3). Most systems used one main pipeline, while CUNY and BuptPris adopted a hybrid approach of combining multiple approaches.

One particular challenge for KBP, in comparison with earlier IE tasks, was the paucity of training data. The official training data, linked to specific text from specific documents, consisted of responses to 100 queries; the participants jointly prepared responses to another 50. So traditional supervised learning, based directly on the training data, would provide limited coverage. Coverage could be improved by using the training data as seeds for a bootstrapping procedure.

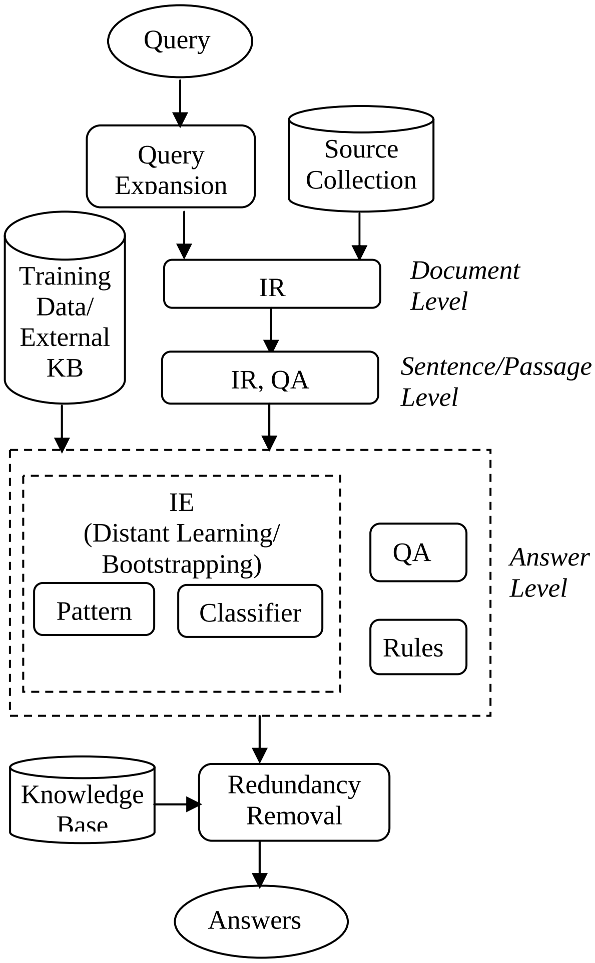

Figure 2. General Slot Filling System Architecture

Methods			System Examples
Trained IE	Pattern Learning	Distant Learning (large seed, one iteration)	CUNY (Chen et al., 2010)
		Bootstrapping (small seed, multiple iterations)	NYU (Grishman and Min, 2010)
	Supervised Classifier	Distant Supervision	Budapestacad (Nemeskey et al., 2010), lsv (Chrupala et al., 2010), Stanford (Surdeanu et al., 2010), UBC (Intxaurrondo et al., 2010)
		Trained from KBP training data and other related tasks	BuptPris (Gao et al., 2010), CUNY (Chen et al., 2010), IBM (Castelli et al., 2010), ICL (Song et al., 2010), LCC (Lehmann et al., 2010), lsv (Chrupala et al., 2010), Siel (Bysani et al., 2010)
QA			CUNY (Chen et al., 2010), iirg (Byrne and Dunnion, 2010)
Hand-coded Heuristic Rules			BuptPris (Gao et al., 2010), USFD (Yu et al., 2010)

Table 3. Slot Filling Answer Extraction Method Comparison

On the other hand, there were a lot of 'facts' available – pairs of entities bearing a relationship corresponding closely to the KBP relations – in the form of filled Wikipedia infoboxes. These could be used for various forms of indirect or distant learning, where instances in a large corpus of such pairs are taken as (positive) training instances. However, such instances are noisy – if a pair of entities participates in more than one relation, the found instance may not be an example of the intended relation – and so some filtering of the instances or resulting patterns may be needed. Several sites used such distant supervision to acquire patterns or train classifiers, in some cases combined with direct supervision using the training data (Chrupala et al., 2010).

Several groups used and extended existing relation extraction systems, and then mapped the results into KBP slots. Mapping the ACE relations and events by themselves provided limited coverage (34% of slot fills in the training data), but was helpful when combined with other sources (e.g. CUNY). Groups with more extensive existing extraction systems could primarily build on these (e.g. LCC, IBM).

For example, IBM (Castelli et al., 2010) extended their mention detection component to cover 36 entity types which include many non-ACE types; and added new relation types between entities and event anchors. LCC and CUNY applied active learning techniques to cover non-ACE types of entities, such as “origin”, “religion”, “title”, “charge”, “web-site” and “cause-of-death”, and effectively develop lexicons to filter spurious answers.

Top systems also benefited from customizing and tightly integrating their recently enhanced extraction techniques into KBP. For example, IBM, NYU (Grishman and Min, 2010) and CUNY exploited entity coreference in pattern learning and reasoning. It is also notable that traditional extraction components trained from newswire data suffer from noise in web data. In order to address this problem, IBM applied their new robust mention detection techniques for noisy inputs (Florian et al., 2010); CUNY developed a component to recover structured forms such as tables in web data automatically and filter spurious answers.

5.2 Use of 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 resources, including DBpedia (Auer 2007), Freebase (Bollacker 2008) and YAGO (Suchanek et al., 2007) and Linked Open Data (http://data.nytimes.com/). 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 guide 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. Both systems gained about 1% improvement in recall with a slight loss in precision.

5.3 Cross-Slot and Cross-Query Reasoning

Slot Filling can also benefit from extracting revertible queries from the context of any target query, and conducting global ranking or reasoning to refine the results. CUNY and IBM developed recursive reasoning components to refine extraction results. For a given query, 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. For example, if *a* is extracted as the answer for org:subsidiaries of the query *q*, we can consider *a* as a new revertible query and verify that a org:parents answer of *a* is *q*. Both systems significantly benefited from recursive reasoning (CUNY F-measure on training data was enhanced from 33.57% to 35.29% and IBM F-measure was enhanced from 26% to 34.83%).

6 Slot Filling: Remaining Challenges

Slot filling remains a very challenging task; only one system exceeded 30% F-measure on the 2010 evaluation. 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. The Precision, Recall and F-measure for the initial human annotation are only about 70%, 54% and 61% respectively. While we believe the annotation consistency can be improved, in part by refinement of the annotation guidelines, this does place a limit on system performance.

Most of the shortfall in system performance reflects inadequacies in the answer extraction stage, reflecting limitations in the current state-of-the-art in information extraction. An analysis of the 2010 training data shows that cross-sentence coreference and some types of inference are critical to slot filling. In only 60.4% of the cases do the entity name and slot fill appear together in the same sentence, so a system which processes sentences in isolation is severely limited in its performance. 22.8% of the cases require cross-sentence (identity) coreference; 15% require some cross-sentence inference and 1.8% require cross-slot inference. The inferences include:

- Non-identity coreference: in the following passage: “***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.” 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 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 the following example:

  “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.”*

  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*”)*

KBP Slot Filling is similar to ACE Relation Extraction, 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, making the role of coreference and cross-sentence inference more critical.

The role of coreference and inference as limiting factors, while generally recognized, is emphasized by examining the 163 slot values that the human annotators filled but that none of the systems were able to get correct. Many of these difficult cases involve a combination of problems, but we estimate that at least 25% of the examples involve coreference which is beyond current system capabilities, such as nominal anaphors:

> "Alexandra Burke is out with the video for her second single … taken from the British artist's debut album"
> "a woman charged with running a prostitution ring … her business, Pamela Martin and Associates"

(underlined phrases are coreferential).

While the types of inferences which may be required is open-ended, certain types come up repeatedly, reflecting the types of slots to be filled: systems would benefit from specialists which are able to reason about times, locations, family relationships, and employment relationships.

7 Toward System Combination

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

The NUSchime entity linking system trained a SVM based re-scoring model to combine two individual pipelines. Only one feature based on confidence values from the pipelines was used for re-scoring. The micro-averaged accuracy was enhanced from 79.29%/79.07% to 79.38% after combination. We also applied a voting approach on the top 9 entity linking systems and found that all combination orders achieved significant gains, with the highest absolute improvement of 4.7% in micro-averaged accuracy over the top entity linking system.

The CUNY slot filling system trained a maximum-entropy-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% and 21.0% (on training data) to 34.3% after combination. When we applied the same re-ranking approach to the slot filling systems which were ranked from the 2nd to 14th, we achieved 4.3% higher F-score than the best of these systems.

8 Conclusion

Compared to traditional IE and QA tasks, KBP has raised some interesting and important research issues: It places more emphasis on cross-document entity resolution which received limited effort in ACE; it forces systems to deal with redundant and conflicting answers across large corpora; it links the facts in text to a knowledge base so that NLP and data mining/database communities have a better chance to collaborate; it provides opportunities to develop novel training methods such as distant (and noisy) supervision through Infoboxes (Surdeanu et al., 2010; Chen et al., 2010).

In this paper, we provided detailed analysis of the reasons which have made KBP a more challenging task, shared our observations and lessons learned from the evaluation, and suggested some possible research directions to address these challenges which may be helpful for current and new participants, or IE and QA researchers in general.

Acknowledgements

The first author was supported by the U.S. Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053, the U.S. NSF CAREER Award under Grant IIS-0953149 and PSC-CUNY Research Program. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

Javier Artiles, Julio Gonzalo and Satoshi Sekine. 2007. The SemEval-2007 WePS Evaluation: Establishing a benchmark for the Web People Search Task. *Proc. the 4th International Workshop on Semantic Evaluations (Semeval-2007).*

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

K. Balog, L. Azzopardi, M. de Rijke. 2008. Personal Name Resolution of Web People Search. *Proc. WWW2008 Workshop: NLP Challenges in the Information Explosion Era (NLPIX 2008).*

Alex Baron and Marjorie Freedman. 2008. Who is Who and What is What: Experiments in Cross-Document Co-Reference. *Proc. EMNLP 2008*.

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*.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber and Tim Finin. 2010. Entity Disambiguation for Knowledge Base Population. *Proc. COLING 2010*.

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*.

Radu Florian, John F. Pitrelli, Salim Roukos and Imed Zitouni. 2010. Improving Mention Detection Robustness to Noisy Input. *Proc. EMNLP2010*.

Sanyuan Gao, Yichao Cai, Si Li, Zongyu Zhang, Jingyi Guan, Yan Li, Hao Zhang, Weiran Xu and Jun Guo. 2010. PRIS at TAC2010 KBP Track. *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.*

Paul McNamee, Rion Snow, Patrick Schone and James Mayfield. 2008. Learning Named Entity Hyponyms for Question Answering. *Proc. IJCNLP2008*.

Paul McNamee. 2010. HLTCOE Efforts in Entity Linking at TAC KBP 2010. *Proc. TAC 2010 Workshop*.

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.*

Luiz Augusto Pizzato, Diego Molla and Cecile Paris. 2006. Pseudo Relevance Feedback Using Named Entities for Question Answering. *Proc. the Australasian Language Technology Workshop 2006.*

J. Prager, P. Duboue, and 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*.

Yang Song, Zhengyan He and Houfeng Wang. 2010. ICL_KBP Approaches to Knowledge Base Population at TAC2010. *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.*

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*.

Ani Thomas, Arpana Rawai, M K Kowar, Sanjay Sharma, Sarang Pitale and Neeraj Kharya. 2010. Bhilai Institute of Technology Durg at TAC 2010: Knowledge Base Population Task Challenge. *Proc. TAC 2010 Workshop*.

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*.