UCD IIRG at TAC 2012

Lorna Byrne
School of Computer Science
& Informatics
University College Dublin
Ireland
lorna.byrne@ucd.ie

John Dunnion
School of Computer Science
& Informatics
University College Dublin
Ireland
john.dunnion@ucd.ie

Abstract

This notebook paper describes the IIRG system entered in the TAC 2012 Knowledge Base Population Track. The overall goal of KBP is to automatically identify salient and novel entities from multiple languages, link them to corresponding Knowledge Base (KB) entries (if the linkage exists) in a target language, then discover attributes about the entities (extract temporal spans about the attributes if there exist dynamic changes), and finally expand the KB with any new attributes.(Ji et al., 2010)

1 Introduction

The Knowledge Base Population Track is composed of two related tasks: Entity Linking (EL), which links entity mentions to their corresponding entities in the Knowledge Base, and Slot Filling (SF), which augments existing Knowledge Base entities with novel information. Entity Linking is comprised of the entity types Person (PER), Organisation (ORG), and Geo-Political Entity (GPE), whereas Slot Filling tasks are limited to PER and ORG entity types only. This year we participated in the Mono-Lingual Entity Linking Task and Regular English Slot Filling Task. The source document collection for the tasks consists of mainly newswire and Web documents. This year TAC provided participants with additional source documents thereby increasing the collection to approximately 3.7 million documents, more than double that of the previous years.

2 Regular English Slot Filling Task

Given the similarities between the slot filling task and Question Answering (QA), we have adapted a classical QA architecture (Pasca, 2003) for use in the Regular English Slot Filling task. The objective of SF systems is to return a slot-value (exact answer) in response to a slot query. Each slot query could easily be transformed into an equivalent question or set of questions for use in a QA system. The SF task could be viewed as a specialised QA task where the questions remain the same and only the target or focus of the questions change. Our system follows a pipeline architecture similar to that of a traditional Question Answering System as described in (Pasca, 2003). The SF pipeline consists of three main modules: Pre-Processing, which includes query and document collection processing, Passage Retrieval and Slot-Value Selection. We have implemented a two-stage retrieval module. In the first stage, we retrieve a set of documents in response to the target entity in the initial query using an off-the-shelf document retrieval component. In the second stage, we reduce this search space to passages in order to identify passages or segments that match generated candidate patterns and then extract a slot-value based on an identified slot-value type in as narrow a search space as possible.

2.1 Pre-Processing

In this phase, the source document collection was indexed using Terrier 3.0 (Ounis et al., 2006). Terrier[1] is a highly flexible, efficient and effective open-source search engine, readily deployable on large-scale collections of documents. Terrier implements state-of-the-art indexing and retrieval functionalities, and provides an ideal platform for the rapid development and evaluation of large-scale retrieval applications.

During this phase, we also used data from the previous Regular Slot Filling tasks as a source of training data. We created training examples by extracting sentences from the document collection which contained a slot-value and annotated occurences of the

[1] http://www.terrier.org

target entity and slot-value. In many cases, the slot-value can be found in close proximity to the target entity. For example, sentences extracted for the slot per:spouse were annotated as follows:

```
<target> is married to <slot-value>
<target> marries <slot-value>
```

Candidate patterns were then generated around the target and slot-value using a combination of the Stanford Part-of-Speech (POS) tagger and Named Entity Recognition (NER) tools[2] and verbs which occur are reduced to their canonical form to allow for greater coverage. This produces patterns of the form:

```
<person> VB:bear <date>
```

for the slot per:spouse. Observing that in some cases the named entities were incorrectly classified we also generated more general patterns using POS tags only e.g.

```
<np> VB:bear <np>
```

These candidate patterns were used later in the selection process. It is also possible to identify an Expected Value Type for each slot from the training examples. For each of the slot queries we generated Expected Value Types from the associated candidate patterns. For example: per:spouse requires a person or a noun phrase as the slot-value, per:date_of birth requires a date, per:age requires a number and org:website requires a URL.

2.2 Passage Retrieval

The Passage Retrieval (PR) module identifies passages of text that are likely to contain a slot-value. Passages can be entire documents or segments of text. Documents are processed using the Stanford NER core pipeline and occurences of the target entity are annotated.

Terrier retrieves a set of documents in response to the target entity as a query. Once the list of documents are processed using the Stanford NER core pipeline and occurences of the target entity are annotated.

2.3 Slot-Value Selection

The final phase of the pipeline selects and extracts the segment of text that is most likely the relevant slot-value. Each returned slot-value must also contain the docid of the supporting document from which the slot-value was extracted. The Slot-Value selection module processes the candidate slot-bearing passages returned by the PR module and selects passages that are of the Expected Value Type, as identified in the Query Processing phase. Smaller passages which conform to the candidate patterns are then selected and added to a candidate answer set.

Similar slot-values which occur in the KB and in the candidate answer set are identified using Levenshtein Distance (Levenshtein, 1966), thereby removing redundant values and where values of the form "chief executive", "Chief Executive" and "Chief-Executive" would be equivalent for the list-valued slot per:title. The candidate answer set is then ranked according to frequency of occurrence, where the highest ranked slot-value occurs in the most documents. For single-value slots, the highest ranked answer in the candidate set is returned as the slot-value. For list-value slots there was no limitation placed on the number of candidate answers returned.

If no slot-value has been found a value of "NIL" is returned as the slot-value, that is, when the PR module fails to return any candidate slot-bearing passages or when the Expected Value Type is not identified.

Submitted run: IIRG1	
Recall	0.10110175
Precision	0.60700387
F1 Score	0.17333332

Table 1: Results of IIRG SF Run Submitted to TAC 2012

2.4 Results

We submitted one official run to the Slot Filling task. This run uses the candidate patterns derived from part-of-speech tags where the PR module returns sentences which contain the target entity.

The submitted run achieves very low coverage across the entire document collection, finding approximately 10% of the novel slot-values (see Table 1). We suspect that such a low recall score is a result of only considering sentences which contain the target entity without implementing any coreference resolution, etc. It is also difficult to identify geographic entities correctly when relying on POS tags only.

3 Mono-Lingual Entity Linking Task

The objective of the Entity Linking task is to associate each query entity with the relevant Knowledge Base entry. In the Entity Linking task participants

[2] http://nlp.stanford.edu/software

are given a query set consisting of 2229 target entities to process; these entities can be a person (PER), organisation (ORG) or geo-political entity (GPE). Each EL query consists of a name string or target entity and a supporting document id, for example,

```
<query id="EL_ENG_00010">
    <name>Burlington</name>
    <docid>eng-NG-31-126076-12043293</docid>
    <beg>433</beg>
    <end>442</end>
  </query>
```

Systems are required to return the ID of the Knowledge Base entry to which the name refers. If no KB entry exists, systems should return NIL. In addition, systems are required to cluster queries which refer to the same entity and are not present in the KB. NILs should be of the format "NILxxxx", where "xxxx" refers to a unique id given to a cluster of similar query entities. NIL clustering allows us to identify novel entities and will support the automatic creation of novel KB entities. This task can be further divided into two subtasks: an entity-linking task using the free text (wiki-text) from the KB pages associated with the knowledge base nodes and an entity-linking task without using this free text. The fact that the same entity can often be referred to by more than one query string and one entity name can refer to more than one entity makes Entity Linking quite a challenging task. We implemented a simple EL system which did not access the available wiki-text field.

3.1 Pre-Processing Phase

In the Pre-Processing phase, we processed the documents in the knowledge base into single documents to conform with standard TREC formats for ease of indexing. We then indexed the collection using Terrier 3.0 based on the fields wiki-title and wiki-text. Queries were normalised to reduce to lowercase and remove any unnecessary punctuation.

Entities can occur using different name variants (alternate names, aliases, abbreviations, nicknames etc.) and possibly docids. During this phase we also calculate the similarity between all of the query entities in order to cluster similar entities together for each query. We calculate a similarity score using Levenshtein Distance (Levenshtein, 1966). This will allow us to cluster similar entities together based on the threshold set for distance score. We have opted for distance scores of zero and one, allowing for the clustering of exact and inexact or fuzzy matches respectively.

3.2 Passage Retrieval

The Passage Retrieval (PR) module returns a list of node ids corresponding to KB entities in response to the name string of the EL query. This is field-based retrieval and runs on the wiki-title field.

3.3 Entity Node Selection

The Entity Node Selection phase applies shallow pattern-matching techniques to the list of wiki-titles returned by the PR module in order to select the node that has the closest match to the query entity. If no match is found, a "NIL" value is returned. All similar NIL nodes are assigned a unique NIL id of the form NIL00001.

3.4 Results

We submitted five official runs to the Entity Linking task. The first four runs were simple runs implemented to provide standard baselines for the system. IIRG1 and IIRG2 implemented fuzzy and exact matching of the KB node names, respectively, using the EL query name string and performed NIL clustering on similar queries where no KB entry was found. IIRG3 and IIRG4 implemented fuzzy and exact matching, respectively, using the EL query name string and performed NIL clustering on similar queries. IIRG5 implemented some query expansion techniques using our Slot Filling system. In this instance, the SF system retrieves slot-values for two slots, per:alternate_name and org:alternate_name. The alternate name strings found were then used as input into our search engine and NIL clustering is performed on EL queries that return no KB entry.

Results for the Entity Linking task are calculated based on B^3 + F1 scores computed over all evaluation queries. Table 2 shows how the system runs performed over the TAC 2012 EL evaluation queries using the B^3+ F1 score.

The highest and median B^3 + F1 score for all systems computed across 93 runs from 24 teams, is illustrated in Table 3. Assessors have also provided a breakdown of the performance of each submitted run over various subsets of "focus" queries. All five submitted runs performed below the median performance of 53.6%. Future runs will include the use of the entire document collection, additional slot-values to perform query expansion and will also incorporate the available wiki-text field.

The run that implemented some simple query expansion techniques (IIRG5) performs slightly better than the runs which do not implement this feature. Our results for both runs are very close to the median. For future runs we hope to incorporate the wiki-text and the source collection as a feature of our

query distance calculations in order to disambiguate query entities from KB entities where they exist.

Future runs will incorporate the use of wiki-text.

4 Conclusions

This year we participated in two KBP tasks, Regular English Slot Filling and Mono-Lingual Entity Linking. The Entity Linking results were not comparable to the median while, our slot filling system achieved a very high precision score, it achieved very low coverage across the document collection. The EL task itself is quite a demanding one given that entities can occur using many different variations and it remains a challenging task to improve our system to achieve a reasonable recall score for the slot filling task.

References

H. Ji, R. Grishman, and H.T. Dang. 2010. Overview of the TAC2011 Knowledge Base Population Track. In *Proceedings of the TAC 2011 Workshop*. Nist publication.

A. Levenshtein. 1966. *Binary codes capable of correcting deletions, insertions and reversals*, volume 10(8). Soviet Physics Doklady.

I. Ounis, Amati G., Plachouras V., B. He, Macdonald C., and Lioma C. 2006. Terrier - A High Performance and Scalable Information Retrieval Platform. In *ACM SIGIR06 Workshop on Open Source Information Retrieval (OSIR 2006)*.

Marius Pasca. 2003. *Open Domain Question Answering from Large Text Collections*. Center for the Study of Language and Information.

B^3 + F1 Scores for English entity-linking-no-wiki-text task								
RunID	All	in KB	not in KB	NW docs	WB docs	PER	ORG	GPE
IIRG1	0.226	0.000	0.480	0.222	0.230	0.249	0.327	0.061
IIRG2	0.249	0.000	0.529	0.243	0.255	0.256	0.378	0.063
IIRG3	0.243	0.066	0.434	0.249	0.226	0.280	0.309	0.101
IIRG4	0.262	0.066	0.476	0.267	0.246	0.286	0.353	0.102
IIRG5	0.387	0.254	0.511	0.406	0.348	0.551	0.415	0.100

Table 2: Results of IIRG EL Runs Submitted to TAC 2012

B^3 + F1 Scores for English entity-linking-no-wiki-text task								
	All	in KB	not in KB	NW docs	WB docs	PER	ORG	GPE
Highest B^3+ F1	0.730	0.687	0.847	0.782	0.646	0.810	0.717	0.694
Median B^3+ F1	0.522	0.407	0.648	0.568	0.432	0.691	0.482	0.365

Table 3: Results of EL Runs Submitted to TAC 2012