S	P	O	T
SyrianCivilWar	type	Event	[2011-03-15,now]
WorldWarI	type	Event	[1914-07-28,1918-11-11]
AngelaMerkel	type	GermanChancellor	[2005-11-20,now]
AngelaMerkel	type	EnvironmentMinister	[1994-11-17,1998-10-26]
AngelaMerkel	bornIn	Hamburg	[1954-07-17,1954-07-17]
FCBayernMunich	won	UEFAChampionsLeague	[2013-05-25,2013-05-25]
FCBayernMunich	won	UEFAChampionsLeague	[2001-05-23,2001-05-23]

Table 1: Examples of SPOT facts.

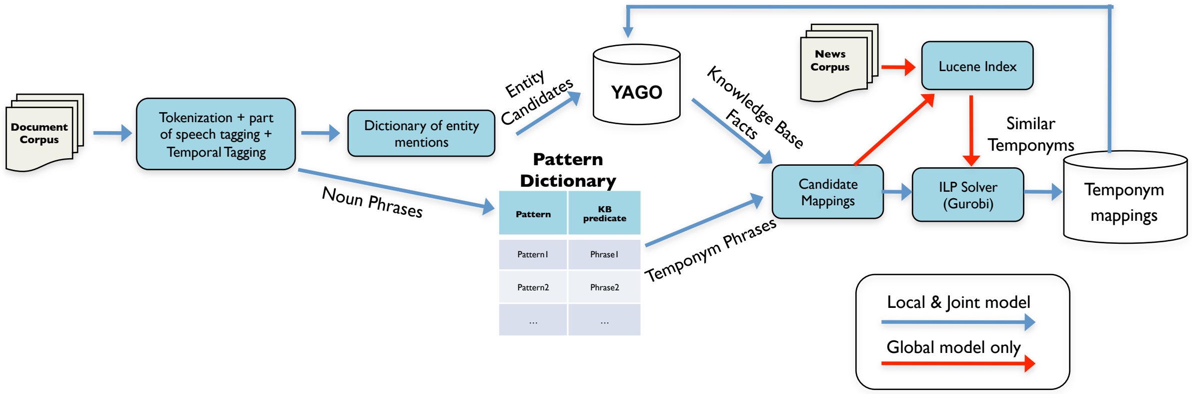

Figure 2: The processing pipeline for temponym detection and resolution.

ponym phrases. These two sets are later fed into integer linear programs (ILP's). The ILP's use different constraints and objective functions to jointly disambiguate mentions to entities and resolve the temponyms. Finally, the temponyms that are mapped to facts are added to the KB for knowledge enrichment. This is the flow for the local and joint ILP models we devised for temponym disambiguation. Additionally, in the global model, we exploit a news corpus to mine more relevant cues to enrich the context of temponym candidates and enhance the efficacy of temponym resolution using ILPs.

Inputs. The following inputs are used by our system.

- Text inputs. We use the Stanford NLP software (`nlp.stanford.edu/software/corenlp.shtml`) for tokenization and part of speech tagging in input documents.
- Knowledge base. We use the Yago knowledge base (`yago-knowledge.org`) to provide us with entities, facts about entities, semantic types of entities, and textual surface forms (alias names) of entities.
- Repository of relational patterns. Based on the PATTY tool [30], we created a dictionary of lexico-syntactic patterns with semantic type signatures for each of the KB relations. We specifically tailor this repository for events and temporal facts. The details of how this repository is constructed are given in Section 4.1.

Significant phrases. We detect several types of textual expressions:

- TempEx's. We identify TempEx's by using the HeidelTime tagger [36]. The TempEx's are later used to construct time histograms for the input documents, which are cues for temponym resolution.
- Mentions. We detect entity mentions by using the Stanford NER tool [15]. These are later mapped to canonical entities in the knowledge base during joint disambiguation of temponyms and entities. A bottleneck is that person, organization, and location entities are addressed by general purpose NER tools. For instance, the phrase "*The United States presidential election of 2016*" is not detected by the NER stage, although it is a named event. Our temponym detector compensates for this limitation, and extracts such mentions for the entities of type event.
- Temponyms. We explain how we detect temponym expressions in Section 4.

Candidate generation. In this step, our system generates entity candidates for mentions and event or fact candidates for temponyms to along with their time scopes. The candidate generation of facts is guided by the repository of relational patterns.

Outputs. Mentions of (non-event-type) entities are disambiguated to canonical entities, and temponyms are mapped to the named events or to facts in the KB. These two tasks are coupled and jointly solved by our ILP methods.

Note that significant phrases might overlap; the disambiguation stage of our system chooses a consistent subset of phrases mapped to targets in the KB. All resolved temponyms are added to the KB. Thus, we provide *knowledge enrichment* in two ways: i) disambiguating the phrases in the text and creating semantic markup, and ii) extending the KB with new "rdfs:label" triples that have a temponym phrase as subject, and an event or a fact as object.

4. TEMPONYM DETECTION

Temponyms typically appear as noun phrases in text. A noun phrase can generally refer to i) the name of an entity (e.g., "president Obama" or "the US president"), ii) a class (type) of entities (e.g., "US presidents" or "football clubs"), iii) a general concept (e.g., "the climate change" or "linear algebra"), iv) textual patterns with temporal scope (e.g., "Greek referendum" or "the FIFA final"), v) miscellaneous cases (e.g., idioms, quotes, etc.).

In order to gather noun phrases for case (iv) from a given input text, we use a small number of handcrafted regular expressions over word sequences and their part-of-speech