Named Entity Recognition

Document Understanding, session 8

Information Extraction

So far, we have focused mainly on ad-hoc web search. This usually starts from a user query and tries to find relevant documents.

Another possible approach is to construct a database of facts inferred from online text. This database can be used to enhance document understanding for better ranking and to answer questions more directly. This process is called **Information Extraction**.

The information panels beside search results are typically populated from these databases.

Image from bing.com; edited

Named Entity Recognition

In the IE subfield of **Named Entity Recognition** (NER), we use automated tools to identify clauses in text which correspond to particular people, places, organizations, etc.

Clauses are generally tagged with an entity type from a predefined list. Each entity type has its own contextual clues for identifying entities of that type.

For instance, times and dates often follow a few predictable formats. Peoples' names are often introduced in the surrounding text (e.g. "*spokesman* Tim Wagner").

NER Example and Tags

Citing high fuel prices, [ORG United Airlines] said [TIME Friday] it has increased fares by [MONEY $6] per round trip on flights to some cities also served by lower-cost carriers. [ORG American Airlines], a unit of [ORG AMR Corp.], immediately matched the move, spokesman [PERS Tim Wagner] said.

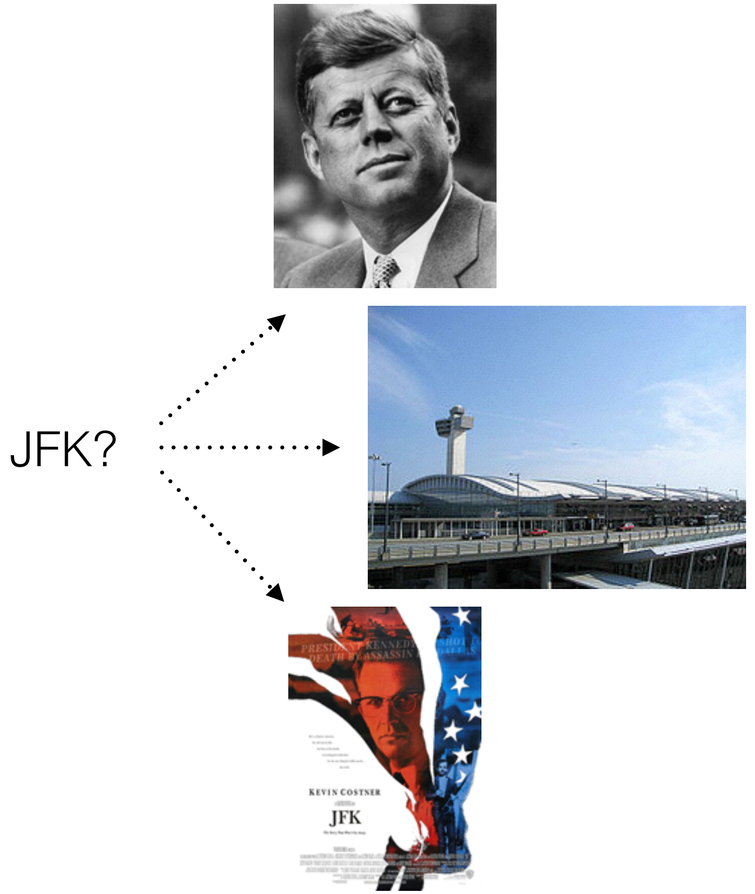
Tag	Entity	Example
PERS	People	Pres. Obama
ORG	Organization	Microsoft
LOC	Location	Adriatic Sea
GPE	Geo-political	Mumbai
FAC	Facility	Shea Stadium
VEH	Vehicles	Honda

Ambiguity in NER

NER systems often have to deal with several important types of ambiguity:

- **Reference resolution**: the same name can refer to different entities of the same type. For instance, JFK can refer to a former US president or his son.

- **Cross-type Confusion**: the identical entity mentions can refer to entities of different types. For instance, JFK also names an airport, several schools, bridges, etc.



Rule-based NER

Rule-based systems for NER are effective for certain entity classes.

Many of them use **lexicons**, which lists names, organizations, locations, etc.

Rules can also be crafted using regular expressions or other pattern matching tools. The rules may be built by hand, or with machine learning.

Entity Patterns

"<number> <word> street" for addresses

"<street address>, <city>" or "in <city>" to verify city names

"<street address>, <city>, <state>" to find new cities

"<title> <name>" to find new names

NER with Sequence Tagging

Sequence tagging is a common ML approach to NER.

Tokens are labeled as one of:

- `B`: Beginning of an entity
- `I`: Inside an entity
- `O`: Outside an entity

We train a Machine Learning model on a variety of text features to accomplish this. We'll see how to do this in the next session.

Word	Label	Tag
American	B	ORG
Airlines	I	ORG
a	O	–
unit	O	–
of	O	–
AMR	B	ORG
Corp.	I	ORG
immediately	O	–
matched	O	–
the	O	–
move	O	–
spokesman	O	–
Tim	B	PERS
Wagner	I	PERS
said	O	–

Features for Sequence Tagging

Feature Type	Explanation
Lexical Items	The token to be labeled
Stemmed Lexical Items	Stemmed version of the token
Shape	The orthographic pattern of the word (e.g. case)
Character Affixes	Character-level affixes of the target and surrounding words
Part of Speech	Part of speech of the word
Syntactic Chunk Labels	Base-phrase chunk label
Gazetteer or name list	Presence of the word in one or more named entity lists
Predictive Token(s)	Presence of predictive words in surrounding text
Bag of words/ngrams	Words and/or ngrams in the surrounding text

Word Shape

In English, the shape feature is one of the most predictive of entity names.

It is particularly useful for identifying businesses and products like Yahoo!, eBay, or iMac.

Shape is also a strong predictor of certain technical terms, such as gene names.

Shape	Example
Lower	cummings
Capitalized	Washington
All caps	IRA
Mixed case	eBay
Capitalized character with period	H.
Ends in digit	A9
Contains hyphen	H-P

NER Pipeline

A full production pipeline for NER will typically combine a few approaches:

1. First, use high-precision rules to tag unambiguous entities:
    - Use hand-tailored regular expressions, e.g. for dates and times.
    - Or write entity parsers for particular web sites, such as infoboxes on Wikipedia.
2. Search for substring matches of previously-detected names on the same page, using probabilistic string-matching metrics.
3. Consult application-specific name lists to identify likely name entity mentions from the given domain.
4. Apply **sequence tagging** using the tags from 1-3 as well as additional features, to find entities missed by the rule-based systems.

Wrapping Up

Named Entity Recognition is an important source of features for IR.

A very large fraction of queries contain named entities, so recognizing them as such and finding documents which mention the same entities is very important.

We may also want to treat the named entity as a single token, instead of as individual words (e.g. “New York Times).

Next, we’ll see how to perform NER using sequence tagging.