COM6115: Text Processing

Information Extraction: Relation Extraction

Chenghua Lin

Department of Computer Science
University of Sheffield

Overview of Lectures on IE

- Introduction to Information Extraction
  - ◇ Definition + Contrast with IR
  - ◇ Example Applications
  - ◇ Overview of Tasks and Approaches
  - ◇ Evaluation + Shared Task Challenges
  - ◇ A Brief History of IE
- Named Entity Recognition
  - ◇ Task
  - ◇ Approaches to NER
  - ◇ Entity Linking
- Relation Extraction
  - ◇ Task
  - ◇ Approaches: Knowledge Engineering; Supervised learning; Bootstrapping; Distant Supervision

Relation Extraction: Outline

- Relation Extraction Task
- Approaches to Relation Extraction
  - ◇ Knowledge-engineering approaches to NER
  - ◇ Supervised learning approaches to NER
  - ◇ Bootstrapping Approaches to NER
  - ◇ Distant Supervision Approaches to NER

Relation Extraction Task: Recap

- **Task**: given a text T and a set of relations $\mathbf{R}$, identify all assertions of relations from $\mathbf{R}$ in T, holding between entities identified in entity extraction.
- Note:
  - relations in $\mathbf{R}$ are usually binary
  - the entity types of arguments of relations in $\mathbf{R}$ are assumed to be a subset of those identifed in the entity extraction process
- May be divided into two subtasks:
  - **Relation detection**: find pairs of entities between which a relation holds
  - **Relation classification**: for pairs of entities between which a relation holds, determine what that relation is

Relation Extraction Task: Examples

- Examples
  - LOCATION_OF holding between
    - ORGANISATION and GEOPOLITICAL_LOCATION
    - medical INVESTIGATION and BODY_PART
    - GENE and CHROMOSOME_LOCATION
  - EMPLOYEE_OF holding between PERSON and ORGANISATION
  - PRODUCT_OF holding between ARTIFACT and ORGANISATION
  - IS_EXPOSED_TO holding between ORGANIZATION and RISK
  - IS_ASSOCIATED_WITH holding between DRUG and SIDE_EFFECT
  - INTERACTION holding between PROTEIN and PROTEIN

Relation Extraction Task: Challenges

Relation Extraction is challenging for several reasons:

- The same relation may be expressed in many different ways:
  - ◇ Canonical: $[\text{Microsoft}]_{ORG}$ is located in $[\text{Redmond}]_{LOC}$
  - ◇ Synonyms: $[\text{Microsoft}]_{ORG}$ is based/headquartered in $[\text{Redmond}]_{LOC}$
  - ◇ Syntactic variations:
    - $[\text{Microsoft}]_{ORG}$, the software giant and ..., is based in $[\text{Redmond}]_{LOC}$
    - $[\text{Redmond}]_{LOC}$-based $[\text{Microsoft}]_{ORG}$ ...
    - $[\text{Redmond}]_{LOC}$'s $[\text{Microsoft}]_{ORG}$ ...; $[\text{Microsoft}]_{ORG}$ of $[\text{Redmond}]_{LOC}$
    - $[\text{Redmond}]_{LOC}$ software giant $[\text{Microsoft}]_{ORG}$ ...

Relation Extraction Task: Challenges

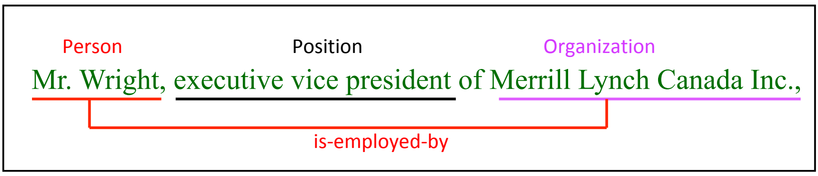
Relation Extraction is challenging for several reasons (cont):

- The information required may be spread across multiple sentences and discovering relations may depend upon following coreference links.
  Dirk Ruthless of MegaCorp made a stunning announcement today. In September he will be stepping down as Chief Executive Officer to spend more time with his pet piranhas.
  - To determine the corporate position of Dirk Ruthless we must correctly resolve the pronominal anaphor "he" in the second sentence with "Dirk Ruthless" in the first
- The information to be extracted may be implied by the text, rather than explicitly asserted, and extracting it may require inference
  - E.g. in the previous example we are not told explicitly that Dirk Ruthless **is** CEO of MegaCorp
  - To determine this requires knowing (*inter alia*) that stepping down from a position presupposes being in the position prior to stepping down

Relation Extraction: Outline

- Relation Extraction Task
- Approaches to Relation Extraction
  - Knowledge-engineering approaches
  - Supervised learning approaches
  - Bootstrapping Approaches
  - Distant Supervision Approaches

Knowledge Engineering Approaches

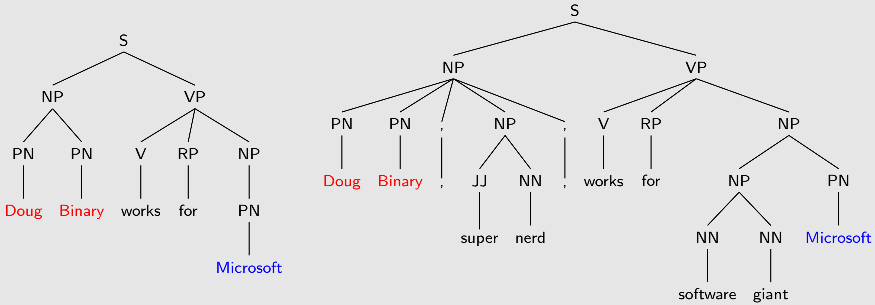


Such systems use manually authored rules and can be divided into

- "shallow" – systems engineered to the IE task, typically using pattern-action rules

  Pattern: `\$Person, $Position of $Organization"`
  Action: add-relation(is-employed-by($Person,$Organization))

Knowledge Engineering Approaches (cont)



- "deep" – linguistically inspired "language understanding" systems
  - ◇ typically parse input using broad coverage NL parser to identify key grammatical relations, like subject and object
  - ◇ use transduction rules to extract relations of interest from parser output
  - ◇ extraction rules over parser output allow a wider set of expressions to be captured than with regex's over words and NE tags alone
    - Example shows how multiple surface forms share underlying syntactic structure: here both have form SUBJECT = PER, OBJECT = ORG and VERB = *works for*

Knowledge Engineering Approaches (cont)

- Strengths
  - ◇ High precision
  - ◇ System behaviour is human-comprehensible
- Weaknesses
  - ◇ The writing of rules has no end
  - ◇ New rules needed for every new domain (pattern action rules for shallow approaches; transduction rules for deep approaches)

Relation Extraction: Outline

- Relation Extraction Task
- Approaches to Relation Extraction
  - Knowledge-engineering approaches
  - Supervised learning approaches
  - Bootstrapping Approaches
  - Distant Supervision Approaches

Supervised learning approaches

- First question to be asked: What is to be learned?
- Answer 1: rules that
  - Match to all and only relation bearing sentences
  - Capture substrings within the matched text that correspond to relation arguments
- Answer 2: binary classifier that when applied to a sentence containing instances of the entity types between which the relation holds
  - Returns 1 if the relation holds in this instance
  - Returns 0 if the relation does not hold in this instance

  As with NER can be divided into detection and classification stages:
  - Classifier 1 (binary) determines whether a given sentence expresses any of a set of relations of interest (relation detection)
  - Classifier 2 (multi-way) determines, for positive outputs from Classifier 1, which relation holds (relation classification)
- Rule learning approach popular in late 1990's/early 2000's; since then most work focusses on classifier approach – we'll look at the 2nd only

Supervised learning approaches: Classifier Learning

In classification approaches to relation extraction:

- Assume entities to be related already tagged
- Use any algorithm for learning binary classifiers to learn to distinguish instances (typically sentences) where
  - ◇ entities co-occur and relation holds (positive instances)
  - ◇ entities co-occur and relation does not hold (negative instances)
- Key issue: iwhat features do we use to represent the instances? Features used fall into 3 broad classes:
  - ◇ Features of the named entities
  - ◇ Features from the words in the text, usually words from 3 locations
    - words between the two NE candidate arguments
    - words in a fixed window to the left of the 1st candidate
    - words in a fixed window to the right of the 2nd candidate
  - ◇ Features about the entity pair within the sentence, e.g.
    - how far the entities are apart (in words or constituents)
    - whether other NE's occur between them
    - features from the syntactic structure of the sentence

Classifier Learning – Example

- Suppose we have the sentence

  [$_{ORG}$ American Airlines], a unit of [$_{ORG}$ AMR Corp.], immediately matched the move, spokesman [$_{PER}$ Tim Wagner] said.

  (Jurafsky and Martin, 2nd ed., p. 730)
- Then features extracted for this example when classifying the tuple: ⟨ American Airlines, Tim Wagner ⟩

Entity-based features	
Entity$_1$ type	ORG
Entity$_1$ head	*airlines*
Entity$_2$ type	PERS
Entity$_2$ head	*Wagner*
Concatenated types	ORGPERS
Word-based features	
Between-entity bag of words	{ *a, unit, of, AMR, Inc., immediately, matched, the, move, spokesman* }
Word(s) before Entity$_1$	NONE
Word(s) after Entity$_2$	*said*
Syntactic features	
Constituent path	$NP \uparrow NP \uparrow S \uparrow S \downarrow NP$
Base syntactic chunk path	$NP \rightarrow NP \rightarrow PP \rightarrow NP \rightarrow VP \rightarrow NP \rightarrow NP$
Typed-dependency path	$Airlines \leftarrow_{subj} matched \leftarrow_{comp} said \rightarrow_{subj} Wagner$

(Jurafsky and Martin, 2nd ed., p. 738)

Supervised learning approaches

- Strengths:
  - No need to write extensive/complex rule sets for each domain
  - Same system straightforwardly adapts to any new domain, provided training data is supplied
- Weaknesses:
  - Quality of relation extraction dependent on quality and quantity of training data, which can be difficult and time consuming to generate
  - Developing feature extractors can be difficult and they may be noisy (e.g. parsers) reducing overall performance

Relation Extraction: Outline

- Relation Extraction Task
- Approaches to Relation Extraction
  - Knowledge-engineering approaches
  - Supervised learning approaches
  - Bootstrapping Approaches
  - Distant Supervision Approaches

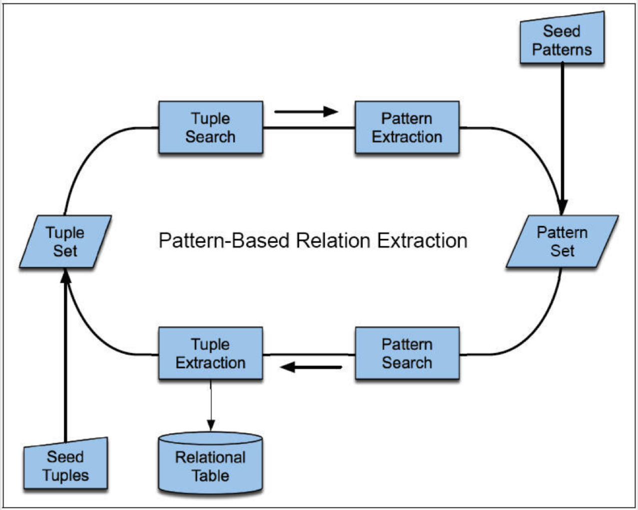
Bootstrapping approaches

- Motivation: reduce number of manually labelled examples needed to build a system
- Key idea: start with a document collection $\mathcal{D}$ and either :
  1. set of trusted tuples **T** (e.g. pairs of entities known to stand in the relation of interest)
  2. set of trusted patterns **P** (i.e. patterns known to extract pairs of entities in the given relation with high accuracy)

  Then, if
  1. then find tuples from **T** in sentences **S** in $\mathcal{D}$, extract patterns from context of sentences in **S**, add patterns to **P** and then use **P** to find new tuples in $\mathcal{D}$ and add to **T**; repeat until convergence
  2. then match patterns from **P** in sentences **S** in $\mathcal{D}$, extract tuples from pattern matches in sentences in **S**, add tuples to **T** and then use tuples in **T** to find new patterns in $\mathcal{D}$ and add to **P**; repeat until convergence

Bootstrapping approaches

- Diagrammatically, this can be shown as follows:



(Jurafsky and Martin, 2nd ed., p. 740)

Bootstrapping approaches – DIPRE

- One early system employing this approach was DIPRE – Dual Iterative Pattern Relation Expansion – proposed by Sergie Brin (1999)
- Aim: to extract useful relational tuples from the Web, of the form (PERSON, BOOK_TITLE) – e.g. (Leo Tolstoy, War and Peace)
- Method:
  - ◇ Exploit "duality of patterns and relations"
    - Good tuples help find good patterns
    - Good patterns help find good tuples
  - ◇ Starting with user-supplied tuples, iteratively
    - Use these tuples to find patterns
    - Use the patterns to find more tuples

Bootstrapping approaches – DIPRE (cont)

The main loop in DIPRE is as follows:

1. $R' \leftarrow$ Sample
   R' is an approximation of the target relation (a set of tuples);
   Sample is a small user-supplied sample (e.g. 5 author-title pairs)
2. $O \leftarrow$ FindOccurrences(R',D)
   Find all occurrences of tuples of R' in D
3. $P \leftarrow$ GenPatterns(O)
   Generate patterns based on the set of occurrences – want patterns to have low error rate and, ideally, high coverage (can compensate for latter with large database (e.g. the Web)
4. $R' \leftarrow M_D(P)$
   Update R' with the set of tuples from documents in D that matched by patterns in P
5. If R' is large enough return ; else go to 2.

Bootstrapping approaches – DIPRE (cont)

Brin reports an experiment with finding (author,title) pairs on the web

- Patterns are defined as 5-tuples:
  $(order, urlprefix, prefix, middle, suffix)$
  - ◇ If order is true an (author, title) pair matches the pattern if there is a document in the collection (web)
    - whose URL matches urlprefix*
    - which contains text which matches the RE *prefix, author, middle, title, suffix*
    - more detailed RE's are given for author and title
  - ◇ If order is false title and author are switched
- Occurrences are defined as 7-tuples:
  $(author, title, order, url, prefix, middle, suffix)$
  - ◇ Order records the order the author and title occurred in the text
  - ◇ URL is the URL of the document the occurrence was found in
  - ◇ Prefix is the m characters (in tests m=10) preceding the author (or title)
  - ◇ Middle is text between author and title
  - ◇ Suffix is m characters following title (or author)

Bootstrapping approaches – DIPRE (cont)

- An algorithm for generating a pattern given a set of occurrences is described
  - ◇ Algorithm inisists *order* and *middle* of all occurrences is the same – they form part of the generated pattern
  - ◇ Additionally pattern contains
    - longest matching prefix of the *url* of all the occurrences
    - longest matching suffix of the *prefix* of all the occurrences
    - longest matching prefix of the *suffix* of all the occurrences
    - See Brin (1999) for details
- Patterns are assessed for specificity and rejected if their specificity is too low, i.e. if they are too general
  - ◇ Specificity of a pattern is defined in terms of the product of the lengths of the pattern's *middle*, *urlprefix*, *prefix* and *suffix*
  - ◇ For a pattern p, $\textit{specificity}(p) \times n$ must exceed some threshold t, where n is the number of books with occurrences supporting the pattern p

Bootstrapping approaches – DIPRE Experiment

- Used 24 million web pages + 5 seed tuples

Author	Title
Isaac Asimov	The Robots of Dawn
David Brin	Startide Rising
James Gleick	James Gleick
Charles Dickens	Great Expectations
William Shakespeare	The Comedy of Errors

- Yielded 199 occurrences and generated 3 patterns
- These 3 patterns produced 4047 unique (author, title) pairs
- A search over 5 million web pages yielded 3972 occurrences of these books – stopped at this point due to computational constraints
- These occurrences produced 105 patterns which in turn produced 9369 (author, title) pairs – some had bad authors and were rejected
- Using these working pairs in a final iteration resulted in 9988 occurrences, then 346 patterns and then 15257 unique books
- Manual inspection of 20 from the final list showed 19 were bonafide books and 1 was an article

Bootstrapping approaches

- Strengths:
  - Need for manually labelled training data is eliminated
- Weaknesses:
  - Can suffer from semantic drift – when an erroneous pattern introduces erroneous tuples, which in turn lead to erroneous patterns
    - Introduction of confidence measures for patterns and tuples can mitigate against this problem to some extent
  - Works well when significant redudancy in assertion of specific tuples an in use of specific patterns to express a relation
    - True for some domains/relations and text collections, not for others
  - Issues when multiple relations hold between the same pair of entities
    - e.g. suppose someone is born, is educated and dies in the same location, then a sentence containing occurrences of person name and location name could be expressing any of three relations

Relation Extraction: Outline

- Relation Extraction Task
- Approaches to Relation Extraction
  - ◇ Knowledge-engineering approaches
  - ◇ Supervised learning approaches
  - ◇ Bootstrapping Approaches
  - ◇ Distant Supervision Approaches

Distant Supervision Approaches

- As with bootstrapping approaches, distant supervision approaches aim to reduce/eliminate the need for manually labelled training data
- Key idea:
  - ◇ Suppose we have a large document collection $\mathcal{D}$ plus a structured data source (e.g. a database) $\mathcal{R}$ that contains
    - many instances of a relation of interest in, e.g., a relational table
    - optionally, for each relation instance a link to a document in $\mathcal{D}$ providing evidence for the relation
  - ◇ Then we can
    - search for sentences in $\mathcal{D}$ containing the entity pairs that occur in relation instances (tuples) in $\mathcal{R}$
    - label these sentences as positive occurrences of the relation instance
    - use the labelled sentences as training data to train a standard supervised relation extractor

Distant Supervision approaches (cont)

- One well-known approach using distant supervision is described by Mintz et al. (2009)
- Mintz et al. use Freebase as their structured data source

Relation name	Size	Example
/people/person/nationality	281,107	John Dugard, South Africa
/location/location/contains	253,223	Belgium, Nijlen
/people/person/profession	208,888	Dusa McDuff, Mathematician
/people/person/place_of_birth	105,799	Edwin Hubble, Marshfield
/dining/restaurant/cuisine	86,213	MacAyo's Mexican Kitchen, Mexican
/business/business_chain/location	66,529	Apple Inc., Apple Inc., South Park, NC
/biology/organism_classification_rank	42,806	Scorpaeniformes, Order
/film/film/genre	40,658	Where the Sidewalk Ends, Film noir
/film/film/language	31,103	Enter the Phoenix, Cantonese
/biology/organism_higher_classification	30,052	Calopteryx, Calopterygidae
/film/film/country	27,217	Turtle Diary, United States
/film/writer/film	23,856	Irving Shulman, Rebel Without a Cause
/film/director/film	23,539	Michael Mann, Collateral
/film/producer/film	22,079	Diane Eskenazi, Aladdin
/people/deceased_person/place_of_death	18,814	John W. Kern, Asheville
/music/artist/origin	18,619	The Octopus Project, Austin
/people/person/religion	17,582	Joseph Chartrand, Catholicism
/book/author/works_written	17,278	Paul Auster, Travels in the Scriptorium
/soccer/football_position/players	17,244	Midfielder, Chen Tao
/people/deceased_person/cause_of_death	16,709	Richard Daintree, Tuberculosis
/book/book/genre	16,431	Pony Soldiers, Science fiction
/film/film/music	14,070	Stavisky, Stephen Sondheim
/business/company/industry	13,805	ATS Medical, Health care

Table 2: The 23 largest Freebase relations we use, with their size and an instance of each relation.

Source: Mintz et al. (2009)

Distant Supervision approaches – Mintz et al. (cont)

- Freebase was a free on-line database of structured semantic data
  - ◇ data derived from, e.g. Wikipedia infoboxes + other open access sources
  - ◇ after filtering Mintz et al. derived 1.8 million instances of 102 relations connecting 940,000 entities
  - ◇ Freebase no longer available – bought by Google and now forms part of Google Knowledge Graph (partly free, partly paid access)
  - ◇ Similar current sources are DBPedia and WikiData
- Mintz et al. use a dump of the text from Wikipedia as their document collection
  - ◇ dump consists of $\approx$ 1.8 million articles, averaging 14.3 sentences/article
  - ◇ used 800,000 articles for training and 400,000 for testing

Distant Supervision approaches – Mintz et al. (cont)

- Distant supervision assumption: if two entities participate in a relation, any sentence that contains those two entities might express that relation.
  - So, tag all sentences containing the two entity mentions as mentions of the relation
- Same relation may be expressed in different ways in different sentences. E.g.

  [Steven Spielberg]'s film [Saving Private Ryan] is loosely based on the brothers' story.

  Allison co-produced the Academy Award- winning [Saving Private Ryan], directed by [Steven Spielberg]...

  - So, combine features from multiple mentions to get a richer feature vector
  - Use multiclass logistic regression as a learning framework
  - At test time features are combined from all occurrences of a given entity pair in the test data and the most likely relation (or none) is assigned

Distant Supervision approaches – Mintz et al. (cont)

- Also need negative instances – an'unrelated' relation!
  - ◇ to get these, randomly select entity pairs that do not appear in any Freebase relation and extract features for them
  - ◇ Could be related – i.e. wrongly omitted from Freebase – but effect of these rare occurrences should be low
- Mintz et al. evaluate their approach
  - ◇ humans evaluate highest ranked 100 and 1000 results per relation for 10 relations
  - ◇ average precision for best feature combinations just under 70% (69% for top 10; 68% for top 1000)
  - ◇ these results are competitive for knowledge engineering and "normal" supervised learning systems, which struggle to get over 75% on similar tasks

Distant Supervision approaches: Strengths and Weaknesses

- Strengths:
  - Need for manually labelled training data is eliminated
  - Can very rapidly get extractors for a wide range of relations
- Weaknesses:
  - Precision still lags behind best knowledge-engineered/directly supervised learning approaches
  - Only works if a good supply of structured data is available for the relation(s) of interest

Conclusion

- Relation extraction aims to detect and classify all mentions of a given set of relations holding between specified entity types within a given text
- Relation extraction is a core IE technology that is stubbornly difficult, due to the highly variable ways relations can be expressed in natural language
- Techniques used have included:
  - Knowledge engineering approaches
  - Supervised learning approaches
  - Bootstrapping Approaches
  - Distant Supervision Approaches
- Open challenges include:
  - improving precision and recall
  - handling: relations expressed over > 1 sentences; textual entailment
  - improving bootstrapping techniques so as to minimise "semantic drift"
  - developing relation extractors for languages other than English

References

- Agichtein, E. and L. Gravano. Snowball: extracting relations from large plain-text collections. In Proceedings of the 5th ACM Conference on Digital Libraries , 85-94, 2000.
- Brin, S. Extracting patterns and relations from the World Wide Web. In WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98 , 172–183, 1998.
- Jurafsky, D and Martin, J. H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. 2nd ed. Pearson Inc. 2009. See Chapter 22.2 "Relation Detection and Classification".
- Mintz, Mike, Steven Bills, Rion Snow and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2 , 1003–1011, 2009.