COMP4650/6490 Document Analysis

Semantics & Coreference Resolution

ANU School of Computing

Administrative matters

- Assignment 3
  - Due: 5pm Thursday 12 October
  - Extension application: 24 hours before due date + supporting documents
- Assignment 2
  - Results will be released later this week
- Final exam:
  - 2pm - 4:15pm Monday 6 November
  - CSIT & Hanna Neumann Lab rooms
  - Centrally invigilated by the University
- Drop-in sessions
  - 1pm - 2pm Friday 20 October
  - 1pm - 3pm Thursday 26 October
  - 1pm - 2pm Friday 27 October
  - Room 3.41, Level 3, Hanna Neumann Building #145

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Semantics

What is Semantics

- **Semantics** is branch of linguistics and logic concerned with meaning
- Modelling semantics is the **holy grail of NLP** and a central question in Artificial Intelligence
  - Building a robot that can follow natural language instructions to execute tasks
  - Answering questions, such as *where is the nearest coffee shop*?
  - Translating a sentence from one language into another, while preserving the underlying meaning
  - Fact-checking an article by searching the web for contradictory evidence
  - Logic-checking an argument by identifying contradictions, ambiguity, and unsupported assertions

Semantics

What is Semantics

- Where is meaning?
  - My brain
  - Your brain
  - Words
  - Sentences
  - Body language
- Semantic theories explain how to linguistically represent meaning:
  - Logical semantics
  - Lexical semantics

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Logical Semantics

Meaning Representation

- In logical semantics, to semantically analyse a sentence is to convert it into a **meaning representation**
- A desired meaning representation that:
  - is **unambiguous**: only one possible interpretation
  - provides a way to **link language to external knowledge**
  - support computational **inference**
  - **expressive** enough

Logical Semantics

Unambiguous Denotations

- For example, code 5 + 3 outputs 8, also $(4 \times 4) - (3 \times 3) + 1$ outputs 8. These outputs are known as **denotations**.

$$[[5 + 3]] = [[(4 \times 4) - (3 \times 3) + 1]] = [[((8))]] = 8$$

- The denotations are determined by the meaning of
  - constants, e.g. 1, 3, 4, 5, 8
  - relations, e.g. $+$, $\times$, $(,)$
- What is the meaning of **double**?
  - $[[\text{double}(4)]] = 8$ or $[[\text{double}(4)]] = 44$?
  - It is defined in a world model $\mathscr{M}$.
  - $[[\text{double}(4)]]_{\mathscr{M}} = \{(0,0), (1,2), (2,4), \ldots\}$, thus $[[\text{double}(4)]] = 8$.
  - Then, denotation of string x in world model $\mathscr{M}$ can be computed unambiguously.

External Knowledge & Inference

- Connecting language to **external knowledge, observations and actions**

  The capital of Australia → knowledge base of geographical facts → Canberra

- **Inference** support is to automatically deduced new facts from premises, e.g. first-order-logic.

  If anyone is making noise, then Max can't sleep.
  Abigail is making noise.

  Inference: *Max can't sleep.*

- **How**?
  - Apply generalised inference rules like **modus ponens** ($P \rightarrow Q,$ if $P,$ then Q).
  - By repeatedly applying such inference rules to a knowledge base of facts it's possible to infer new knowledge and produce proofs.
- **Algorithms**: backward chaining, e.g. in prolog (logic programming language)

Logical Semantics

Semantic Parsing

- In NLP, semantic parsing is the transformation of sentences to a meaning representation, e.g. a logical formula
- The logic formalism is usually **lambda calculus** (an extension of first-order-logic)
  - *Alex likes Sam*
    $\rightarrow (\lambda x\,.\,\text{LIKES}(x, \text{SAM}))\text{@ALEX}$
    $\rightarrow \text{LIKES(ALEX, SAM)}$
- Traditionally, semantic parsing analysis was based on syntax structures
- Nowadays, semantic parsing is modelled as a Seq2Seq problem using deep learning (like machine translation)
- Logical formulas are hierarchical, thus Seq2Tree algorithms are also used

Logical Semantics

Semantic Parsing Datasets

- GEO
  - **Input** (question):
    Which is the longest river in USA?
  - **Output** (lambda formula):
    _answer(A, _longest(A, (_river(A), _loc(A, B), _const(B, _countryid(USA)))))
- ATIS
  - **Input** (question):
    Show me a flight from ci0 to ci1 tomorrow
  - **Output** (lambda formula):
    (lambda $0 e(and (flight $0) (from $0 ci0) (to $0 ci1) (tomorrow $0)))
- WikiSQL
  - **Input** (question):
    How many engine types did Val Musetti use?
  - **Output** (SQL query)
    SELECT COUNT Engine WHERE Driver = Val Musetti

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Predicate-argument Semantics

- Predicate-argument semantics is considered a **light semantic representation**
- A predicate is seen as a property that a subject has or is characterised by.
  - Verb-only predicate, e.g. *She dances*.
  - Verb-plus-direct-object predicate, e.g. *Ben reads the book*.
- Predicates have arguments (required / optional)
  - (arg1: someone) **dance**
  - (arg1: someone) **read** (arg2: something)
  - (arg1: someone) **rent** (arg2: something), (arg3: x money), (arg4: n time)
    e.g. *Annie rents an apartment for $200 per week.*

Predicate-argument Semantics

- Predicates participants or arguments are **constrained**
- Semantic roles
  - *John_(AGENT) dance salsa*.
  - *John_(EXPERIENCER) has a headache*.
- Selection restrictions
  - ~~*John rent a dance*~~
- How can we model this in NLP?
  - **PropBank**: sentences annotated with semantic roles
  - **FrameNet**: a hierarchical database of events (e.g. *act of teaching*) and arguments to these events (e.g. *teacher*, *student*, *subject being taught*)
  - **VerbNet**: a hierarchical verb lexicon with verbs and their arguments organised as thematic roles (e.g. *AGENT*, *RECIPIENT*, …)
- Using these resources, we can train machine learning models to tag predicates and arguments.

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Lexical Semantics

What is Lexical Semantics

- Lexical semantics is the linguistic study of word meaning
- Key questions:
  - What is the meaning of words?
    - Most words have more than one sense
  - How are the meanings of different words related?
    - Specific relations between senses
      e.g. ***vehicle*** *is more general than* ***car***
    - Semantic fields
      e.g. ***travel*** *is related to* ***flight***

Lexical Semantics

Terminology

- **Word sense**: a discrete representation of one aspect of the meaning of a word.
  For example, *bank*
  - A financial institution:
    CommBank
  - A particular branch of a financial institution:
    CommBank in Civic
  - The bank of a river:
    The bank of the Murrumbidgee
  - A ‘repository’:
    Blood bank
- In this example, *bank* has four senses

Lexical Semantics

Terminology

- **Homonymy**: coincidentally share an orthographic form. e.g. *bank*
  - *The bank took my deposit.* (financial institution)
  - *The bank was grassy.* (sloping mound)
- **Polysemy**: two senses are semantically related. e.g. *solution*
  - *Work out the solution in your head*.
  - *Heat the solution to 75° Celsius*.
- **Homophone**: same pronunciation, but different spellings. e.g., *wood / would, to / two*
- **Homograph**: same orthographic form, but different pronunciation (this is a problem in speech synthesis). e.g. *bass*
  - *I like to play the bass* (a musical instrument – bass guitar)
  - *Fresh bass is tasty* (a fish)

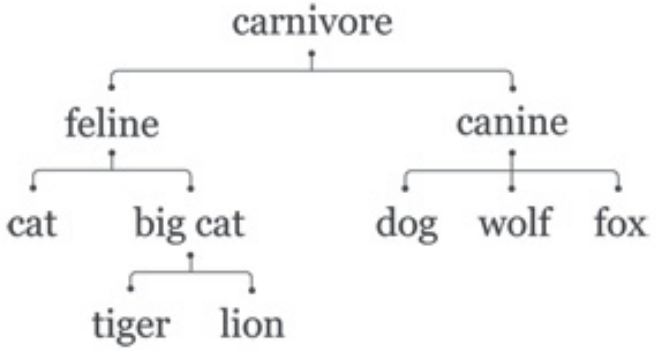
Lexical Semantics

Terminology

- **Synonyms**: two word lemmas are identical or nearly identical. e.g.
  - *couch / sofa, car / automobile*
- **Antonyms**: two word lemmas with opposite meaning
  - *long / short, big / little, rise / fall, in / out*
- **Hyponyms**: one sense is an hyponym of another sense if the first sense is more specific. e.g.
  - *car* is a hyponym of *vehicle*
  - *dog* is a hyponym of *animal*
  - *mango* is a hyponym of *fruit*
- **Hypernyms**: one sense is an hypernym of another sense if the first sense is its class
  - *vehicle* is a hypernym of *car*
  - *animal* is a hypernym of *dog*
  - *brie* is a hypernym of *cheese*

Datasets: WordNet

- A database of lexical (ontological) relations
  - ~120k nouns, ~115k verbs, ~20k adjectives, and ~5k adverbs
- Groups words into sets of (near-)synonyms called *synsets*, provides short definitions and usage examples
  - synset: the set of near-synonyms for a sense
  - ~80k noun synsets, ~15k verb synsets, ~20k adj. synsets, ~5k adv. synsets
- Hand constructed!
  - English https://wordnet.princeton.edu/
  - In many languages http://globalwordnet.org/resources/wordnets-in-the-world/



Lexical Semantics

Datasets: DBpedia & Wikidata

- DBpedia: another huge lexical graph
  - It use RDF triplets to encode relations between entities
  - RDF triplet:
    subj(Golden Gate Park) pred(location) obj(San Francisco)
  - ~900 million RDF triplets
  - Extracted from Wikipedia info-boxes
  - https://www.dbpedia.org/

- Wikidata: multilingual knowledge graph
  - well-known people, places, and things
  - ~12.5 billion triples
  - Database of facts, anybody can edit
  - https://www.wikidata.org/

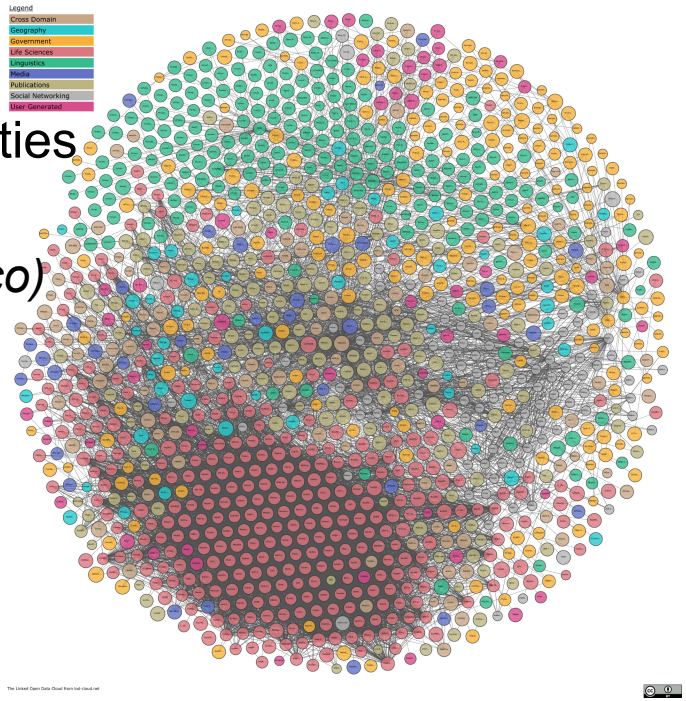


Diagram: Many open data sources (e.g. DBpedia, WordNet) are linked. This diagram shows the linkages between different sources.
https://lod-cloud.net/

Lexical Semantics

Distributional Hypothesis

- **How to learn representations for word meanings from unlabelled data**?
- This idea is based on the theoretical principle of the **distributional hypothesis**:
  You shall know a word by the company it keep (Firth, 1957).
- For example, the word "*chicha*" is not in my training data, but I know "*chicha*" is used in several contexts:

  (1) *A bottle of _____ is on the table.*

  (2) *Everybody likes _____.*

  (3) *Don't have _____ before you drive.*

  (4) *We make _____ out of corn*.

	(1)	(2)	(3)	(4)
chicha	1	1	1	1
loud	0	0	0	0
wine	1	1	1	0
tortilla	0	1	0	1
cook	0	0	0	0

Distributional statistics distilled from a dataset

- These vectors are word representations
- According to the distributional hypothesis, vector similarity implies semantic similarity
- Thus, *chicha* is similar to *wine*, other terms are fairly similar: *tortilla*, and others not similar at all: *cook*, *loud*

Lexical Semantics

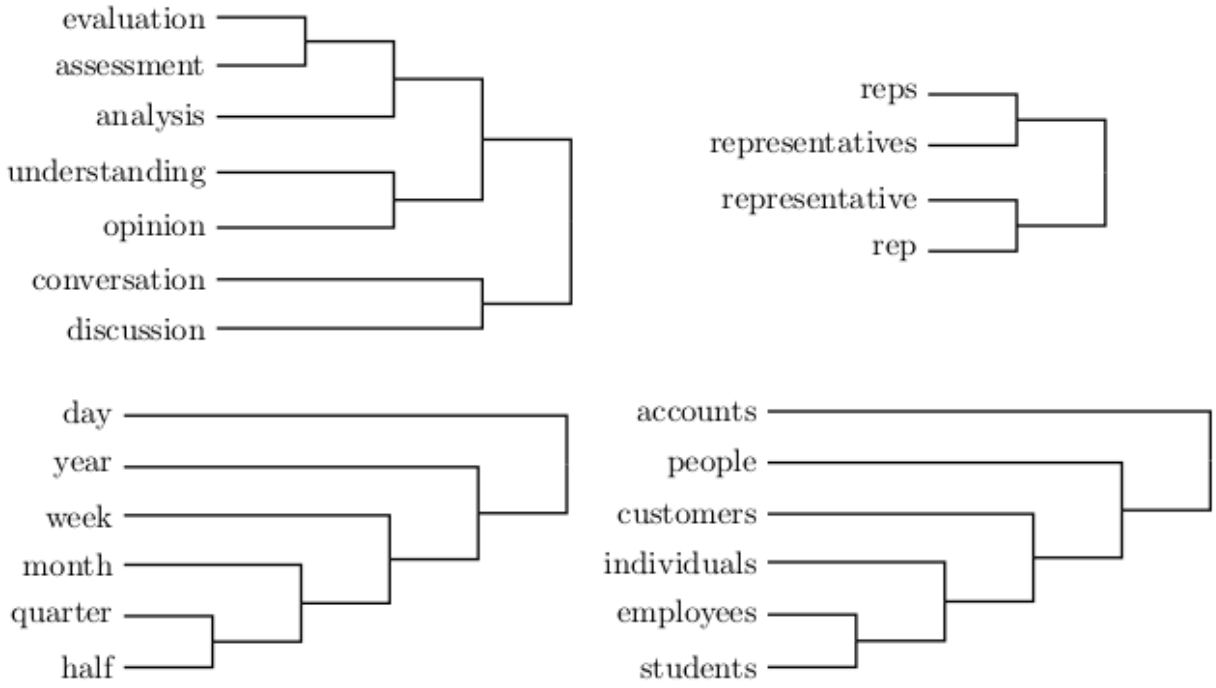
Distributional Hypothesis

- **Distributional** semantics are computed from context statistics (e.g. Brown clusters)
- **Distributed** semantics represent meaning by numerical vectors, rather than symbolic structures (e.g. word2vec, which is also distributional)

Lexical Semantics

Brown clustering

- **Brown clustering** induces hierarchical representations using cluster mergers that optimise an objective defined on word occurrence counts.
  (no distributed vectors used)
- Some Learning algorithms like CRF and Perceptron perform better with **discrete feature vectors**.
- Discrete representations can be distilled from word vectors by clustering.

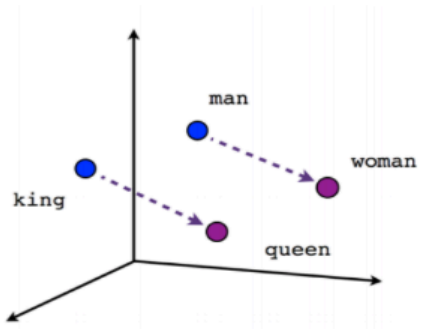


Subtrees produced by bottom-up Brown clustering on news texts (Miller et al. 2004)

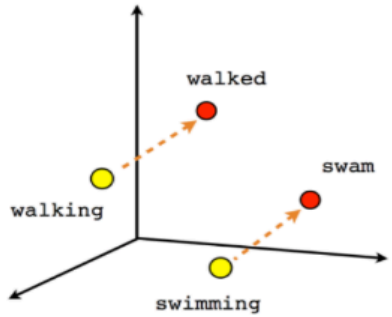
Lexical Semantics

Embeddings

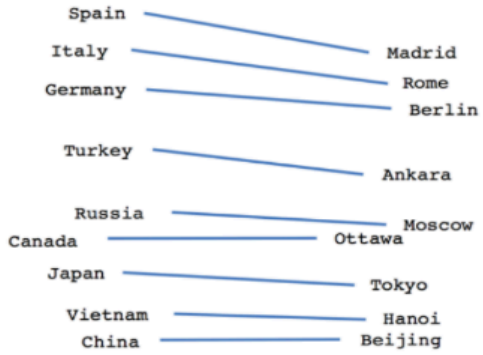
- **Word embedding**: Ensure words with similar context occupy close spatial positions in multidimensional space, as represented by the vector representations
- **Character embedding**: same idea, but encode at the character level to overcome unseen words (out-of-vocabulary)
- **Sentence embedding**: learn representations for sentences



Male-Female



Verb tense



Country-Capital

Source: https://www.tensorflow.org/tutorials/representation/word2vec

Australian National University

Outline

- What is semantics
- Logical semantics
- Predicate-argument semantics
- Lexical semantics
- Coreference resolution

Coreference Resolution

- In NLP we can model semantics by looking at the words and the logical structures underlying sentences.
- However, we still need to model other aspects of language such as references and grammar.
- Some of these aspects are easy for humans, but very hard for machines.

Coreference Resolution

What is Coreference Resolution

- Coreference resolution aims to solve referential ambiguity, e.g.

*The **trucks** shall treat the **roads** before **they** freeze*.

*(**Apple Inc**) (**Chief Executive Tim Cook**) has jetted into (**China**) for talks with government officials as (**he**) seeks to clear up a pile of problems in (**the firm's**) biggest growth market ... (**Cook**) is on (**his**) first trip to the (**country**) since taking over*.

Coreference Resolution

Terminology

Tim Cook *has jetted into* ***China*** *for talks with government officials as* ***he*** *seeks to clear up a pile of problems in* ***the firm's*** *…*

- **Mentions**: text spans that mention an entity.
  e.g. *Tim Cook, China, he, the firm*
- **Coreferent**: text spans that refer to the same entity.
  e.g. *Tim Cook, he*
- **Antecedents** of a mention are all coreferent mentions earlier in the text.
  e.g. the antecedent of *he* is *Tim Cook*
- Grouping text spans which refer to the same entity is typically called **coreference resolution**.

Coreference Resolution

Referring Expression: Pronouns

*Tim Cook has jetted in for talks with officials as [**he**] seeks to clear up a pile of problems.*

- Pronouns
  - Search for candidate antecedents: any noun phrase is the proceeding text is a candidate
  - Match against hard agreement constraints

he: singular, masculine, animate, third person

Tim Cook: singular, masculine, animate, third person

officials: plural, animate, third person

talks: plural, inanimate, third person

- Select using heuristics
  - Recency – close in the sentence
  - Subject over objects – nouns in the subject position (in a dependency parse) are more likely to be coreferent

Coreference Resolution

Referring Expression: Proper Nouns

*Apple Inc Chief Executive [**Tim Cook**] has jetted into China ...*
*[**Cook**] is on his first business trip to the country…*

- Proper nouns often corefers with another proper nouns
- Strategies (easier to solve):
  - Match the syntactic head words of the reference with the referent
  - In machine learning, a solution is to include a range of matching features: exact match, head match, and string inclusion.
  - Gazetteers of acronyms (*Australian National University / ANU*), and other aliases (*Queensland Technology University / Queensland Tech*)

Coreference Resolution

Referring Expression: Nominals

*The firm [**Apple Inc**]; the firm's biggest growth market [**China**]; and the country [**China**]*

- Nominals are noun phrases that are not pronouns nor proper nouns.
- **Hard to solve, as it requires world knowledge**:
  - *Apple Inc.* is a *firm*
  - *China* is a *growth market*

Algorithms for Coreference Resolution

Modelled as structured prediction problem with two tasks:

1. Identifying spans of text mentioning entities:
   - Get noun phrases from a sentence structure (e.g. using constituency parsing),
   - and filter using simple rules (e.g. remove numeric entities, remove nested noun phrases).
2. Clustering those mentions:
   - Mentioned-based models: supervised learning or ranking
   - Entity-based models: clustering

Algorithms for Coreference Resolution

Clustering Mentions: Classification (mentioned-based models)

- **Mention-pair models**
  - Binary label y_{ij} is assigned to each pair of mentions (i, j), $i < j$
  - If i and j corefer, then $y_{ij} = 1$, otherwise $y_{ij} = 0$
  - Use any off-the-shelf classifier to solve this binary classification problem.
  - Also need to use heuristics to construct coherent groups for each entity.

Algorithms for Coreference Resolution

Clustering Mentions: Ranking (mentioned-based models)

- A classifier learns to identify a single antecedent.
  - For each referring expression i,

    $\hat{a}_i = \text{argmax}_{a \in \{\epsilon, 1, 2, \ldots, i-1\}} \psi_M(a, i)$

    where $\psi_M(a, i)$ is a score for the mention pair (a, i)
  - If $a = \epsilon$, then mention i does not refer to any previously introduced entity.
- Mention ranking is like the mention-pair model, but all candidates are considered simultaneously, and at most one antecedent is selected.
- As a learning problem, ranking can be trained using the same objectives as in discriminative classification.
  - For each mention i, we can define an antecedent a_i^* (ground truth), and an associated loss, e.g. hinge loss or negative log-likelihood.

Algorithms for Coreference Resolution

- Mention embedding
  - Entity mentions can be embedded into a vector space, providing the base layer for neural networks that score coreference decisions (Wiseman et al., 2015)
- Constructing the mention embedding
  - One approach is inspired in embedding multi-word expressions
  - Run a bidirectional LSTM over the entire text, obtaining hidden states from the left-to-right and right-to-left passes
  - Each candidate mention span (s, t) is then represented by the vertical concatenation of four vectors:

$$\boldsymbol{u}^{(s,t)} = \left[\boldsymbol{u}^{(s,t)}_{\text{first}};\ \boldsymbol{u}^{(s,t)}_{\text{last}};\ \boldsymbol{u}^{(s,t)}_{\text{head}};\ \boldsymbol{\phi}^{(s,t)}\right]$$

where $\boldsymbol{u}^{(s,t)}_{\text{first}} = \boldsymbol{h}_{s+1}$ is the embedding of the first word in the span,
$\boldsymbol{u}^{(s,t)}_{\text{last}} = \boldsymbol{h}_t$ is the embedding of the last word,
$\boldsymbol{u}^{(s,t)}_{\text{head}}$ is the embedding of the head word,
and $\boldsymbol{\phi}^{(s,t)}$ is a vector of surface features, such as the length of the span

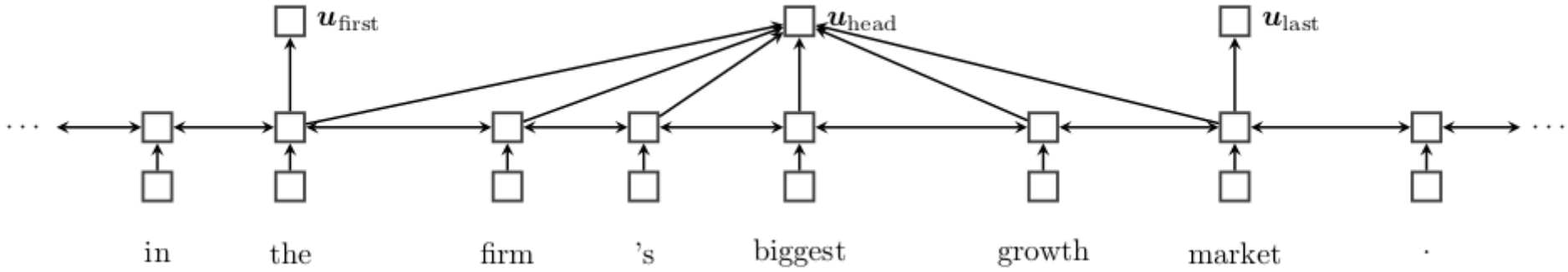
Algorithms for Coreference Resolution

Using mention embedding

- Given a set of mention embedding, each mention i and candidate antecedent a is scored as

$$\psi(a,i) = \psi_S(a) + \psi_S(i) + \psi_M(a,i)$$
$$\psi_S(a) = \text{FeedForward}_S(\boldsymbol{u}^{(a)})$$
$$\psi_S(i) = \text{FeedForward}_S(\boldsymbol{u}^{(i)})$$
$$\psi_M(a,i) = \text{FeedForward}_M([\boldsymbol{u}^{(a)}; \boldsymbol{u}^{(i)}; \boldsymbol{u}^{(a)} \odot \boldsymbol{u}^{(i)}; \boldsymbol{f}(a,i,\boldsymbol{w})]),$$

  - where $\boldsymbol{u}^{(a)}$ and $\boldsymbol{u}^{(i)}$ are the embeddings for spans a and i respectively, as defined in the previous equation on the vertical vector concatenation



A bidirectional recurrent model of mention embeddings (Lee et al., 2017)

Algorithms for Coreference Resolution

Clustering Mentions: Entity-based models

- It's more realistic as coreference resolution is a clustering problem rather than classification or ranking
- Entity-based model require a scoring function at the entity level, e.g.

$$\max_{z} \sum_{e} \psi_E(\{i : z_i = e\})$$

where z_i is the entity referenced by mention i, and $\psi_E(\{i : z_i = e\})$ is a scoring function applied to all mentions i that are assigned to entity e.

- To implement this in practice requires some form of search to group mentions so that they can be scored:
  - Brute force
  - Incrementally build up clusters
  - Actions learnt by reinforcement learning

Summary

- Meaning in natural language can be modelled in several ways, e.g.
  - using logic or
  - using relations between words (synonyms, hypernyms, etc.)
- In NLP we use formalism (first-order-logic, predicate structures, etc.) and resources (WordNet, DBpedia, etc.) to enrich text with meaning and knowledge about the world
- Modelling reference expressions add another layer of knowledge to the text, and it's crucial in applications such as machine translation and automatic summarisation

Australian National University

References

- Chapters 19, 26, Speech and Language Processing (3rd ed. draft)
- Chapter 12, Natural Language Processing. Jacob Eisenstein. 2018.
- Wiseman, S. J., A. M. Rush, S. M. Shieber, and J. Weston. Learning anaphoricity and antecedent ranking features for coreference resolution. In ACL. 2015.
- Lee, K., L. He, M. Lewis, and L. Zettlemoyer. End-to-end neural coreference resolution. In EMNLP. 2017.