Discourse

COMP90042 Lecture 20

Beyond the sentence

- **Discourse**: a coherent, structured group of sentences (utterances)

Yesterday, Ted was late for work. [It all started when his car wouldn't start. He first tried to jump start it with a neighbour's help, but that didn't work.] [So he decided to take public transit. He walked 15 minutes to the tram stop. Then he waited for another 20 minutes, but the tram didn't come. The tram drivers were on strike that morning.] [So he walked home and got his bike out of the garage. He started riding but quickly discovered he had a flat tire. He walked his bike back home. He looked around but his wife had cleaned the garage and he couldn't find the bike pump.] He started walking, and didn't arrive until lunchtime.

Three Key discourse tasks

- Discourse segmentation
- Discourse parsing
- Anaphor resolution

Discourse segmentation

- Assumption: text can be divided into a number of discrete, contiguous sections
- Task: classifying whether a boundary exists between any two sentences

An unsupervised approach

- TextTiling algorithm: looking for points of low lexical cohesion
- For each sentence gap:
  - Create two BOW vectors consisting of words from *k* sentences on either side of gap
  - Use cosine to get a similarity score (*sim*) for two vectors
  - For gap *i,* calculate a depth score, insert boundaries when depth is greater than some threshold

$$depth(\text{gap}_i) = (sim_{i-1} - sim_i) + (sim_{i+1} - sim_i)$$

Text Tiling example (k=2)

He walked 15 minutes to the tram stop.

Dot product: 3

Then he waited for another 20 minutes, but the tram didn't come.

Dot product: 2

The tram drivers were on strike that morning.

Dot product: 0

So he walked home and got his bike out of the garage.

Dot product: 2

He started riding but quickly discovered he had a flat tire

Dot product: 3

He walked his bike back home.

Dot product: 1

He looked around but his wife had cleaned the garage and he couldn't find the bike pump.

Supervised discourse segmentation

- Get data from easy sources (documents with paragraphs)
- Apply a binary classifier to identify boundaries
- Or use sequential classifiers
  - Potentially include classification of section types (introduction, conclusion, etc.)
- Integrate a wider range of features, including
  - distributional semantics
  - coreference cues
  - discourse markers

Discourse parsing

- A proper discourse must be coherent

Then he waited for another 20 minutes, but the tram didn't come.

✓ *The tram drivers were on strike that morning.*

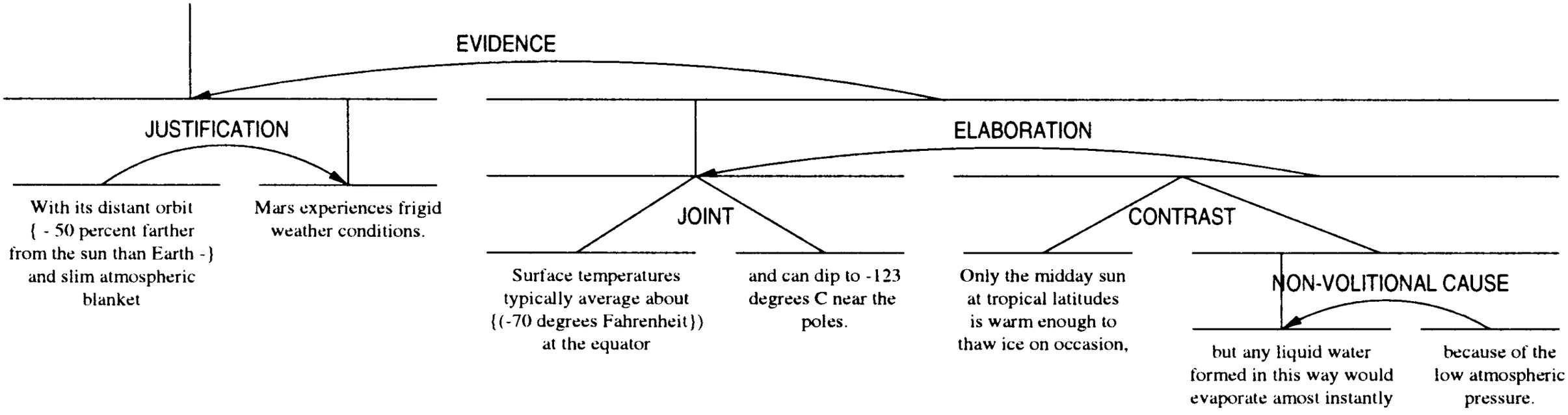
✗ *His bike had a flat tire.*

- Discourse units (DUs) are related by specific coherence relations
- Two related DUs form a new DUs
- All DUs in a coherent discourse must be related
- A discourse will form a tree, which can be parsed

An RST Tree

RST= Rhetorical Structure Theory

23 relations, most with a *nucleus* and *satellite*



Source: Marcu (2000), *The Rhetorical Parsing of Unrestricted Texts*, Computational Linguistics

Parsing using discourse markers

- Some discourse markers (cue phrases) explicitly indicate relations
  - Some examples: *although*, *but*, *for example*, *in other words*, *so*, *because, in conclusion*,…
- Can be used to build a simple rule-based parser
- However, (another discourse marker!)
  - Many relations are not marked by discourse marker at all
    - Particularly at the supra-sentential level
  - Many important discourse markers (e.g. *and*) ambiguous
    - Sometimes not a discourse marker
    - Can signal multiple relations

Full-text Discourse Parsing

- Full CYK/chart discourse parsing of entire texts not practical
- Most existing discourse parsers use a greedy approach

- Steps:
  - First, segment into elementary DUs (EDUs)
  - Classify all adjacent EDUs as being same DU or not
  - Combine most probable into a single DU
  - Repeat until entire text is a single DU

Discourse parsing features

- Discourse markers
- Starting/ending *n*-grams
- Location in the text
- Syntax (POS, dependency arcs, CFG productions)
- Lexical and distributional similarities
- Discourse structure within DUs
- Relations in adjoining DUs

Why identify Text structure?

- Summarization
- Sentiment analysis
- Argumentation
- Authorship attribution
- Essay scoring
- Anaphor resolution

Anaphors

- **Anaphor**: linguistic expressions that refer back to earlier elements in the text
- Anaphors have a **antecedent** in the discourse, often but not always a noun phrase

Yesterday, Ted was late for work. It all started when his car wouldn't start.

- Pronouns are the most common anaphor
- But there are various others
  - Demonstratives (*that problem*)
  - Definites (*the problem*)

Antecedent Restrictions

- Pronouns must agree in *number* with their antecedents

His coworkers were leaving for lunch when Ted arrived. They invited him, but he said no.

- Pronouns must agree in *gender* with their antecedents

Sue was leaving for lunch when Ted arrived. She invited him, but he said no.

- Pronouns whose antecendents are the subject of the same syntactic clause must be *reflexive (…self)*

Ted was angry at him. [*him* ≠ Ted]
Ted was angry at himself. [*himself* = Ted]

Antecedent Preferences

- The antecedent should satisfy the selectional preferences of the verb
- The antecedents of pronouns should be recent

He waited for another 20 minutes, but the tram didn't come. So he walked home and got his bike out of the garage. He started riding it to work.

- The antecedent should be salient, as determined by grammatical position
  - Subject > object > argument of preposition

Ted usually rode to work with Bill. He was never late.

The Centering Algorithm

- Idea: at any given moment, discourse is focused on a single entity, the "center"
- Goal: assign pronoun to antecedents in a manner which avoids "rough" shifts in the "center"

- Definitions
  - U_n = utterance n
  - $C_f(U_n)$ = list of entities of U_n, ordered by salience
  - $C_b(U_n)$ = backward center of U_n, highest ranked entity $C_f(U_{n-1})$ that appears in $C_f(U_n)$
  - $C_p(U_n)$ = preferred forward center of U_n, highest ranked entity of $C_f(U_n)$

The Centering Algorithm

- If some entity in $C_f(U_{n-1})$ is realized as a pronoun in U_n, then $C_b(U_n)$ must be realized as a pronoun.

Ted worked in an office in the city.
He usually rode there with Bill.
He had a nicer car than Ted.

- Prefer assignments of pronouns where $C_b(U_n) = C_b(U_{n+1})$

Ted worked in an office in the city.
He usually rode there with Bill.
He was never late.

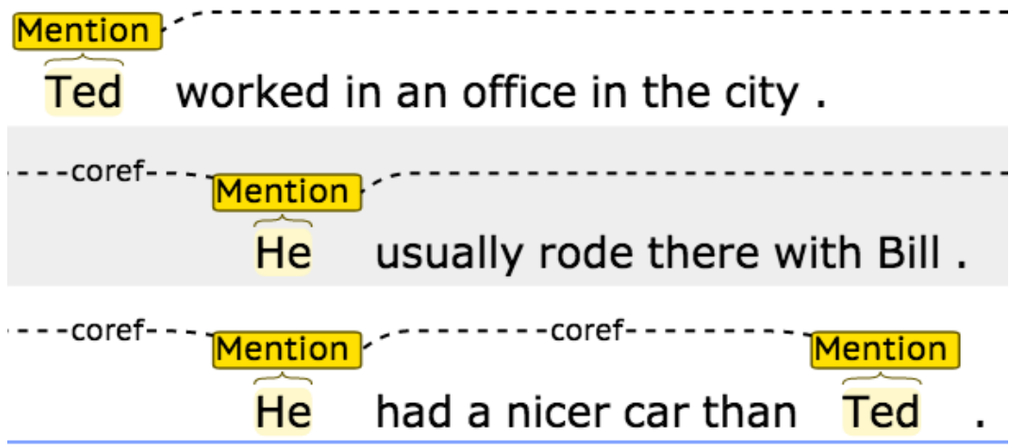
- Otherwise, prefer $C_b(U_n) = C_p(U_n)$

Supervised anaphor resolution

- Build a binary classifier for anaphor/antecedent pairs
  - Use confidence values to optimize globally
- Convert restrictions and preferences into features
  - Binary features for number/gender compatibility
  - Include features about type of antecedent
- With enough data, can approximate the centering algorithm
- But also easy to include features which indicate tendencies, rather than rules
  - Like repetition, parallelism

Anaphora Resolution Tools

- Stanford CoreNLP includes pronoun coreference models
  - rule-based system does very well
  - considerably faster than learned models



SYSTEM	LANGUAGE	PREPROCESSING TIME	COREF TIME	TOTAL TIME	F1 SCORE
Deterministic	English	3.87s	0.11s	3.98s	49.5
Statistical	English	0.48s	1.23s	1.71s	56.2
Neural	English	3.22s	4.96s	8.18s	60.0
Deterministic	Chinese	0.39s	0.16s	0.55s	47.5
Neural	Chinese	0.42s	7.02s	7.44s	53.9

Source: https://stanfordnlp.github.io/CoreNLP/coref.html
Evaluated on CoNLL 2012 task.

Motivation for Anaphor resolution

- Essential for deep semantic analysis
  - Very useful for QA, e.g., reading comprehension

Ted's car broke down. So he went over to Bill's house to borrow his car. Bill said that was fine.

Who borrowed a car?

A final word

- For many tasks, the larger context of language is important
- Traditional NLP has been sentence-focused, but that is beginning to change…

Further reading

- J&M2, Ch 21.1-21.3, 21.5-21.6