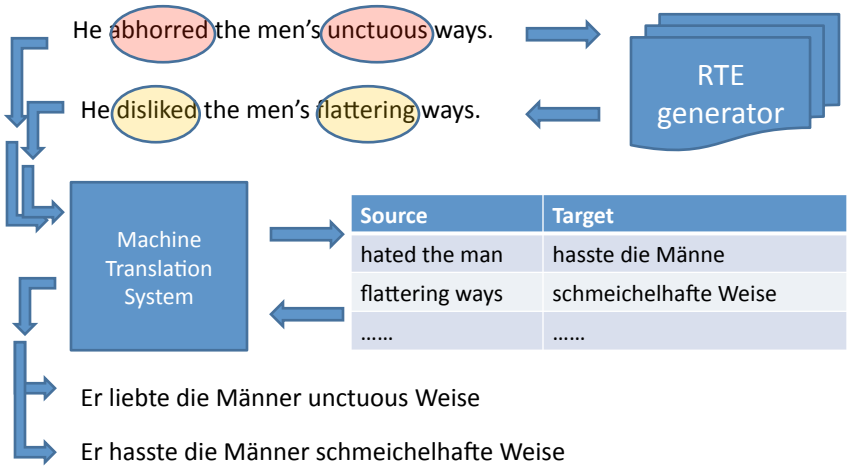
CS544: Paraphrase Acquisition

March 1, 2012

Zornitsa Kozareva
USC/ISI
Marina del Rey, CA
kozareva@isi.edu
www.isi.edu/~kozareva

Machine Translation

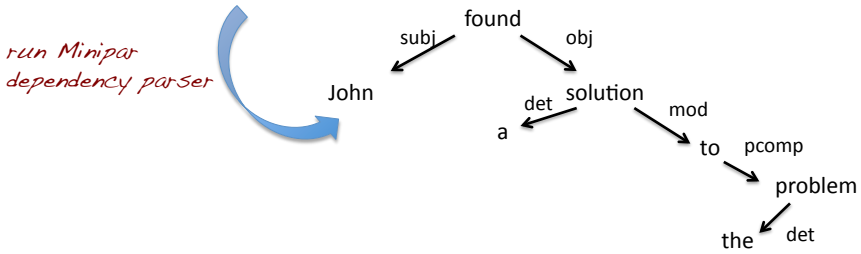
(Mirkin et al., 2009)



Dependency Tree

- A sentences can be represented by a set of dependency relations that form a tree

John found a solution to the problem.

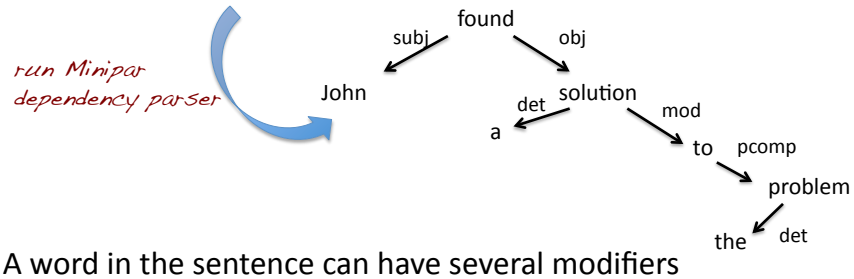


2

Dependency Tree

- A sentences can be represented by a set of dependency relations that form a tree

John found a solution to the problem.

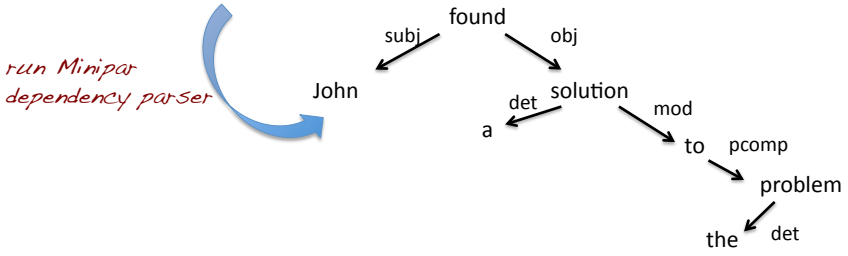


- A word in the sentence can have several modifiers
- A *head* word is the governor, a *modifier* is the dependent
- The root of the dependency tree does not modify any word

3

Dependency Tree

John found a solution to the problem.



- Links represent dependency relations ←
- The direction of the links is from head to modifier in the relation

John ← found

- Labels represent types of dependency relations

John ←subj— found

4

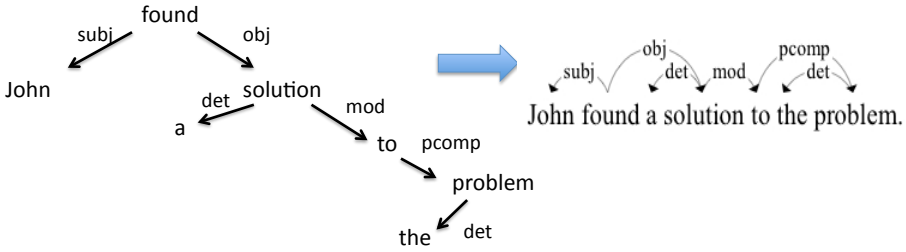
Example of Minipar's Dependency Relations

Relation	Description	Example
subj	subject of a verb	**John** loves Mary.
det	determiner of a noun	**the** dog
gen	genitive modifier of a noun	**John's** dog
mod	adjunct modifier of any head	**tiny** hole
nn	prenominal modifier of noun	**station** manager
appo	appositive of a noun	the CEO, **John**

5

Paths in Dependency Tree

• Representation

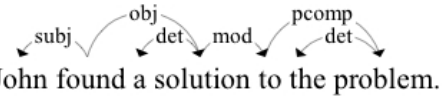


• In Minipar, each link between two words in the dependency tree represents a direct semantic relationship

•However, a path allows to capture indirect semantic relationships between two content words

6

Paths in Dependency Tree



• A path concatenates the dependency relations and words along the path

Ex: What is the path between *John* and *problem*

N:subj:V<-find->V:obj:N->solution->N:to:N

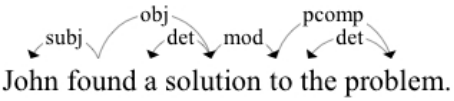
(X finds solution to Y)

N:to:N<-solution<-N:obj:V<-find->V:subj:N

(reverse path)

7

Paths in Dependency Tree



• A path concatenates the dependency relations and words along the path

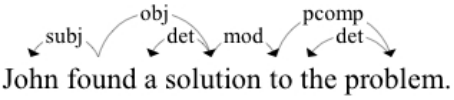
root

SlotX *N:subj:V<-**find**->V:obj:N->solution->N:to:N* SlotY

Filler for SlotX is *John*
Filler for SlotY is *problem*

8

Paths in Dependency Tree



• A path concatenates the dependency relations and words along the path

*N:subj:V<-**find**->V:obj:N->solution->N:to:N*

• Dependency relations that are not slots are called internal relations

Ex. *find->V:obj:N->solution*

9

Discovering Inference Rules From Text

- Extract all paths in a corpus together with the slot fillers
- Measure the relatedness of a slot filler and a path
- Measure the similarity between two paths
  (the more features two paths share, the more similar they are)
- Use on the Extended Distributional Hypothesis

> If two paths tend to occur in similar contexts, the meaning of the paths tends to be similar

10

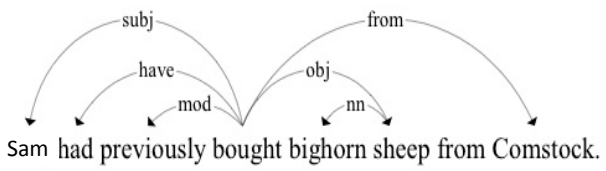
Constraints for Path Extraction

- **Slot fillers** must be **nouns or pronouns**, because in the **paraphrasing** task the slots correspond to variables which are instantiated by entities
- Any dependency relation which does not connect two content words (noun, verb, adjective or adverb) is excluded from the path (won't extract a relation between **John** and **a**)
- An internal relation must be between a verb and an object-noun (because in **paraphrasing** the center must be a verb)

•Obtain meaningful paths
•Reduce the amount of computation
•Remove long paths that have few occurrences

11

Example of Extracted Paths



Extracted Path	Meaning
N:sbuj:V<-buy->V:from:N	X buy something from Y
N:sbuj:V<-buy->V:obj:N	X buys Y
N:sbuj:V<-buy->V:obj:N->sheep->N:nn:N	X buys Y sheep
N:nn:N<-sheep<-N:obj:V<-buy->V:from:N	X sheep is bought from Y
N:obj:V<-buy->V:from:N	X is bought from Y

12

Example of Slot Fillers for Two Paths

"X finds a solution to Y"	
SlotX	***SlotY***
commission	strike
committee	civil war
committee	crisis
government	crisis
government	problem
he	problem
i	situation
legislator	budget deficit
sheriff	dispute

"X solves Y"	
SlotX	***SlotY***
committee	problem
he	crisis
government	problem
he	mystery
she	problem
petition	woe
researcher	mystery
resistance	crime
sheriff	murder

13

Discovering Inference Rules From Text

✓ Extract all paths in a corpus together with the slot fillers

• Measure the relatedness of a slot filler and a path

14

Mutual Information

• Mutual Information (MI) is between two events

$$PMI(w_1, w_2) = \log_2 \frac{p(w_1, w_2)}{p(w_1) * p(w_2)}$$

• But, in the paraphrase model we have three events: ***path***, ***slot*** and ***slot filler***

slot can be either SlotX or SlotY

$$PMI(pattern, slot, filler) = \log \frac{P(pattern, slot, filler)}{P(slot) * P(pattern \mid slot) * P(filler \mid slot)}$$

This model assumes that the path and the filler are conditionally independent given a slot

15

MI for a *Path-Slot-Filler* Triple

$$PMI(pattern, slot, filler) = \log \frac{P(pattern, slot, filler)}{P(slot) * P(pattern \mid slot) * P(filler \mid slot)}$$

$$PMI(pattern, slot, filler) = \log \frac{\frac{|pattern, slot, filler|}{|*,*,*|}}{\frac{|*, slot, *|}{|*,*,*|} \times \frac{|pattern, slot, *|}{|*, slot, *|} \times \frac{|*, slot, filler|}{|*, slot, *|}}$$

$$= \log \frac{|pattern, slot, filler| \times |*, slot, *|}{|pattern, slot, *| \times |*, slot, filler|}$$

16

Generated Triples

"X pulls body from Y"							
SlotX		Freq	MI	SlotY		Freq	MI
	driver	1	2.45		bus	2	3.09
	equipment	1	1.65		coach	1	2.05
	police	2	2.24		debris	1	2.36
	rescuer	3	4.84		feet	1	1.75
	resident	1	1.60		hut	1	2.73
	who	2	1.32		landslide	1	2.39
	worker	1	1.37		metal	1	2.09
					wreckage	3	4.81

17

Discovering Inference Rules From Text

- ✓ Extract all paths in a corpus together with the slot fillers
- ✓ Measure the relatedness of a slot filler and a path
- • Measure the similarity between two paths
  (the more features two paths share, the more similar they are)

18

Similarity Between Pairs of Slots

$$sim(slot_1, slot_2) = \frac{\sum_{w \in T(p_1,s) \cap T(p_2,s)} mi(p_1,s,w) + mi(p_2,s,w)}{\sum_{w \in T(p_1,s)} mi(p_1,s,w) + \sum_{w \in (p_2,s)} mi(p_2,s,w)}$$

p1 and *p2* are paths
s is a slot
T(pi,s) is a set of words that fill in the *s* slot for path *pi*

19

Generated Triples

"X pulls body from Y"			
SlotX	MI	SlotY	MI
driver	2.45	bus	3.09
equipment	1.65	coach	2.05
police	2.24	debris	2.36
rescuer	4.84	feet	1.75
resident	1.60	hut	2.73
who	1.32	landslide	2.39
worker	1.37	metal	2.09
		wreckage	4.81

"X retrieves Y"			
SlotX	MI	SlotY	MI
driver	3.45	information	2.20
police	2.65	body	2.05
dog	2.15	money	3.45
		weapon	1.00

20

Similarity Between Paths

$$sim(p_1, p_2) = \sqrt{sim(SlotX_1, SlotX_2) \times sim(SlotY_1, SlotY_2)}$$

p1 and *p2* are paths defined by the geometric average of similarities of their SlotX and SlotY

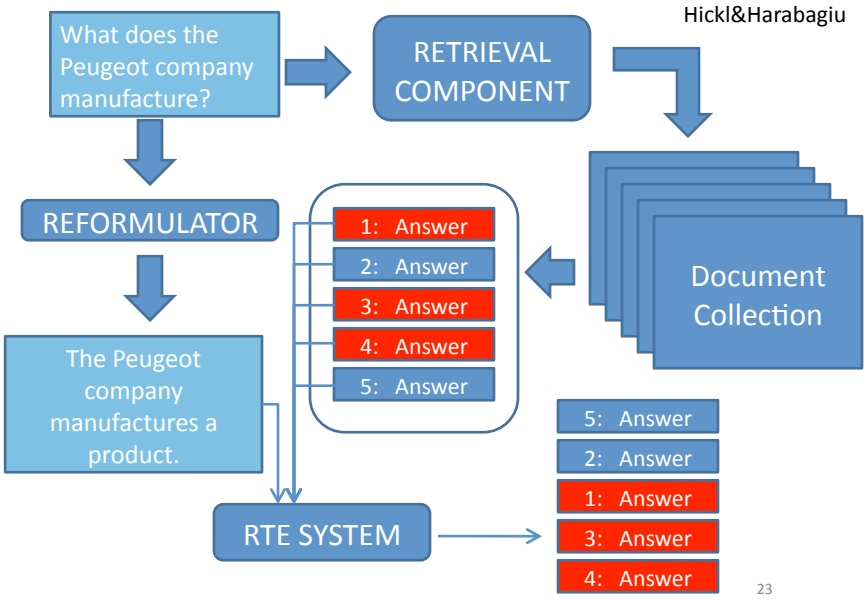
21

Finding the Most Similar Path

- Given a path, find the most similar path
  - For every word, store its SlotX and SlotY in all paths
  - Form candidate paths
  - Count the number of features between these paths

22

Question answering



23

Question	Path	Manual Paraphrases	Automated Paraphrases
What does the Peugeot company manufacture?	*X* manufactures *Y*		

Quick Task:

1) Write your worn paraphrase rules.
2) Who generated more rules the human or the machine?
3) How well do you think the paraphrases generated by the machine are?

24

Question	Path	Manual Paraphrases	Automated Paraphrases
What does the Peugeot company manufacture?	*X* manufactures *Y*	*X* makes *Y* *X* produce *Y* *X* is in *Y* business *Y* is manufactured by *X* *Y* is provided by *X* *Y* is *X*'s product *Y* is product from *X* *Y* is *X* product *Y* is product made by *X* Y is example of *X* product *X* is manufacturer of *Y* find *Y* in *X*'s product line find *Y* in *X* catalog	*X* produces *Y* *X* markets *Y* *X* develops *Y* *X* is supplier of *Y* *X* ships *Y* *X* supplies *Y* *Y* is manufactured by *X* *X* is a maker of *Y* *X* introduces *Y* *X* exports *Y* *X* makes *Y* *X* builds *Y* *X*'s production of *Y* *Y* is bought from *X* *X*'s line of *Y* *X* assembles *Y* *X* is *Y* maker *X*'s *Y* factory *X* is manufacturer of *Y* ……

25

Evaluation

Question	Path	Manual	Automated	Intersection	Accuracy
What does the Peugeot company manufacture?	*X* manufacture *Y*	13	37	4	92.5%
Who is the author of the book, "The Iron Lady: A Biography of Margaret Thatcher"?	*X* is author of *Y*	7	21	2	52.5%
Why did David Koresh ask the FBI for a word processor?	*X* asks *Y*	2	23	0	57.5%

26

More Examples

Question	Path	Manual Paraphrases	Automated Paraphrases
Who is the author of the book, "The Iron Lady: A Biography of Margaret Thatcher"?	X is author of Y	Y is the work of X X is the writer of Y X penned Y X produced Y X authored Y X chronicled Y X wrote Y	X co-authors Y X is co-author of Y X writes Y X edits Y Y is co-authored by X Y is authored by X X tells story in Y X translates Y X writes in Y X notes in Y

27

Next Time …

- Read the following papers
  - "Low-Cost Supervision for Multiple-Source Attribute Extraction", Joseph Reisinger and Marius Pasca
  - "Weakly-Supervised Acquisition of Open-Domain Classes and Class Attributes from Web Documents and Query Logs", Marius Pasca and Benjamin Van Durme
- Answer the following questions on a piece of paper
  - Define what is an attribute
  - Describe what is the attribute extraction task
  - Why is it important
  - What kinds of application would benefit from it
  - Describe the approach and give an example
  - Described the evaluation
  - Discuss +/- of the approach
  - Propose an alternative solution to the problem

28