

Paraphrases

- What are paraphrases?
 - alternate ways to convey same information
 - ex. “The parrot is dead”, “The parrot has ceased to be”, “This is a late parrot”
 - in the “Dead Parrot Sketch”, by Monty Python, this fact is conveyed in over 15 ways- and this is by no means an exhaustive list
- Why is it useful to have an understanding of paraphrases?

Practical Reasons

- Automatic Language Processing
 - existence of paraphrases greatly complicates this
 - ex. to find relation $\text{love}(X,Y)$, cannot simply search text for “X loves Y”
- Multidocument Summarization
 - can be used to recognize and avoid duplicate information
 - ex. used in Columbia’s MultiGen, which summarizes multiple documents (used in Newsblaster)
- Text Generation
 - can be used to produce varied and fluent text

Linguistic/Theoretical Reasons

- paraphrasing is common to all natural languages; humans use them often and with ease
- open question as to what relations define paraphrases (not just synonyms)
- two major linguistic theories, Generative-Transformation Grammar and Meaning-Text Theory rely heavily on paraphrases
 - GTG (Chomsky, 1957) and (Harris, 1981) uses meaning-preservation transformations- these are basically syntactic paraphrases
 - MTT (Melcuk, 1988) has a lexicon including 60 paraphrasing rules, which are, in theory, enough to cover all paraphrases in any language

Earlier Approaches

- only include lexical paraphrases, not phrasal or syntactically based ones
- use either manually collected paraphrases selected for a particular domain, or
- use existing lexical resources such as WordNet

Barzilay and McKeown (2001)

- Regina Barzilay
 - did her Ph.D. at Columbia
 - developed MultiGen, a system for doing multidocument summarization
 - currently an Assistant Professor at MIT
- Kathleen McKeown
 - Professor and Chair of Computer Science at Columbia University
 - was Barzilay's supervisor

Approach

- corpus based
 - uses parallel English translations of novels
- builds on existing machine learning methodology
- based on the assumption that phrases in aligned sentences which appear in similar contexts are paraphrases
- relies on morphological information and part-of-speech tagging
- some advantages:
 - does not rely on human-collected data
 - provides insight as to interchangeability of paraphrases

Corpus

- consists of 11 English translations of 5 novels
- different from classic MT corpus:
 - a complete match between the words of related sentences is impossible
 - no two translations are the same
 - open to different interpretations
 - there's an irregularity in word matches, as the same word is often used in both translations
 - word-paraphrase pairs have lower co-occurrence rates than word-translation pairs in MT
 - however, this helps the process of matching sentences from different translations

Preprocessing

- align sentences using dynamic programming with weight function based on # of common words
 - achieves good results, due to 42% of words in corresponding sentences being identical
 - produces 44,562 pairs of sentences
 - 126 were analyzed, and 120 (94.5%) were identified as correct
- use a POS tagger and chunker to identify noun and verb phrases in the sentences
 - these become the atomic units in the algorithm
- record for each token its derivational root, using CELEX

Paraphrase Features

- paraphrase features include lexical and syntactic descriptions of the paraphrase pair, and contextual features
 - lexical feature set consists of a sequence of tokens for each phrase in the pair
 - syntactic feature consists of a sequence of POS tags for each phrase in the pair, where indices indicate equal words or words with the same root
 - ex. (“the vast chimney”, “the chimney”) \implies
 (“DT₁ JJ NN₂”, “DT₁ NN₂”)

Contextual Features

- a contextual feature is a combination of the left and right syntactic contexts surrounding known paraphrases
 - ex. “tried to **comfort** her,” , “tried to **console** her,”

⇒

$left_1 = "VB_1 TO_2"$ (“tried to”)

$left_2 = "VB_1 TO_2"$ (“tried to”)

$right_1 = "PRP\$_{3,4}"$ (“her,”)

$right_2 = "PRP\$_{3,4}"$ (“her,”)

Co-Training

- necessary: two distinct partitions of data which are going to be trained on, and a small labeled data set
- idea: use two learning algorithms; each one trains on one of the partitions, using the predictions generated by the other

Example: Web Pages and Hyperlinks

- goal: learn to download all of the of the CS faculty member pages
- labeled examples: a few faculty web pages, and a few unrelated pages
- step 1: using labeled examples, learn which links are likely to lead to CS faculty member pages and which are not
- step 2: using information from step 1, find new likely CS faculty pages and some negative examples
- step 3: using information from step 2, learn which links are likely to lead to CS faculty member pages, and which are not
- repeat steps 2 and 3 until some threshold is reached

Method

- Hypothesis: If the contexts surrounding two phrases are very similar, then the two phrases are likely to be paraphrases.
- Algorithm:
 1. Initialization: create seed paraphrases using matching words
 2. Training of the Contextual Classifier: the contexts surrounding paraphrases are extracted and filtered according to their predictive power
 3. Training of the Paraphrasing Classifier: these contexts are used to extract new paraphrases, which are filtered according to their predictive power
 4. were new paraphrases extracted?
 - yes: go to step 2
 - no: the algorithm is finished

Initialization

- create a set of positive paraphrasing examples using identical words in the aligned sentences with each other
- create a set of negative paraphrasing examples using identical words in the alignment with every other word in the aligned sentences

Definition: Strength

- strength of positive context $x = \frac{\# \text{ of times } x \text{ surrounds a positive example}}{\# \text{ of times } x \text{ appears}}$
- strength of negative context $x = \frac{\# \text{ of times } x \text{ surrounds a negative example}}{\# \text{ of times } x \text{ appears}}$

Training of the Contextual Classifier

- record contexts around positive and negative paraphrasing examples
- filter the contexts for strong predictors, based on their strength and frequency
 - select k rules ($k = 10$) with the highest frequency and strength $> 95\%$
- record all contexts with length \leq maximal length (in this case, 3)
- similarity between translations varies from one book to another, so the contextual classifier is trained for each pair of translations separately

Training of the Paraphrasing Classifier

- contextual rules are applied to corpus by searching sentence pairs for subsequences which match the left and right parts of the rules, and are less than N tokens apart
 - allowing them to be N tokens apart means that the algorithm can extract multi-word paraphrases
- paraphrasing rules recorded and filtered in a similar manner to contextual rules

Precision Test

- algorithm produces 9483 pairs of lexical paraphrases and 25 morpho-syntactic rules
- authors picked at random 500 paraphrasing pairs as test data
- performed two experiments: with and without context
 - human judge first given a pair without context, then asked to evaluate same pair with context
 - experiment done with two judges, neither of whom was the author
- authors were unable to evaluate recall, as corpus does not cover all English tokens, and direct comparison with an electronic thesaurus is impossible

Results

- without context, results were:
 - First judge: 439 (87.8%) accurate
 - Second judge: 426 (85.2%) accurate
 - agreement was 0.68 using Kappa coefficient
- with context, results were:
 - First judge: 459 (91.8%) accurate
 - Second judge: 457 (91.4%) accurate
 - agreement was 0.97 using Kappa coefficient

Coverage Test

- had a human extract paraphrases from 50 sentences
- had the algorithm extract paraphrases from same 50 sentences
- from 70 paraphrases extracted by the human, the algorithm identified 48 (69%) as such
- authors don't state whether there were false positives

Comparison With Melamed's System

- 60% of the dataset was evaluated, and each system produced 6,826 word pairs
- randomly ordered 1000 pairs were evaluated by six humans
- this system had 71.6% accuracy, vs Melamed's 52.7%

Comparison With WordNet

- selected 112 paraphrasing pairs which appeared at least 20 times in the corpus, and such that the words in each pair were in WordNet
- 35% were synonyms, 32% hypernymns, 18% siblings, 10% unrelated, and 5% covered by other relations.
- this was further evidence that synonymy and paraphrasing are not the same