Relation Alignment for Textual Entailment Recognition

Mark Sammons, V.G.Vinod Vydiswaran, Tim Vieira,
Nikhil Johri, Ming-Wei Chang, Dan Goldwasser,
Vivek Srikumar, Gourab Kundu, Yuancheng Tu,
Kevin Small, Joshua Rule, Quang Do, Dan Roth

Department of Computer Science
University of Illinois at Urbana-Champaign

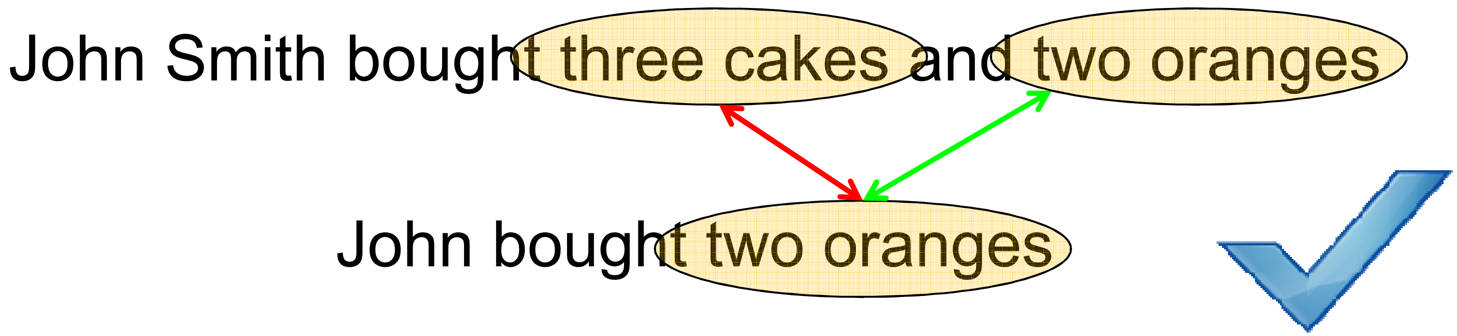
Cognitive Computation Group approach to RTE

- Problem statement: Given a corpus of entailment pairs (T,H)
  - Task requires us to 'explain' the content of the Hypothesis (H) using content in the Text (T) (plus some background knowledge).
  - If all elements in H can be explained, label is 'YES'; otherwise, 'NO'.
- Goals:
  - Attack the TE problem using divide-and-conquer strategy
    - Plug-and-play architecture
    - Avoid pipeline architecture where possible
  - Apply Machine Learning with "justifiable" features
- Assumptions:
  - Each element of H (word, phrase, predicate) is explained by one element of T (may require representing T differently).
  - Semantics is (largely) compositional: we can determine entailment for element s of (T, H) individually, and integrate them to get the global label for the entailment pair.

CCG TE System: Approach

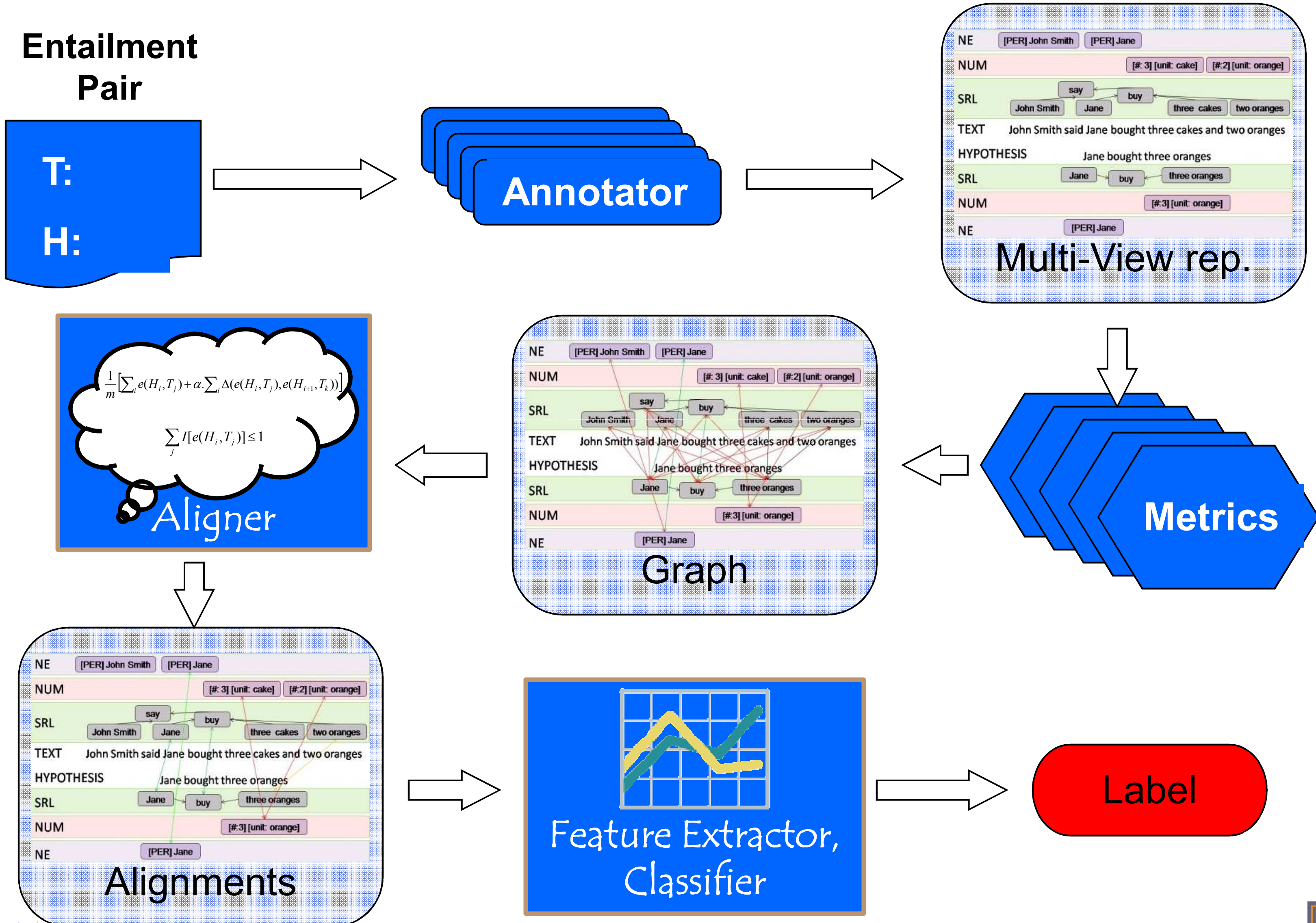
- Develop and apply Similarity Metrics for semantic elements annotated by standard NLP resources
  - Named Entity, Numerical Quantity, Semantic Role, Shallow Parse chunks; others in development
  - Allow encapsulation and modular incorporation of background knowledge
  - Compare two constituents (generated from a specific analysis source); return a real number $\in [-1, 1]$ (oppositional – not comparable – similar)
- Use similarity metrics to resolve local entailment decisions
- No guarantees that outputs are scaled – e.g. 0.8 may mean 'low similarity' for NE, 'nearly identical' for lexical metric

Alignment in RTE: Lexical Level



- But which decisions are relevant to global label? Many possible comparisons, when all constituents considered
- →Use alignment as basis for inference; select 'relevant' local decisions

System Overview

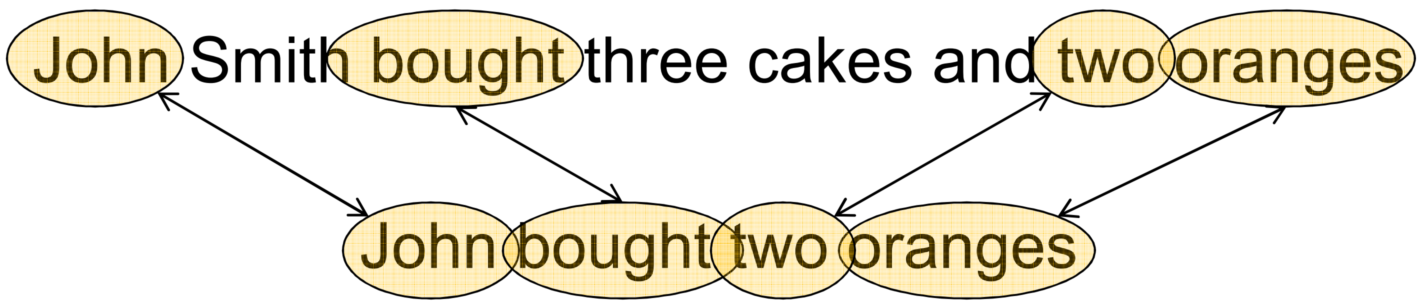


Outline: Alignment in Textual Entailment

- What is Alignment?
- Models of Alignment/Previous work
- CCG RTE
- Results and Conclusions

Alignment in RTE: Lexical Level

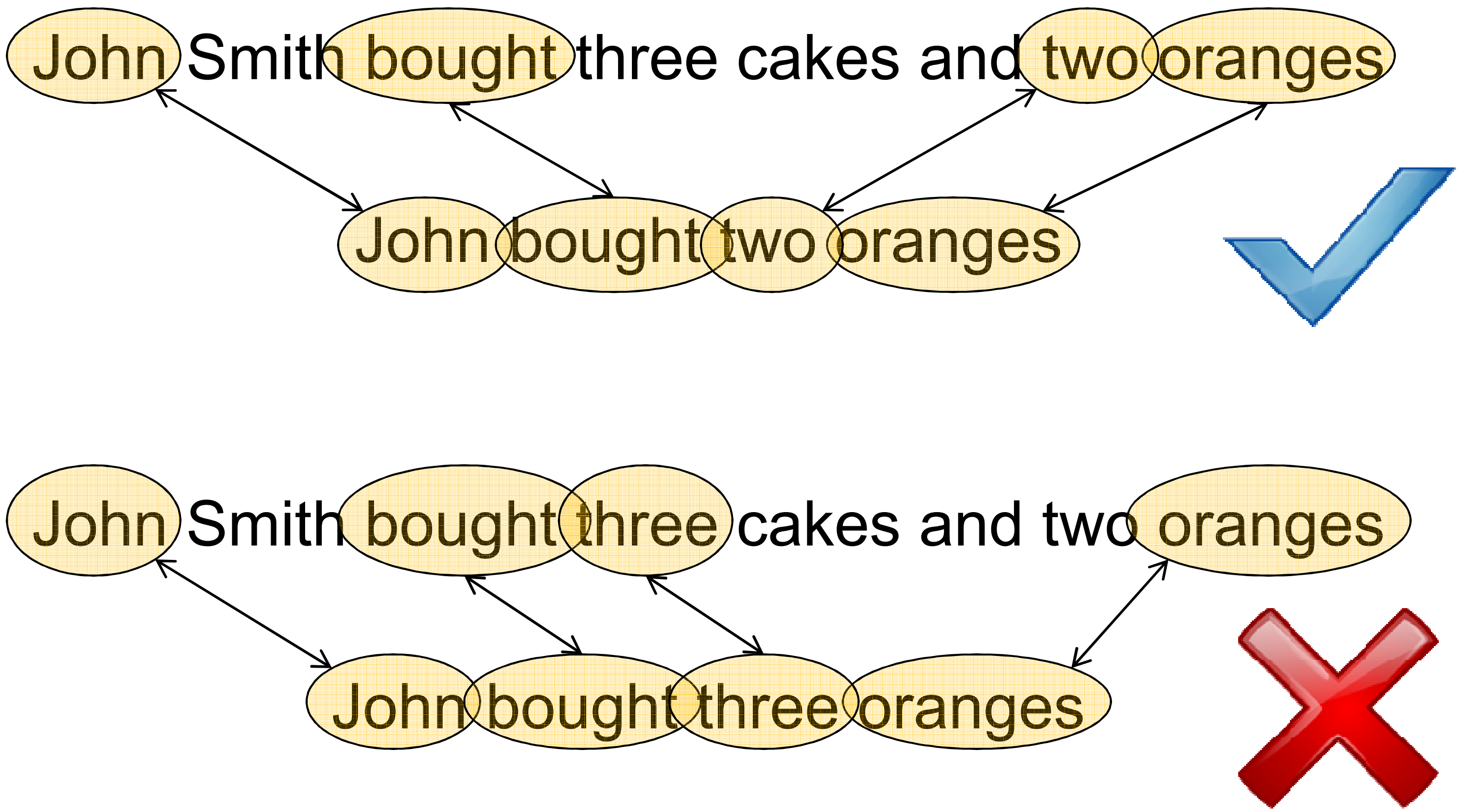
- Alignment: a mapping from elements in the Hypothesis to elements in the Text



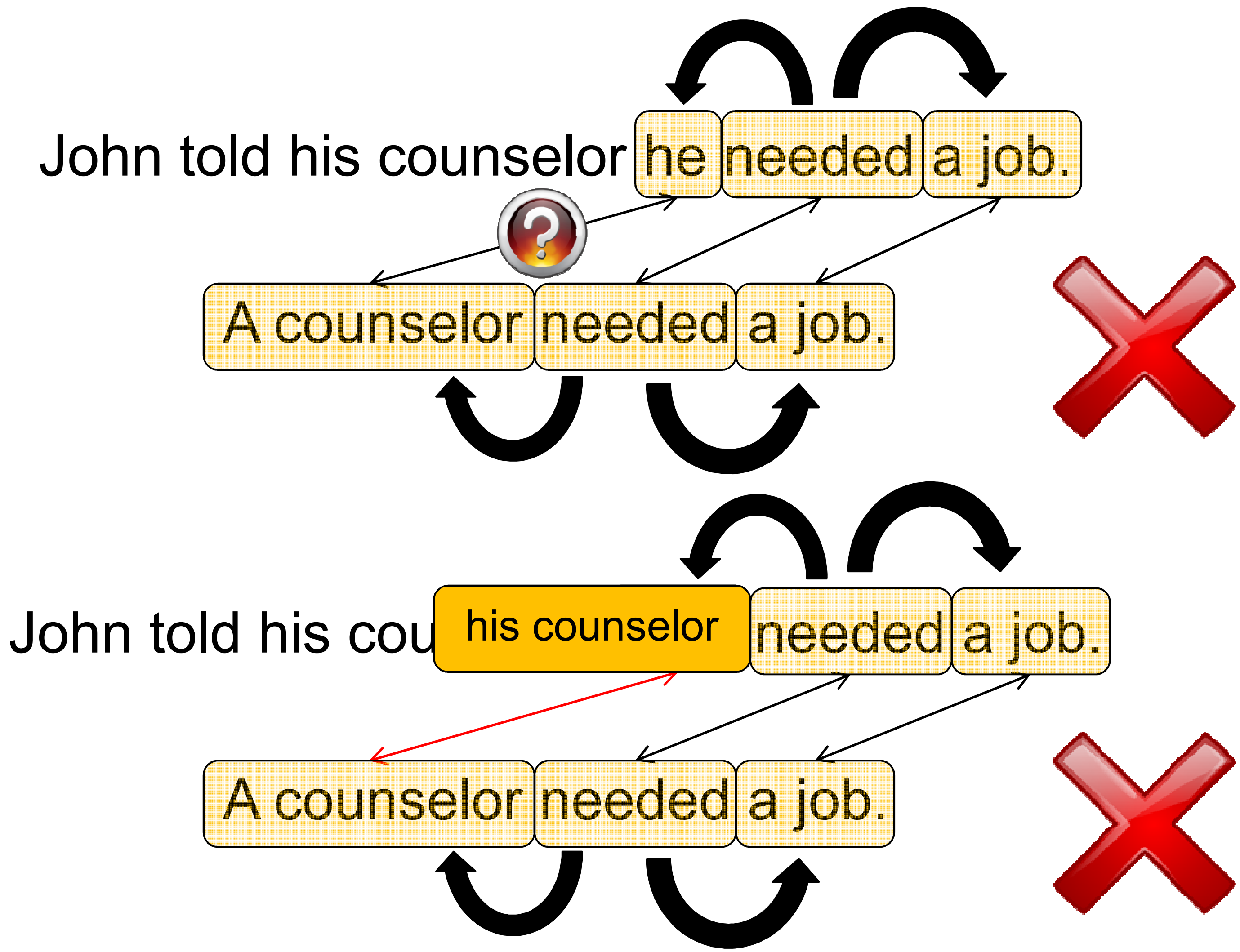
Alignment is Useful for Machine Learning in RTE

- Machine Learning approaches provide much-needed robustness for NLP tasks
- RTE data sets are small, given complexity of problem
- Global, 2- or 3-class label on each pair
- We would like to resolve entailment by combining local decisions (e.g. word-level, phrase level); but *which* decisions?
- Alignment can be used to select a subset of the many possible comparisons, and thereby augments global label with (proxy for) finer-grained structure; can be used…
  - …to determine active features
  - …to generate labels for local classifiers

Alignment in RTE: Lexical Level



Alignment in RTE: Parse level



Models of Alignment in RTE (Previous Work)

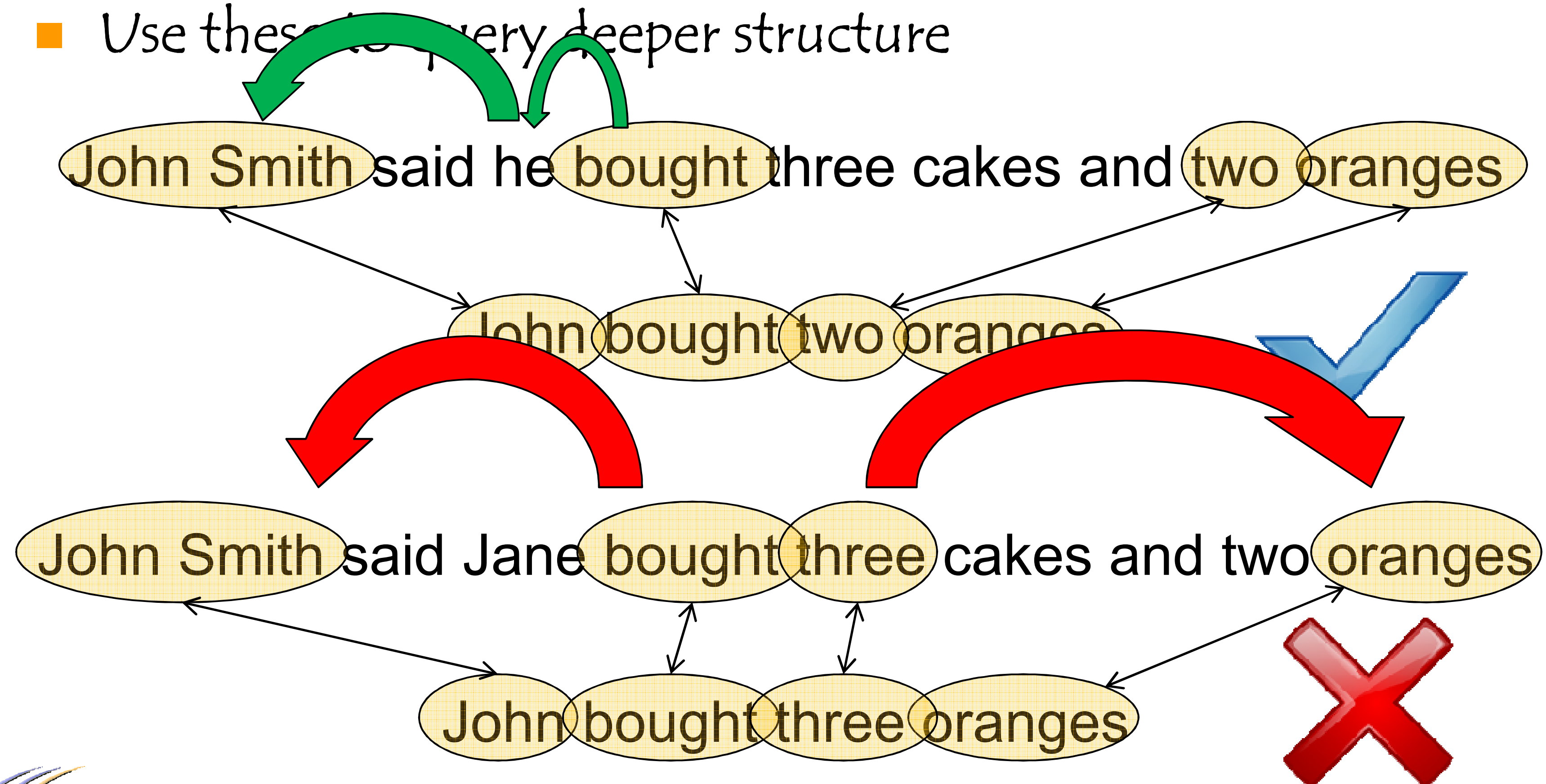
- Alignment as entailment discriminator:
  e.g. Zanzotto et al. (2006)
  - Use lexical matching + parse tree similarity measure to build 'best' match graph for entailment pairs
  - Allows robustness against parse errors, minor variations in structure
  - Use inter-pair graph similarity measure to determine entailment
  - "Alignment As Entailment"

- Alignment as feature selector:
  e.g. de Marneffe et al. (2007), Hickl et al. (2006)
  - Infer alignment over words/phrases in entailment pairs
  - Extract features from aligned constituents, train entailment classifier
  - "Alignment as Filter"

Approaches to "Alignment As Filter"

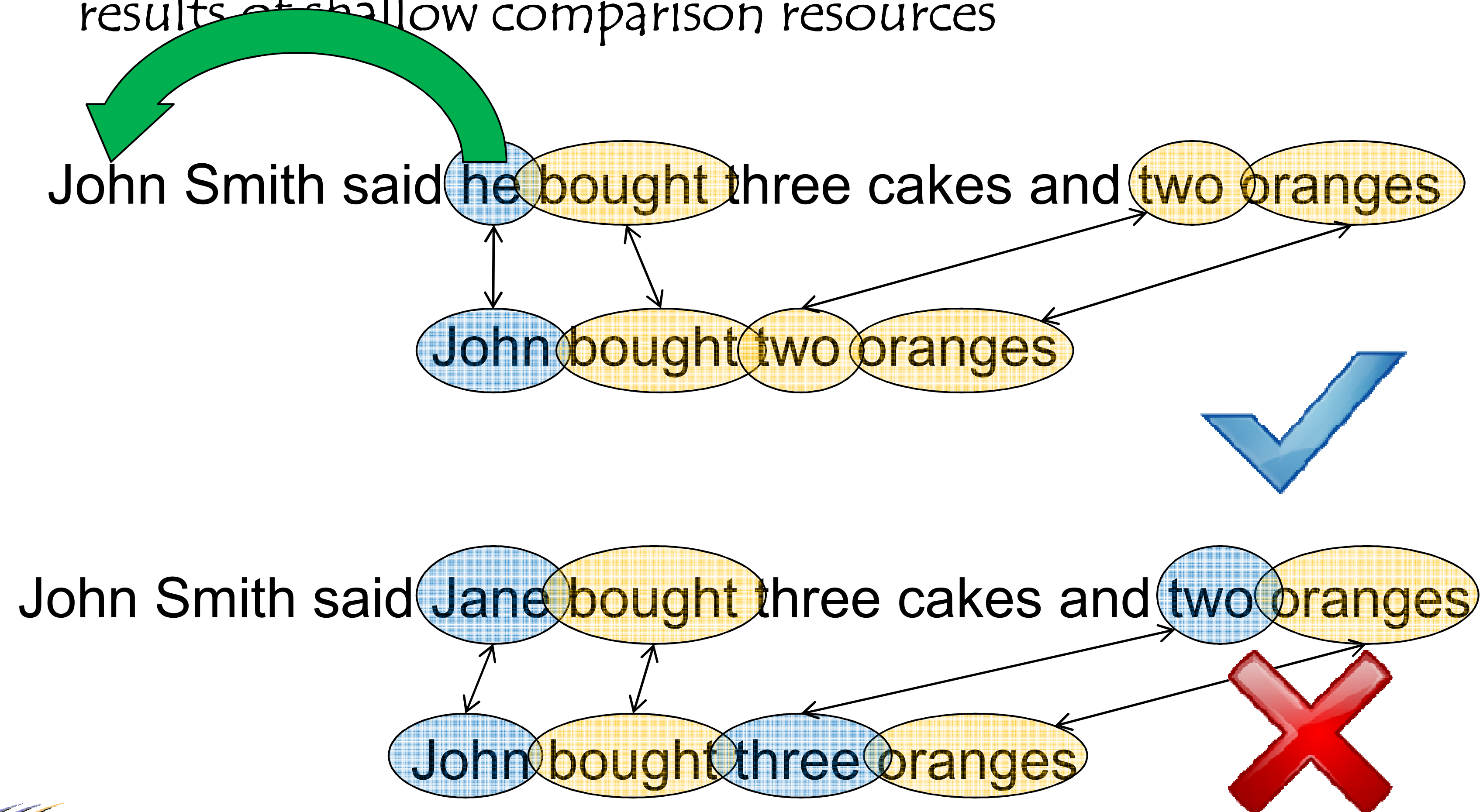
Shallow Alignment as Focus Of Attention

- Pick a "good" shallow alignment (possibly, highest scoring matches)
- Use these to query deeper structure



Using Structure as Focus Of Attention

- Find best structural match; base entailment results on results of shallow comparison resources



Some Questions

- It seems we have a choice:
  - Find a shallow alignment, and then rely on subsequent stages to model the structural constraint.

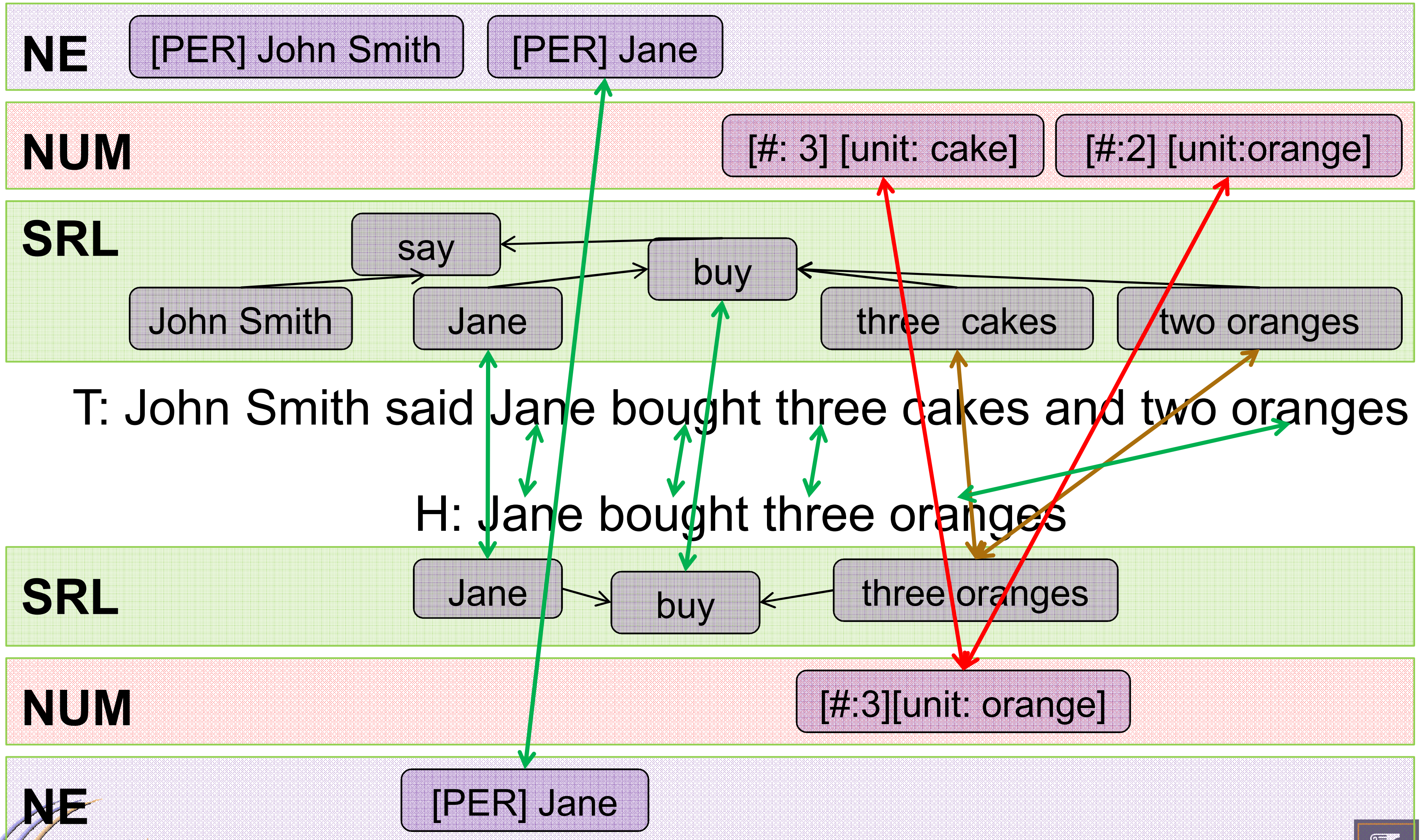
    OR
  - Use the structural constraint to inform the shallow alignment.
- Is one of the two approaches preferable?
  - Mistakes in shallow alignment → problem inferring structural constraint.
    - There may be many 'good' shallow alignments…
  - Errors in deep structure → problem selecting correct local decision
  - Other preprocessing errors – e.g. Coreference – will propagate either way
- Can we avoid these problems?

CCG RTE

Multiple alignments at multiple granularities

- Intuition: exploit differences/agreements between different views of the entailment pair; avoid canonization
- Accommodates analysis at different granularities
- Resources with comparable scores can compete with each other – pick the "best"
  - e.g. Words, Multi-word Expressions, Phrasal Verbs
- Unscaled resources occupy different alignments (SRL, NE)
- Metrics can return negative numbers; use magnitude in alignments, preserve negative edge label
  - May be useful for contradiction features

Multiple Alignments for RTE



Learning from Multiple Alignments

- Extract features based on individual alignments
  - Can use high-precision, low-recall resources as filter features
  - Typical match features within alignments – e.g. proportion of tokens matched
- Extract features based on agreement, disagreement between different alignments
  - E.g. Predicate-Argument, Numerical Quantities
- Allows graceful degradation if some resources are unreliable; learner assigns low weights to corresponding features

Multiple Alignments ctd.

- Model each alignment as optimization problem
  - Penalize distant mappings of neighboring constituents in H, T (proxy for deep structure – favor chunk alignment)
  - Constraints: each token in H can be covered exactly once by an aligned constituent; edge scores must account for number of constituents covered
  - Solve by brute-force search

$$\frac{1}{m}\left[\sum_i e(H_i, T_j) + \alpha.\sum_i \Delta(e(H_i, T_j), e(H_{i+1}, T_k))\right]$$

$$\sum_j I[e(H_i, T_j)] \leq 1$$

Feature Extraction

- Main types of features:
  - Features assessing quality of alignment in a given view
  - Features assessing agreement between views
- Quality of Alignment features:
  - Proportion of constituents matched in Word, NE, SRL views
  - "Distortion" of match pattern
- Agreement features:
  - Proportion of token alignments agreeing with SRL constituent alignments
  - Negation of predicate in SRL relation match
- Extension: Using Coreference:
  - Augment SRL predicates: add arguments using Coref chains
  - Introduces inter-sentence structure

Results and Conclusions

Results

Corpus	System					
	Baseline	No NE*	Basic NE	No WN	All*	All + Coref
RTE5 Dev	0.628	0.640	0.623	0.647	0.648	0.663
RTE5 Test	0.600	0.629	0.633	0.603	0.644	0.666

* Submitted runs had ~60 buggy alignments in dev test; results using non-buggy alignments shown here

Conclusions

- Good improvement over 'smart' lexical baseline; not (yet) close to best system (73.5%)
- Some further performance gain may be possible from tuning distance penalty in alignment optimization, additional features
- Reasonable behavior in ablation
- Coreference-based structure helps – improves shallow semantic match
- Possible utility from avoiding canonization (pipeline)
- Multiple Alignment + Feature Extraction offers flexible system, avoids shallow→deep, deep→shallow problems
- Alignment + Metric approach seems promising as general framework for RTE