Controlling Redundancy in Referring Expression Generation

Jette Viethen, Robert Dale – Macquarie University, Sydney

Emiel Krahmer – University of Tilburg, The Netherlands

Mariët Theune, Pascal Touset – University of Twente, The Netherlands

Mimicking redundancy in referring expressions increases human-likeness.

Using corpus statistics to fine-tune referring expression generation increases human-likeness.

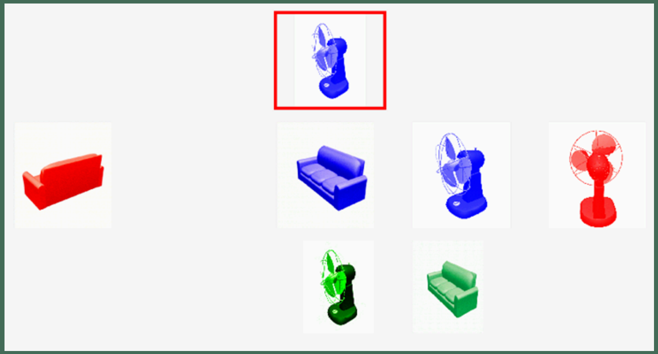
Outline

1) Referring Expression Generation (REG)
    - Redundancy in REG
2) The Graph-Based Algorithm
    - Cost Functions and Property Orderings
    - Redundancy in the Graph-Based Algorithm
3) The ASGRE TUNA Data
4) Tuning the Algorithm
    - Corpus-Based Costs and Free Properties
    - Frequent Properties Considered First
5) Evaluation
6) Conclusions and Future Work

Referring Expression Generation

...means automatically building

distinguishing object descriptions.



The small blue fan.

- ***Target referent***: the object to be described.
- ***Distractors***: other objects in the domain that the referent has to be distinguished from.
- ***Content selection*** from the *properties* of the referent and its *relations* to other objects.
- ***Minimal***: all the properties used are needed for descriptiveness.
- ***Redundant***: would still be descriptive if one or more properties were removed.

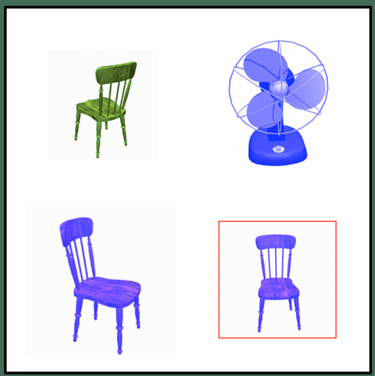
Redundancy in REG

- Humans use redundant properties.
- The Incremental Algorithm (IA) doesn't allow redundancy in a principled way ...
- ... but the Graph-Based Framework provides two parameters to control content selection: cost function, property ordering.
- We present the first corpus-based approach to setting these two parameters.
- The focus is on redundantly used properties.

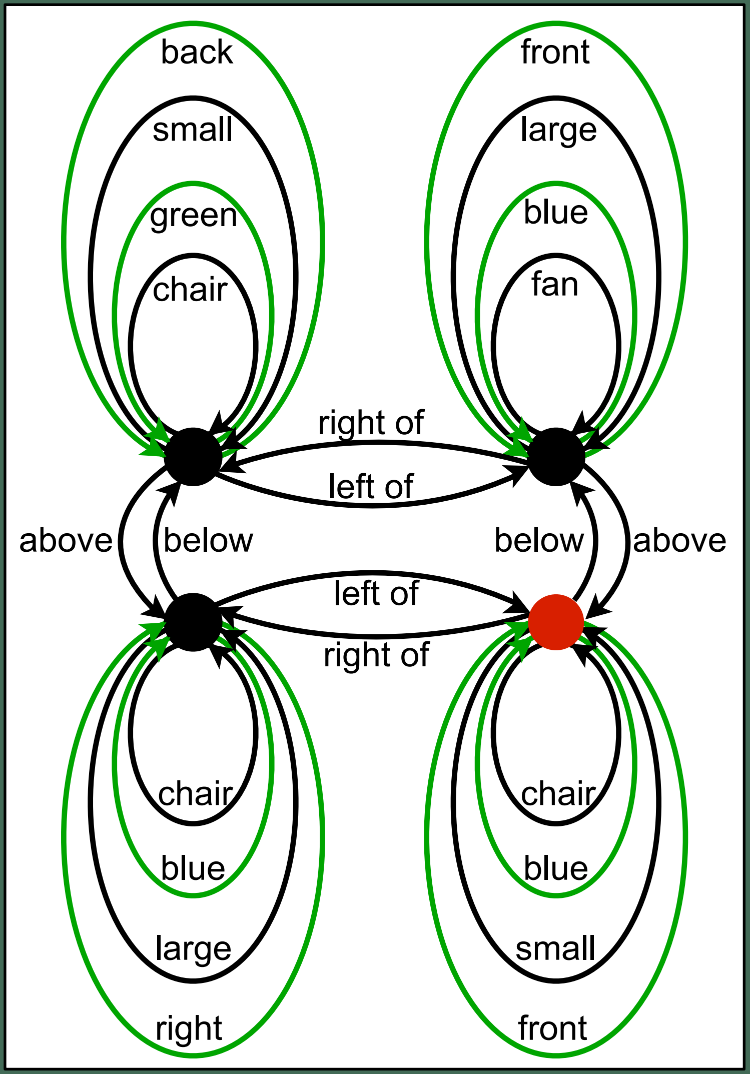
Outline

1) Referring Expression Generation
   - Redundancy in REG
2) The Graph-Based Algorithm
   - Cost Functions and Property Orderings
   - Redundancy in the Graph-Based Algorithm
3) The ASGRE TUNA Data
4) Tuning the Algorithm
   - Corpus-Based Costs and Free Properties
   - Frequent Properties Considered First
5) Evaluation
6) Conclusions and Future Work

The Graph-Based Framework for REG[1]



A scene is represented as a labelled directed graph.



A distinguishing description:

- is a *unique connected subgraph.*
- contains the node representing the target object.

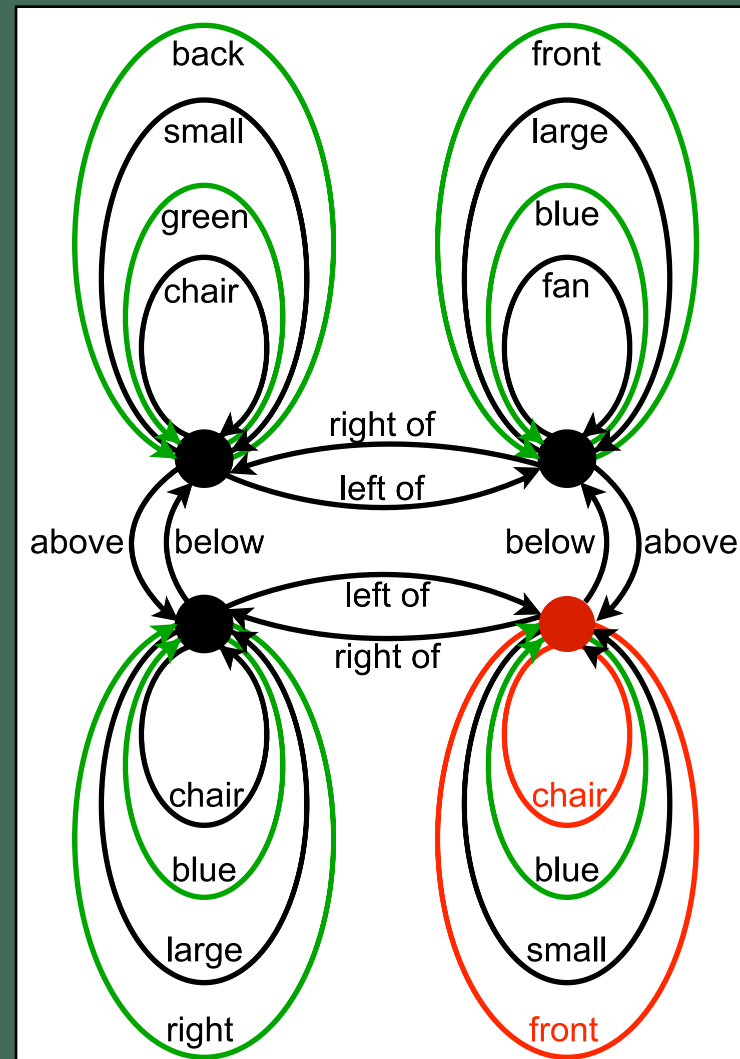
The algorithm:

- does a depth-first search.
- uses a cost function over the edges.
- returns the cheapest description.

[1]Krahmer et al. (2003) – CL

The Graph-Based Framework for REG[1]

A scene is represented as a labelled directed graph.



A distinguishing description:

- is a *unique connected subgraph.*
- contains the node representing the target object.

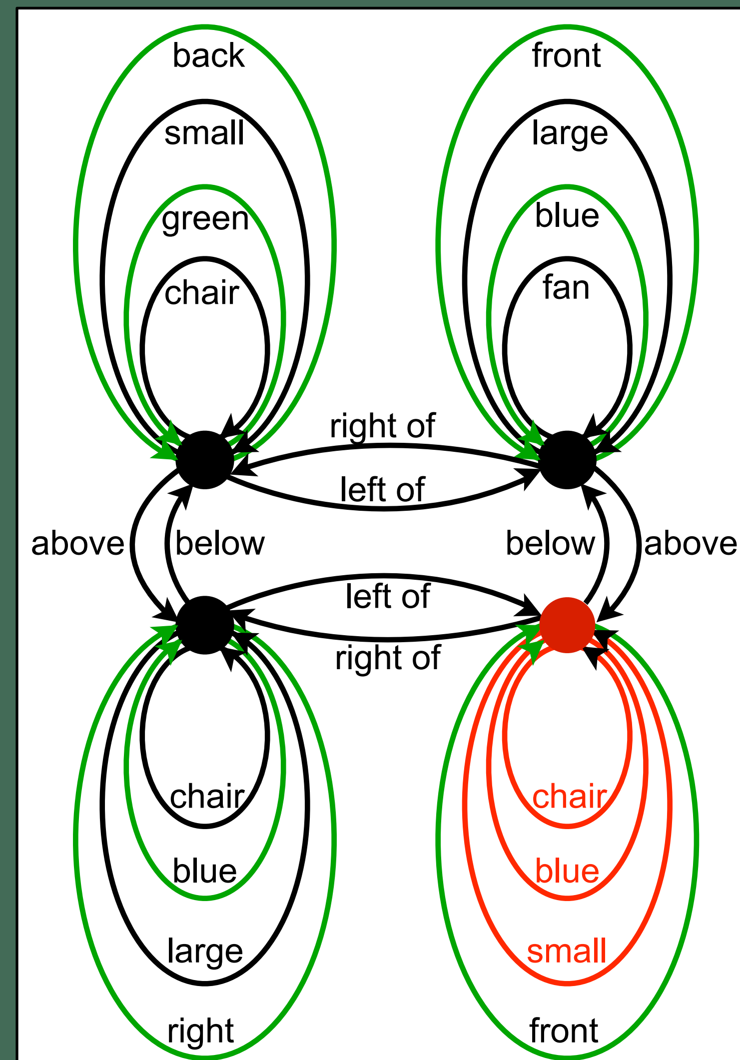
The algorithm:

- does a depth-first search.
- uses a cost function over the edges.
- returns the cheapest description.

[1]Krahmer et al. (2003) – CL

The Graph-Based Framework for REG[1]

A scene is represented as a labelled directed graph.



A distinguishing description:

- is a *unique connected subgraph.*
- contains the node representing the target object.

The algorithm:

- does a depth-first search.
- uses a cost function over the edges.
- returns the cheapest description.

[1]Krahmer et al. (2003) – CL

Graphs and Cost Functions

Two distinguishing descriptions:

1) The front-facing chair.
2) The small blue one.

Three cost functions:

cost function	CHAIR	FRONT	SMALL	BLUE
#1	1	12	11	11
#2	1	12	2	3
#3	1	4	2	3

cost function #1 ⟶ **cost(1) = 13**, cost(2) = 22

cost function #2 ⟶ cost(1) = 13, **cost(2) = 5**

cost function #3 ⟶ cost(1) = cost(2) = 5

Property Orderings in the Graph-Based Algorithm

Determine which description is found first:

1) The front-facing chair.
2) The small blue one.

cost function #3: cost(1) = cost(2) = 5

Property Ordering 1: [CHAIR, SMALL, FRONT, BLUE]

⟶ The front-facing chair is chosen.

Property Ordering 2: [SMALL, CHAIR, BLUE, FRONT]

⟶ The small blue one is chosen.

Redundancy in the Graph-Based Framework

- The cost function is required to be monotonic increasing ...
- ... but properties can be free.

1) The front-facing chair.
2) The *blue* front-facing chair.

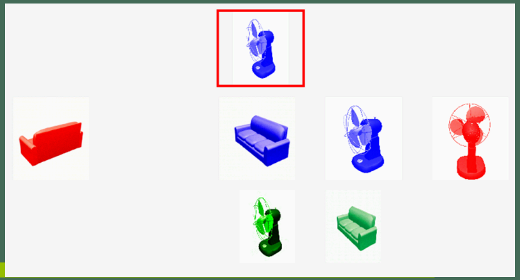
Description (2) will be returned only if:

- cost(BLUE) = 0 and
- description (2) is found first.

Outline

1) Referring Expression Generation
   - Redundancy in REG
2) The Graph-Based Algorithm
   - Cost Functions and Property Orderings
   - Redundancy in the Graph-Based Algorithm
3) The ASGRE TUNA Data
4) Tuning the Algorithm
   - Corpus-Based Costs and Free Properties
   - Frequent Properties Considered First
5) Evaluation
6) Conclusions and Future Work

The ASGRE TUNA Data[1]



- Largest data set of human-produced referring expressions.
- Two domains:

	Furniture	People
development / test	80	68
training	239	206
# of properties	6	12

- Stochastic costs derived from frequency counts:

Property	# in desc	P(v)	$-\log_2(P(v))$	cost
colour	211	0.88	0.18	2
orientation	84	0.35	1.51	15
size	86	0.36	1.47	15
x-dimension	48	0.20	2.32	23
y-dimension	64	0.27	1.90	19

[1]http://www.csd.abdn.ac.uk/research/evaluation

Tuning the Graph-Based Algorithm

Cost Functions:

- *Simple Costs*: All properties cost 1.
- *Stochastic Costs*: Costs determined by frequency counts.
- *Free–Stochastic*: Most frequent properties are free, rest stochastic.
- *Free–Naïve*: Most frequent properties are free, least frequent cost 2, rest cost 1.

Property Orderings:

- *Random Order (baseline)*
- *Free Properties First*

⟶ 8 Conditions to be tested.

Outline

1) Referring Expression Generation
    - Redundancy in REG
2) The Graph-Based Algorithm
    - Cost Functions and Property Orderings
    - Redundancy in the Graph-Based Algorithm
3) The ASGRE TUNA Data
4) Tuning the Algorithm
    - Corpus-Based Costs and Free Properties
    - Frequent Properties Considered First
5) Evaluation
6) Conclusions and Future Work

ASGRE 2007 Evaluation Metrics

- ASGRE 2007: first challenge on Attribute Selection for Generating Referring Expressions.
- *DICE* coefficient of set similarity:
  - $DICE(A,B) = \frac{(2 \times |A \cap B|)}{|A| + |B|}$
  - Perfect match: $A = B \rightarrow DICE(A,B) = 1$
  - No overlap: $| A \cap B | = 0 \rightarrow DICE(A,B) = 0$
- *PRP* (Perfect Recall Percentage):
  proportion of DICE scores of 1.

Results

Random Order Baseline:		Furniture		People	
	Cost function	DICE	PRP	DICE	PRP
	Simple Cost	0.550	2.5	0.606	17.6
	Stochastic	0.658	18.8	0.625	17.6
	Free-Stoch	0.701	27.5	**0.665**	16.2
	Free-Naïve	**0.757**	**33.8**	0.647	**19.1**

Free Properties First:		Furniture		People	
	Cost function	DICE	PRP	DICE	PRP
	Simple Cost	0.597	12.5	0.569	17.7
	Stochastic	0.658	21.3	0.625	17.7
	Free-Stoch	0.775	46.3	**0.689**	**25**
	Free-Naïve	**0.796**	**50**	0.639	20.6

Outline

1) Referring Expression Generation
   - Redundancy in REG
2) The Graph-Based Algorithm
   - Cost Functions and Property Orderings
   - Redundancy in the Graph-Based Algorithm
3) The ASGRE TUNA Data
4) Tuning the Algorithm
   - Corpus-Based Costs and Free Properties
   - Frequent Properties Considered First
5) Evaluation
6) Conclusions and Future Work

Conclusions

- Corpus-based content selection makes sense for REG.
- Allowing redundancy better matches human data.
- Combination of cost functions and property orderings (to increase redundant properties) boosts performance.
- Free–Naïve does better on Furniture than on People because the People domain has more properties, so more information is lost.

Future Work

- Handling the (redundant) use of relations between objects in the graph-based framework.
- Extending the framework to allow dynamic cost functions and property orderings.
- Performing human task-based evaluation.
- Collecting larger or more specified data sets.
- Developing different evaluation metrics.