DMRS

Overview and current work

Ann Copestake

Natural Language and Information Processing Group
Computer Laboratory
University of Cambridge

July 2010

Outline

Outline

Dependency MRS: an introduction

DMRS packing and comparison

Inducing systematic semantic relationships

Conclusions

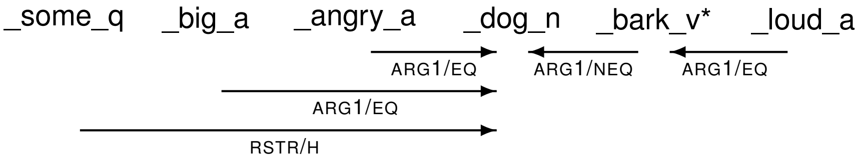
Semantic dependency representations

- Oepen: MRS elementary dependencies, a partial representation. Treebanking, features for parse ranking.
- Dependency MRS (DMRS) goals:
  - predicates with simple inventory of links, no variables
  - all information is retained so interconvertible with MRS (one-to-one mapping)
  - structure is minimal (no redundancy)
  - applicable to different grammars, robust to changes in grammars
- No direct logical interpretation.

Semantic dependency representations

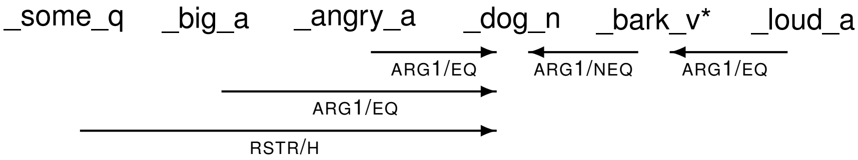
- Oepen: MRS elementary dependencies, a partial representation. Treebanking, features for parse ranking.
- Dependency MRS (DMRS) goals:
  - predicates with simple inventory of links, no variables
  - all information is retained so interconvertible with MRS (one-to-one mapping)
  - structure is minimal (no redundancy)
  - applicable to different grammars, robust to changes in grammars
- No direct logical interpretation.

DMRS



l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

DMRS



l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

Characteristic variables

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

_some_q(x4,_big_a(e8,x4) ∧ _angry_a(e9, x4) ∧_dog_n(x4),
_bark_v(e2,x4) ∧_loud_a(e10,e2))

RMRS: EPs may have a distinguished argument.
Characteristic variable property: the distinguished argument of an RMRS EP (arg0) is unique to it (NB: not arg0 of quantifiers, so for simplicity here, use BV).
Introduced into DELPH-IN grammars for grammar-internal reasons.

Characteristic variables

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

_some_q(x4,_big_a(e8,x4) ∧ _angry_a(e9, x4) ∧_dog_n(x4),
_bark_v(e2,x4) ∧_loud_a(e10,e2))

RMRS: EPs may have a distinguished argument.
Characteristic variable property: the distinguished argument of an RMRS EP (arg0) is unique to it (NB: not arg0 of quantifiers, so for simplicity here, use BV).
Introduced into DELPH-IN grammars for grammar-internal reasons.

Characteristic variables

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

_some_q(x4,_big_a(e8,x4) ∧ _angry_a(e9, x4) ∧_dog_n(x4),
_bark_v(e2,x4) ∧_loud_a(e10,e2))

RMRS: EPs may have a distinguished argument.
Characteristic variable property: the distinguished argument of an RMRS EP (arg0) is unique to it (NB: not arg0 of quantifiers, so for simplicity here, use BV).
Introduced into DELPH-IN grammars for grammar-internal reasons.

Adjectives and characteristic variables

- Use (and misuse) of event variables: e.g., Hobbs (1985), Asher (1993), Maienborn (2005).
- Long-standing use of event variables on adjectives in DELPH-IN grammars.
- Predicative uses without copula in semantics, tense as a property of the event variable.

  (1) She was angry.

  (2) pron(x), angry(e_{past}, x)

- Attributive adjective temporal modification in German.

  (3) Der im Fruehling gruene Rasen ist jetzt braun und ausgetrocknet.
  The in spring green lawn is now brown and dried-out.

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

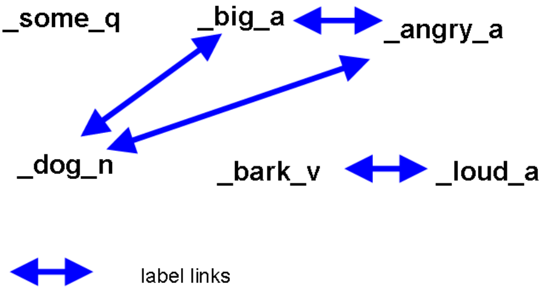
1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

RMRS to DMRS: RMRS graphs

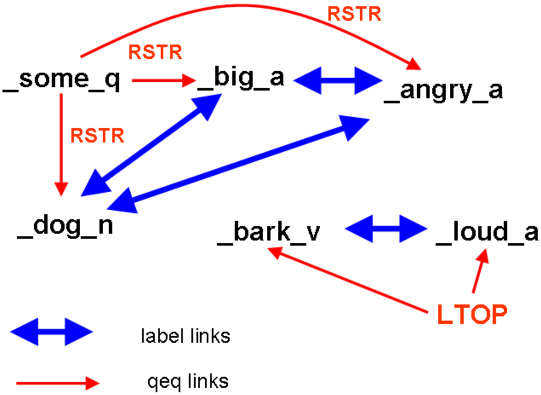
l1:a1:_some_q, BV(a1,x4), RSTR(a1,h5), BODY(a1,h6),
h5 qeq l2,
l2:a2:_big_a(e8), ARG1(a2,x4),
l2:a3:_angry_a(e9), ARG1(a3,x4),
l2:a4:_dog_n(x4),
l4:a5:_bark_v(e2), ARG1(a5,x4),
l4:a6:_loud_a(e10), ARG1(a6,e2)

1. label equality: EPs with equal labels
2. qeq graph: scopal argument in EP to label
   ltop: label of one of more EPs
3. variable graph: non-scopal arguments to characteristic variables

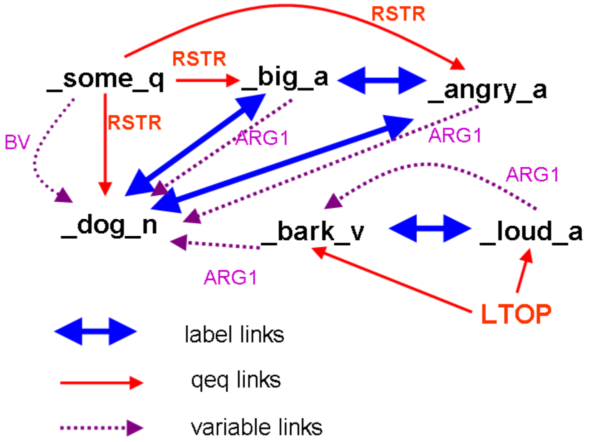
RMRS label equality graph



Label equality and qeq graph

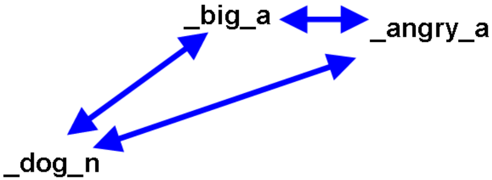


Label equality, qeq and variable graph



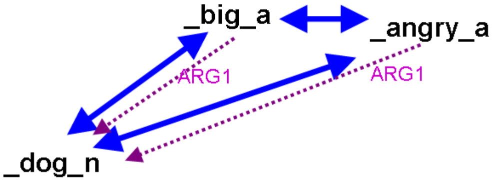
Redundant link problem

Label equalities give $n(n-1)/2$ binary links.



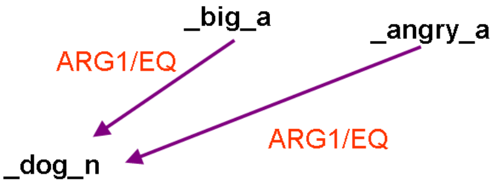
Variable links

Variable links relate an EP argument to a unique EP because of the characteristic variable property.



Merged links

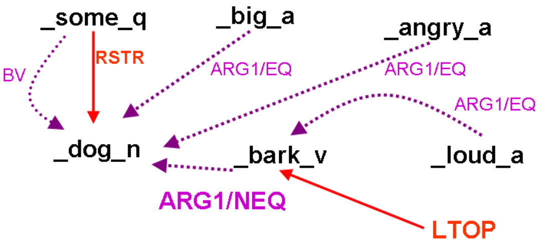
Use variable graph to decide on canonical links.



Selection of qeq/LTOP target

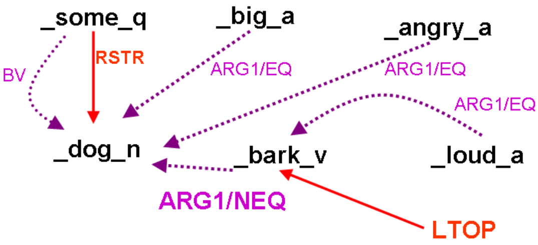
- qeq and LTOP point to labels, so how to select a unique target node from EPs with that label?
- Syntactic head: unique, intuitive.
- Syntactic head without syntax:
  - either mirror variable graph (esp. quantifier RSTR, mirrors BV (ARG0))
  - or EP with no argument EPs in equal label set (i.e., not modifier)
- Choice of LTOP uses the second principle.

Merged links on full graph



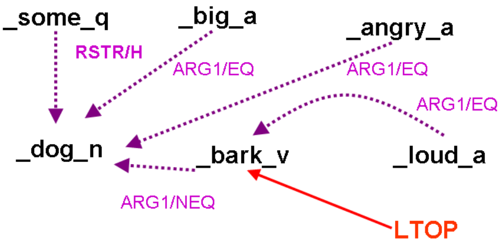
- RSTR and BV always parallel, so remove BV.

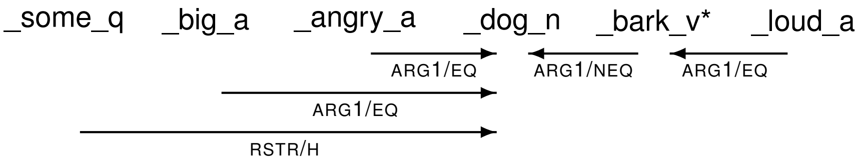
Merged links on full graph



- RSTR and BV always parallel, so remove BV.

DMRS


_some_q
_big_a
_angry_a
RSTR/H
ARG1/EQ
ARG1/EQ
ARG1/EQ
_dog_n
_bark_v
_loud_a
ARG1/NEQ
LTOP


_some_q _big_a _angry_a _dog_n _bark_v* _loud_a
ARG1/EQ
ARG1/NEQ
ARG1/EQ
ARG1/EQ
RSTR/H

More on link selection

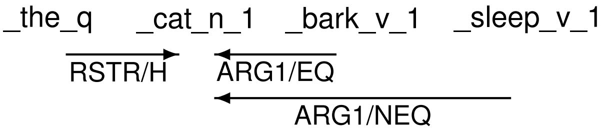
Links reflect syntax without syntax being used in RMRS-to-DMRS conversion:

- Intersective modification (and some PP-arguments) normally gives merged ARG/EQ links because of the variable graph.
  - Undirected /EQ links needed for modification without an argument relation to head (e.g., some relative clauses).
- NP arguments result in ARG/NEQ links, because quantifiers float.
- Scopal arguments give ARG/H link to syntactic head of items with equal labels (also LTOP).

Semantics of relative clauses

Two pieces of semantics associated with relative clause attachment:

1. Modified noun as filler of gap in the relative clause.
2. Relative clause conjoined with noun (hence part of quantifier RSTR).



Relative clauses and the EQ link

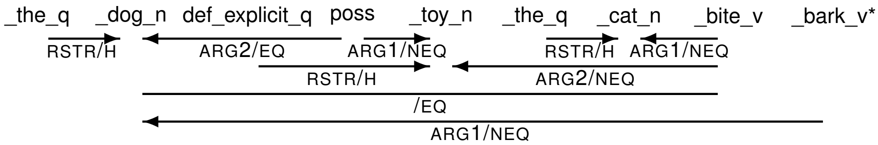
who the cat bit: gap is in main verb of relative clause.

$[\ l, e\ ]\ \{[\ l, y\ ]_{\mathrm{mod}}\}\ [\ \mathrm{cat}(z), l{:}\mathrm{bite}(e,z,y)\ ]$

whose toy the cat bit: gap not in main verb of rel. clause

$[\ l, e\ ]\ \{[\ l, x\ ]_{\mathrm{mod}}\}\ [\ \mathrm{poss}(x,y), \mathrm{toy}(y), \mathrm{cat}(z), l{:}\mathrm{bite}(e,z,y)\ ]$

The dog whose toy the cat bit barked.



Relative clauses and the EQ link

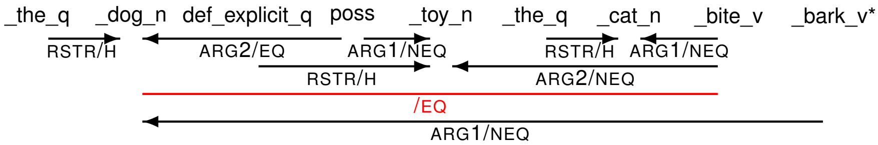
who the cat bit: gap is in main verb of relative clause.

$[\ l, e\ ]\ \{[\ l, y\ ]_{mod}\}\ [\ cat(z), l{:}bite(e,z,y)\ ]$

whose toy the cat bit: gap not in main verb of rel. clause

$[\ l, e\ ]\ \{[\ l, x\ ]_{mod}\}\ [\ poss(x,y), toy(y), cat(z), l{:}bite(e,z,y)\ ]$

The dog whose toy the cat bit barked.



Outline

Dependency MRS: an introduction

DMRS packing and comparison

Inducing systematic semantic relationships

Conclusions

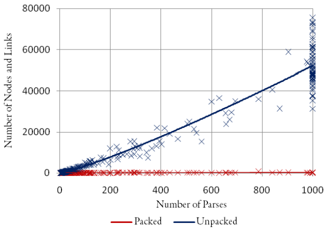
Packed DMRS

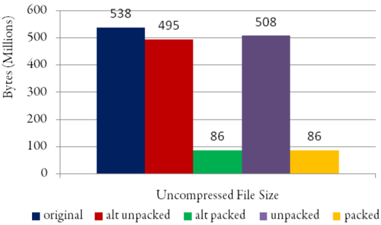
- DMRS is represented by set of nodes and set of links.
- Packed DMRS: shared nodes and links with associated ids (e.g., parse number).
- Easier than packing (R)MRS because no variables, so no variable (re)naming.
- Vaughan Eveleigh (Cambridge MPhil project): implement packing and exploit in DMRS comparison.
- `wiki.delph-in.net/moin/RmrsDmrsComparison`
  `http://code.google.com/p/cstitproject/`

DMRS comparison

- Identity: all nodes and links the same.
- Comparison: pair identical (comparable) nodes and their links and record in a data structure that can be used/rendered in various ways.
- Efficiency depends on sorting. Works best with data from the same utterance, because of character position.
- Tested for parser version comparsion on hike with up to 1000 parses (plus csli and vm with up to 5 parses) comparing ERG 0909 and 1004.

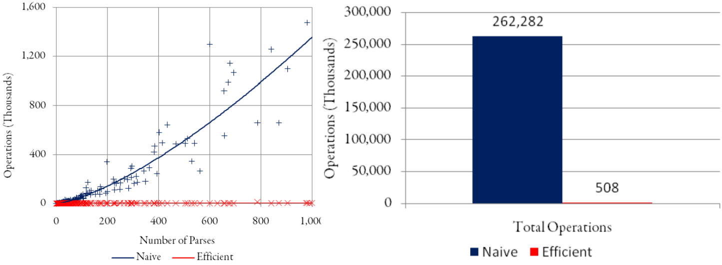
Runtime Improvements – File Size





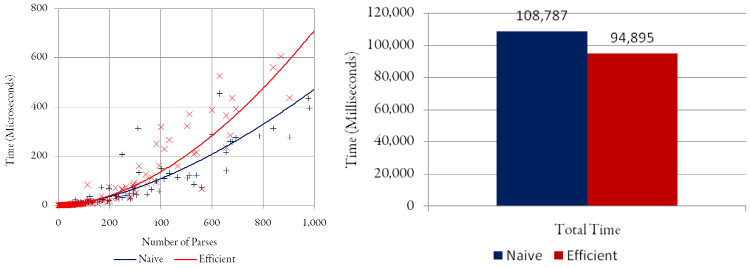
Difference in file size between representation types (Hike 0909)

Comparison Runtime Performance – Fundamental Operations



Number of fundamental operations using Naive and Efficient implementations of DMRS comparison

Comparison Runtime Performance – Time



Time required for Naive and Efficient implementations of DMRS comparison

Outline

The case for DMRS inference rules

Work by Andy MacKinlay, visiting Cambridge from Melbourne

- For tasks such as IE, compare two DMRS structures.
- Sometimes (more-or-less) the same:
  - *Hoffman synthesised aspirin*
  - *Aspirin was synthesised by Hoffman*
- But often quite different DMRSs from semantically similar sentences.
- A systematic way to map between these different DMRSs would be useful.
- Also paraphrase, summarization (cf RTE etc)

Similarity examples

- *synthesis of aspirin*
- *aspirin synthesis*
- *aspirin's synthesis*
- *synthesis for aspirin*
- NOT *synthesis from aspirin*
- *synthesized aspirin*: not for this study, just looked at relationships between two nominals.

Similarity examples

- *synthesis of aspirin*
- *aspirin synthesis*
- *aspirin's synthesis*
- *synthesis for aspirin*
- NOT *synthesis from aspirin*
- *synthesized aspirin*: not for this study, just looked at relationships between two nominals.

Similarity examples

- *synthesis of aspirin*
- *aspirin synthesis*
- *aspirin's synthesis*
- *synthesis for aspirin*
- NOT *synthesis from aspirin*
- *synthesized aspirin*: not for this study, just looked at relationships between two nominals.

Automatically constructed DMRS inference rules

- In general, there can be a large number of “light predicates” in a DMRS:
  - Construction predicates, or
  - Lexical predicates with relatively little semantic content
- We wish to find correspondences between different paths of such predicates.
- Map between structures, keeping DMRSs well-formed.
- Experiment with “anchor text”

Anchor Text

- Constrained 'extended distributional hypothesis' (Lin and Pantel, 2001):
  - If two paths tend to occur in similar contexts, the meanings of the paths tend to be similar.
- In a corpus of DMRSs, if we frequently see the same noun pair as endpoints to different paths, the paths may be related.
- e.g., If we frequently see two nouns A and B (eg *aspirin* and *synthesis*) connected by two different paths of light predicates X and Y, there is evidence for a correspondence between X and Y.
- If large number of overlapping endpoint pairs, good evidence for correspondence.

Algorithm for finding correspondences

- Parse a corpus with the ERG, outputting as DMRS
- Find all paths in all DMRSs connecting two nouns.
- Decompose each path found which fulfils certain criteria into a tuple (N_1, G, N_2), where G is an ‘abstract subgraph‘ representing the path found.
- Add (N_1, N_2) to the set of endpoint pairs found for subgraph G
- From the table of subgraphs and attested endpoint contexts, calculate a correspondence score for each possible subgraph pairing, using the overlap of contexts.

Scoring rule correspondences

- We don't expect all correspondence pairs to be equally useful – some relationships may be weak
- We tried several scoring metrics, all based on the number of overlapping noun endpoint pairs:
  - RAW Raw number of overlapping matches, scaled to (0, 1)
  - IDFRAW Multiply raw counts by the inverse document frequency of each endpoint noun, as rare terms are clearer indicators
  - PAIRIDFRAW Multiply raw counts by the inverse document frequency of pair of endpoints.
  - JACC Jaccard coefficient over the sets *A* and *B* of endpoints attested with each subgraph.

Evaluation

- QA based evaluation:
  - If a test DMRS shares a (hypothetically) related subgraph (between similar nouns), boost the score.
- But no suitable QA system, so tried paraphrase:
- For correspondence rules with scores above some threshold, apply the mappings to a test corpus:
  - Look for subgraphs in a test corpus that match the LHS of a rule
  - Replace them with the RHS of the rule
  - Convert to MRS and generate
- Tests well-formedness (but perhaps too strict)

Corpora

- For training we want corpora to be in a single domain and reasonably large.
- High quality parse trees are useful although not required
- Results here for WeScience ($\sim$ 10000 Wikipedia sentences) with hand-selected gold trees.
- Not reported: LOGON, and WeScience with auto-selected trees.
- Different domain for test. Parsed every 1000th sentence of the BNC and discarded sentences longer than 12.

Not Exactly Spectacular Results

Impenetrable table of numbers

		Number of Rules		
Metric	Thresh	Learnt	Matched	Gen'd
Jacc	0.006	6406	1184	35 (0.5%)
Jacc	0.008	5707	1155	32 (0.6%)
Jacc	0.010	4362	1150	31 (0.7%)
PairIDF	0.020	4696	1171	*153 (3.3%)
PairIDF	0.040	874	250	*47 (5.4%)
PairIDF	0.060	406	109	*20 (4.9%)
IDF	0.030	884	288	*61 (6.9%)
IDF	0.040	496	177	*40 (8.1%)
IDF	0.050	240	85	*22 (9.2%)
IDF	0.060	176	63	*16 (9.1%)

Samples of generation

Generation Samples

- ***The authors state that citation counts indicate impact rather than quality.***
  - *The authors state the counts of citations indicate impact rather than quality.*
  - *The authors state the count of the citations indicates impact rather than quality.*
  - *The authors state that counts of citations indicate impact rather than quality.*
  - *The authors state the count of some citation indicates impact rather than quality.*
- ***Doc Threadneedle leaned over and kissed her.***
  - *Threadneedle, a doc, leaned over and kissed her.*
  - *Threadneedle the docs leaned over and kissed her.*

Analysis, a.k.a Lessons Learnt

- Only a small percentage generate, but that doesn't necessarily mean all of the rest are useless (although some clearly are!).
- Not all of the generated sentences look good.
- Treatment of determiners was quite complex.
- Don't know whether this would be useful for QA yet.
- Learning curve expected to flatten off more: rule-learning needs to be tweaked.

Outline

DMRS for evaluation

(4) Not all those who wrote opposed the proposal.

PARC	pron form(pro3, those) adjunct(pro3, write) adjunct type(write, relative) pron form(pro4, who) pron type(pro4, relative) pron rel(write, pro4) topic rel(write, pro4)
GR	(cmod who those wrote) (ncsubj wrote those)
Stanford	nsubj(wrote, those) rel(wrote, who) rcmod(those, wrote)

MRS treatment uses several construction predicates: ‘those people who wrote’.
No predicate from relative clause *who* because of reduced relatives *the people consulted objected*.

Conclusions

- DMRS shares benefits of tractability with elementary dependencies, but complete (apart from uninstantiated optional arguments).
- Hence, we can replace MRSs with DMRSs in many contexts.
  - Direct DMRS composition (producing packed DMRS?)
  - New forms of underspecification.
  - Integration with distributional techniques.
  - Manual annotation of unparsed items (via fix up of partial/incorrect structures).
- Theoretical interest?