NLP Tools
LVG - Derivations
(SD-Rules)

By: Dr. Chris Lu

The Lexical Systems Group

NLM. LHNCBC. CGSB

June, 2013

Table of Contents

Introduction - NLP

- Natural Language (English)
  - is ordinary language that humans use naturally
  - may be spoken, signed, or written

- Natural Language Processing
  - NLP is to process human language to make their information accessible to computer applications
  - The goal is to design and build software that will analyze, understand, and generate human language
  - Most NLP applications require knowledge from linguistics, computer science, and statistics

NLP Example

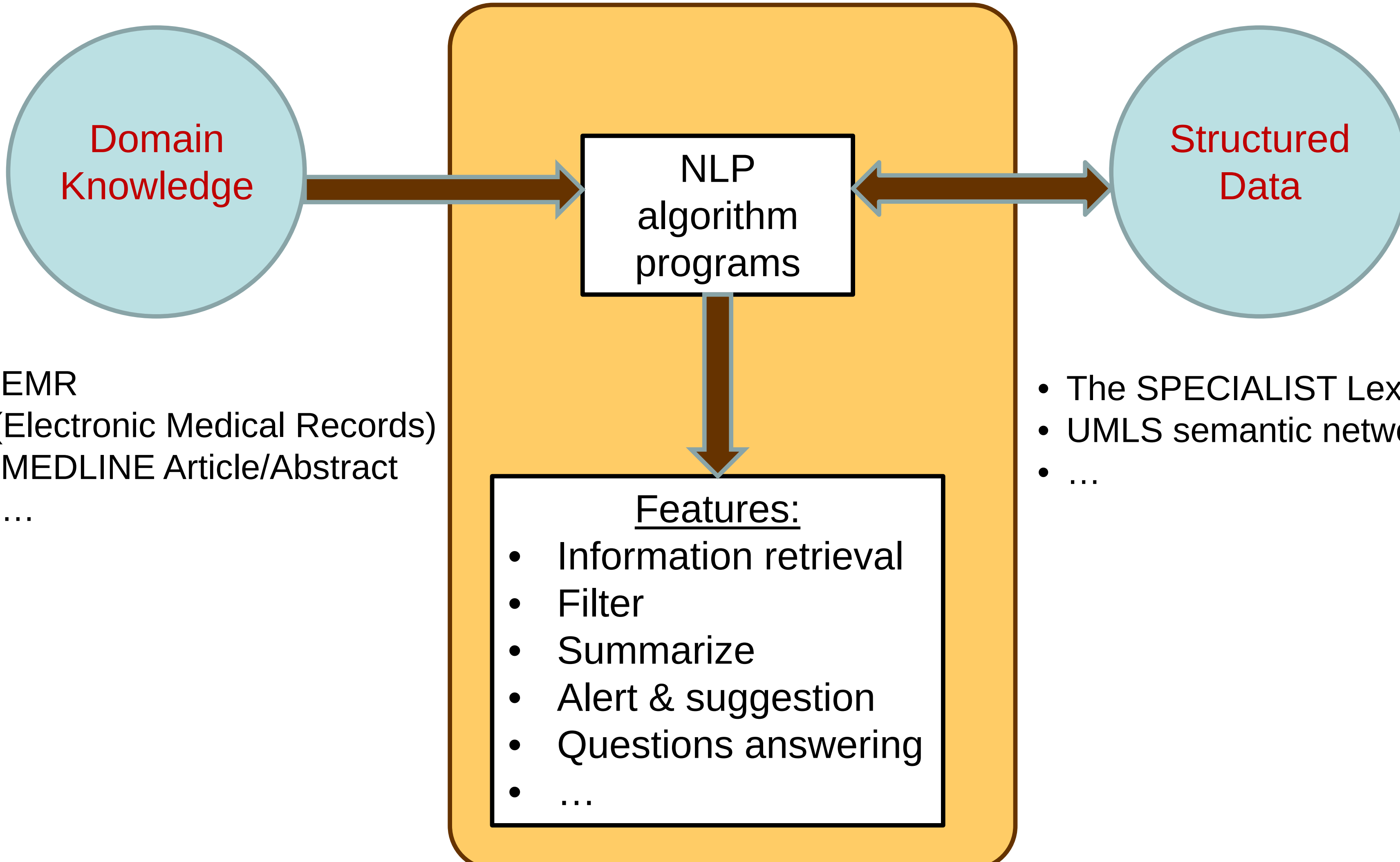
Questions
Symptoms

NLP System

Features:

- Information retrieval
- Filter
- Summarize
- Alert & suggestion
- Questions answering
- …

NLP System



- EMR
  (Electronic Medical Records)
- MEDLINE Article/Abstract
- …

- The SPECIALIST Lexicon
- UMLS semantic network
- …

NLP – Concepts

- UMLS (**Unified Medical Language System)**
  - is a comprehensive thesaurus and ontology of biomedical concepts
  - created in 1986 by NLM
  - provides terms to concepts mapping from different controlled vocabularies sources, such as ICD-10, MeSH, SNOMED CT, etc.
  - includes:
    - Metathesaurus
    - Semantic Network
    - SPECIALIST Lexicon and Lexical Tools

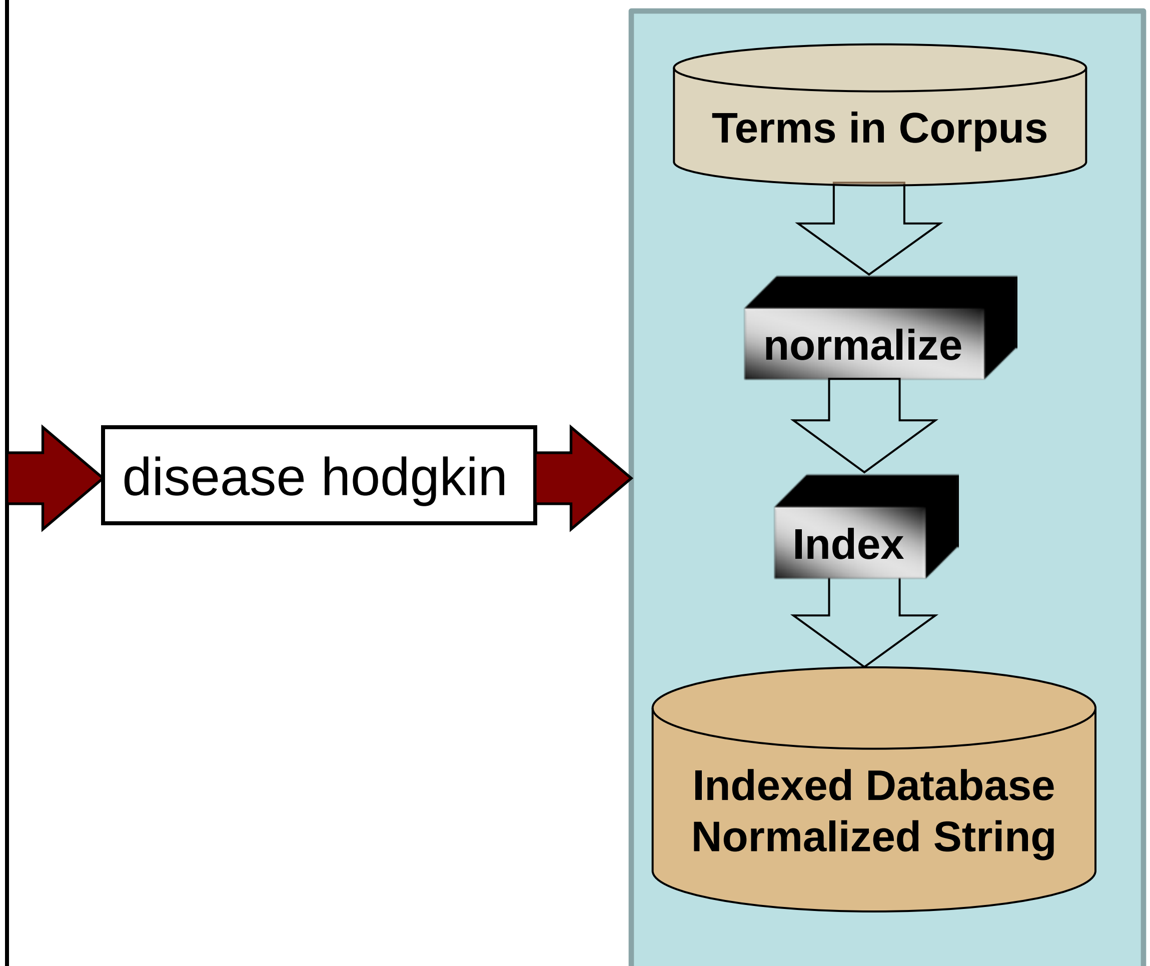
Challenge in Concept Mapping

- Terms have multiple concepts
  - Example: cold (7 CUIs in UMLS-2013AA)
    - Cold Temperature
    - Common Cold
    - Cold Therapy
    - Cold Sensation
    - etc..
  - ➢ Word Sense Disambiguation (WSD)

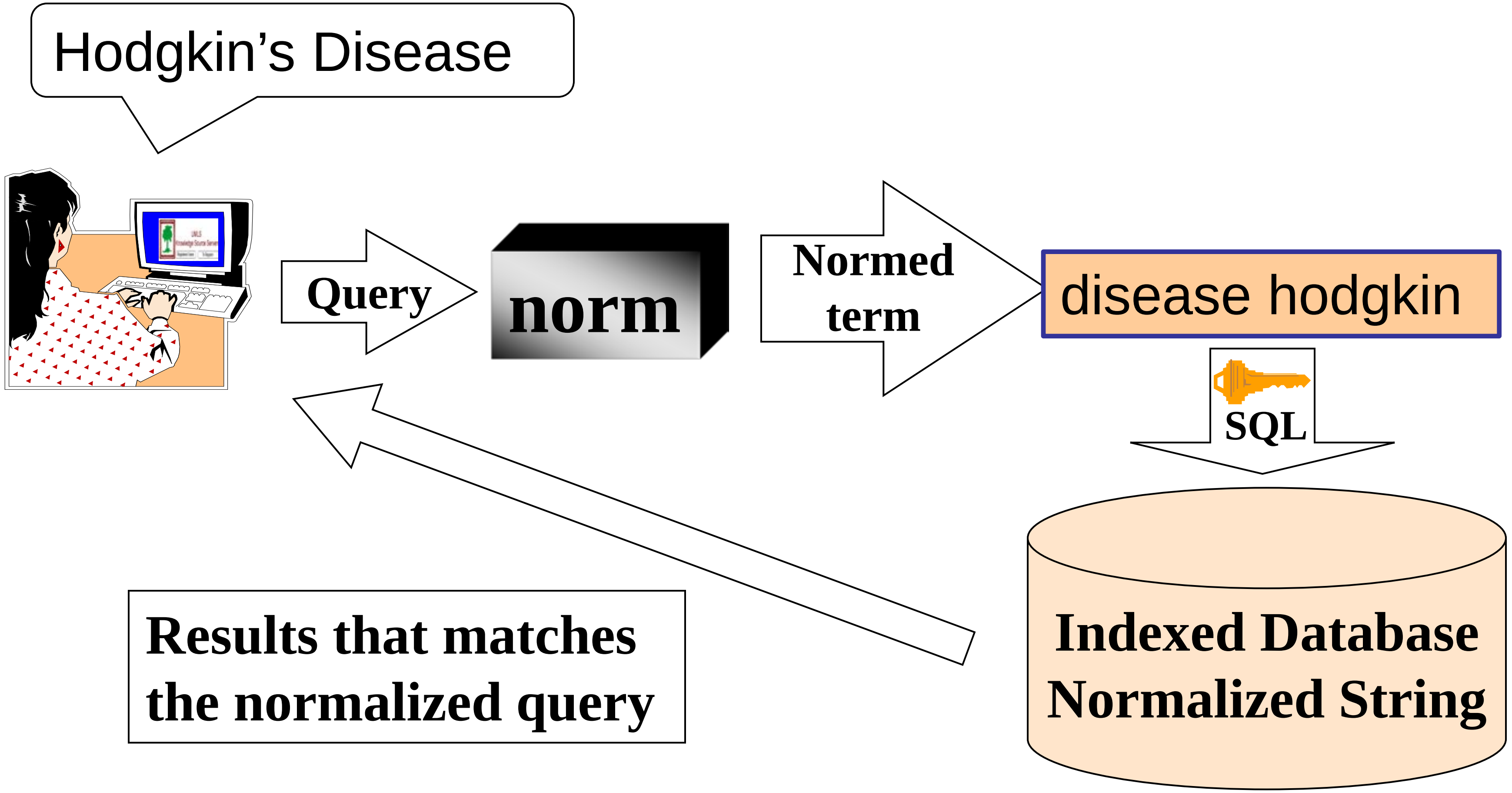
- Concepts has variety of ways to express
  - Example: Hodgkin's Disease
  - ➢ Normalization
  - ➢ Query Expansion

NLP - Norm

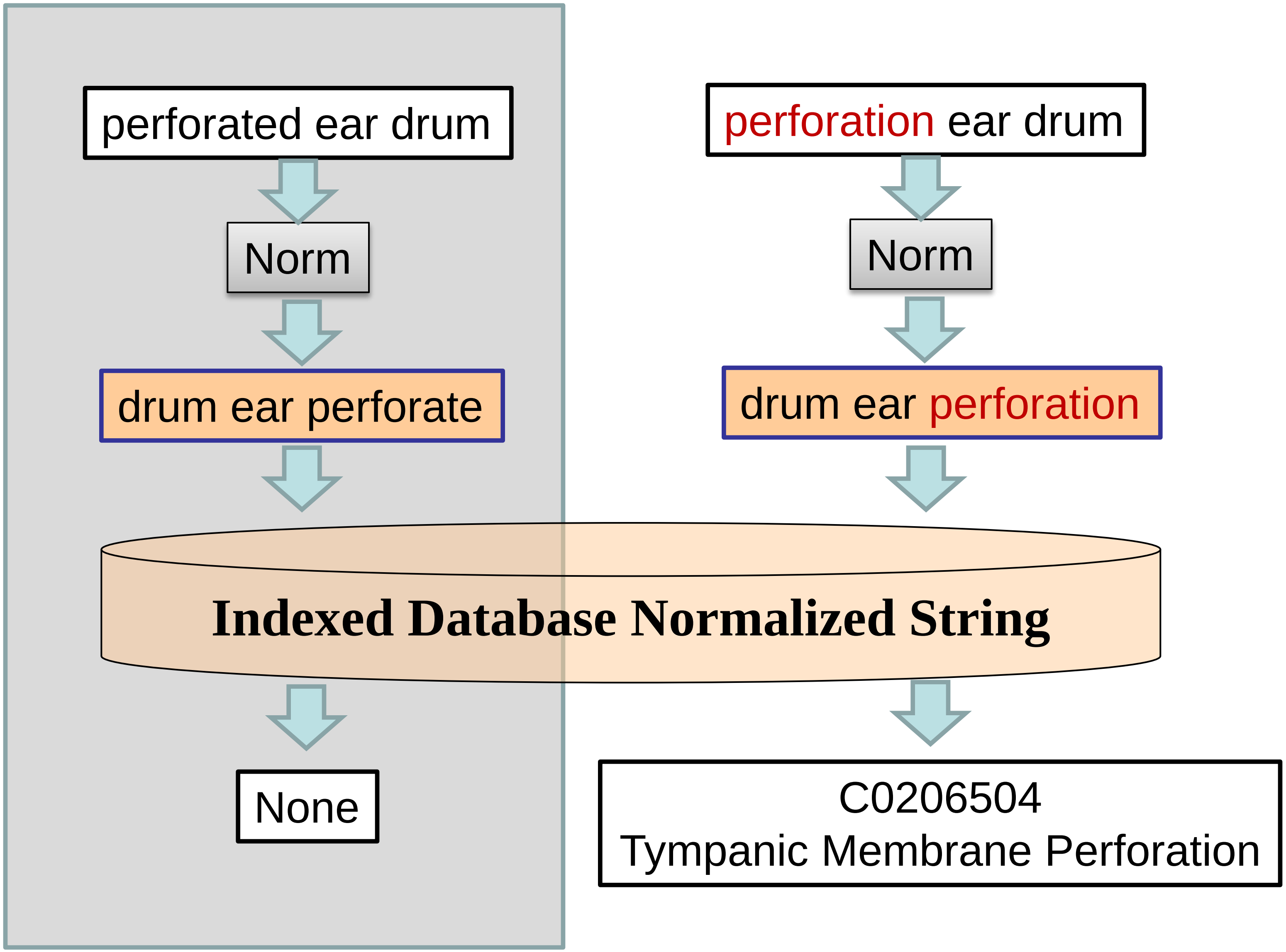
- Hodgkin Disease
- HODGKINS DISEASE
- Hodgkin's Disease
- Disease, Hodgkin's
- HODGKIN'S DISEASE
- Hodgkin's disease
- Hodgkins Disease
- Hodgkin's disease NOS
- Hodgkin's disease, NOS
- Disease, Hodgkins
- Diseases, Hodgkins
- Hodgkins Diseases
- Hodgkins disease
- hodgkin's disease
- Disease;Hodgkins
- Disease, Hodgkin
- …



NLP - Norm


Hodgkin’s Disease
Query
norm
Normed
term
disease hodgkin
SQL
Indexed Database
Normalized String
Results that matches
the normalized query

NLP – Query Expansion



Lexical Variants

- To increase recall & precision

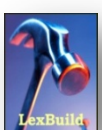
	Query expansion (Recall)	**POS Tagging (Precision)**
Inputs	perforated ear drum	saw
UMLS-CUI	None	• C1947903|See • C0183089|saw (device)
Process	perforation ear drum	noun
UMLS-CUI	C0206504	• C0183089
Preferred term	Tympanic Membrane Perforation	saw (device)

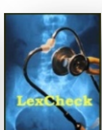
NLP Core Tasks

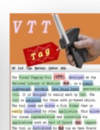
Example: Information retrieval (search engine)

- Tokenize & tagging (entity recognition)
  - break inputs into words <Text Tools, wordInd>
  - POS tagging <dTagger>
  - Other annotation <Visual Tagging Tool, VTT>
- spelling check
  - suggest correct spelling for misspelled words <gSpell>
- lexical variants (normalization/query expansion)
  - spelling variants, inflectional/uninflectional variants, synonyms, acronyms/abbreviations, expansions, derivational variants, etc. <Lexical Tools, LexAccess, LexCheck, STMT>
- semantic knowledge (concept mapping)
  - map text to Metathesaurus concepts <MetaMap, MMTX, STMT>
  - Word Sense Disambiguation <TC – StWSD>

NLP Tools by LSG

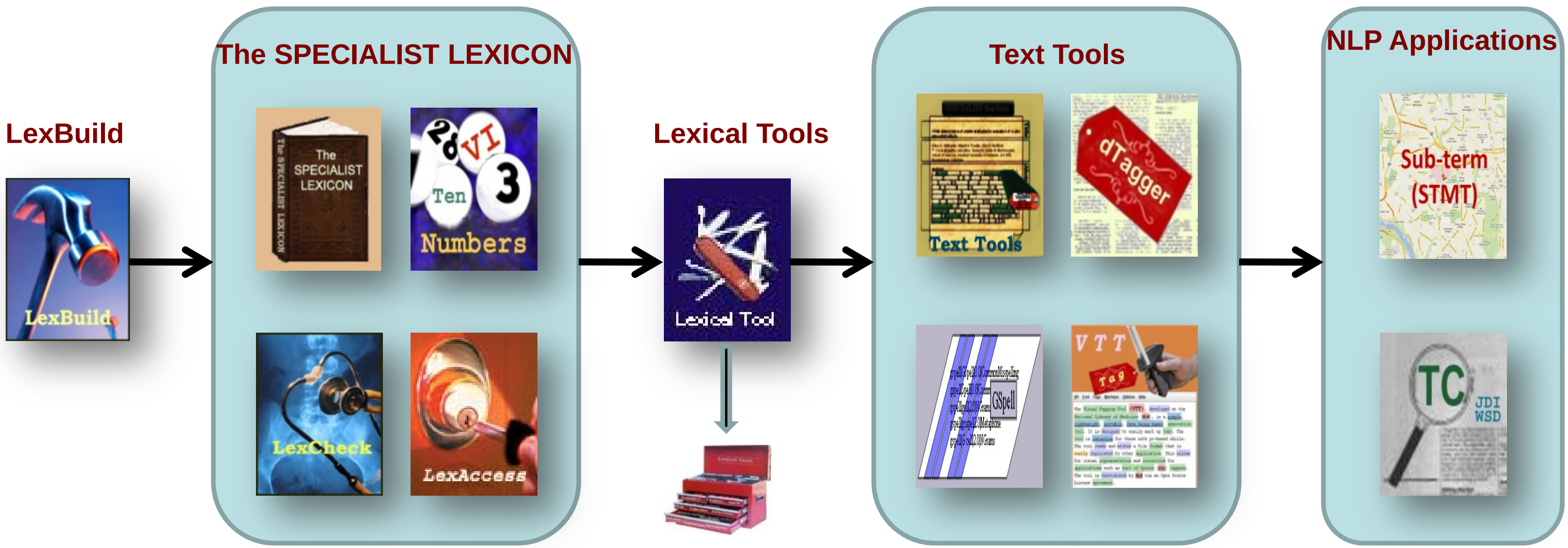




- Lexical Systems Group: http://umlslex.nlm.nih.gov
- The SPECIALIST NLP Tools: http://specialist.nlm.nih.gov

The SPECIALIST NLP Tools



- Lexical Systems Group: http://umlslex.nlm.nih.gov
- The SPECIALIST NLP Tools: http://specialist.nlm.nih.gov

Derivational Variants

- Words are related by a derivational process
  - Used to create new words based on existing words
  - Meaning change (related)
  - Category may change
  - Derivational process: suffix, prefix, and conversion

- Focus on relatedness (no direction)

Derivation Types (-kdt)

- Example (kind|adj):
  - zeroD: kind|adj|kind|noun
  - prefixD: kind|adj|unkind|adj
  - suffixD: kind|adj|kindly|adv

Derivational Pair

- Each link and the associated two nodes in derivational network define a derivational pair
- Includes base forms and syntactic category information
- Bi-directional
- Only involves one or none derivational affix
- Lvg format: base 1|category 1|base 2|category 2
- Examples:
  - kind|adj|kindness|noun
  - kind|adj|kindly|adv
  - kind|adj|unkind|adj
  - kind|adj|kind|noun

Derivations in LVG

- 7 flow components (62):
  - -f:d
  - -f:dc
  - -f:R
  - -f:G
  - -f:Ge
  - -f:Gn
  - -f:v
- 3 flow specific options (39):
  - -kd: 1|2|3 (default: 1)
  - -kdn: B|N|O (default: O)
  - -kdt: Z|S|P (default: ZSP)

LVG - Derivation Examples

- shell> lvg –f:d –p –SC –SI

```
- Please input a term (type "Ctl-d" to quit) >
hyperuricemic

hyperuricemic|hyperuricemic|<noun>|<base>|d|1|
hyperuricemic|hyperuricemia|<noun>|<base>|d|1|
hyperuricemic|hyperuricemic|<adj>|<base>|d|1|
```

Derivations Generation

- Before 2011-, issues of precision and recall
- A new systematic approach to automatically generating derivational variants using LVG conjunction with Specialist Lexicon:

2012	2013	2014
prefixD & zeroD	suffixD	SD-Rules

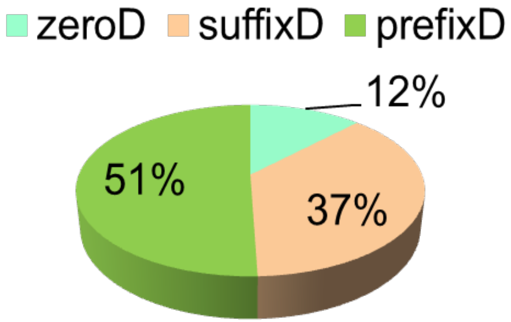
References:

- "A Systematic Approach for Automatically Generating Derivational Variants in Lexical Tools Based on the SPECIALIST Lexicon", Chris J Lu, Lynn McCreedy, Destinee Tormey, and Allen Browne, IEEE IT Professional Magazine, May/June, 2012, p. 36-42
- "Implementing Comprehensive Derivational Features in Lexical Tools Using a Systematical Approach", Chris J Lu, Lynn McCreedy, Destinee Tormey, and Allen Browne, AMIA 2013 Annual Symposium, Nov. 16-20, Washington, DC (submitted for publication)

Systematic Approach

- Better coverage:
  - Facts: cover all dPairs known to Lexicon (grow proportionally with Lexicon annually)

2011-	2012	2013
4,559	89,950	121,078



- Better precision:
  - Mainly relies on facts: virtually 100% accurate

- Derivations not in Lexicon?

Derivational Flow

- Facts
  - derivational pairs database table

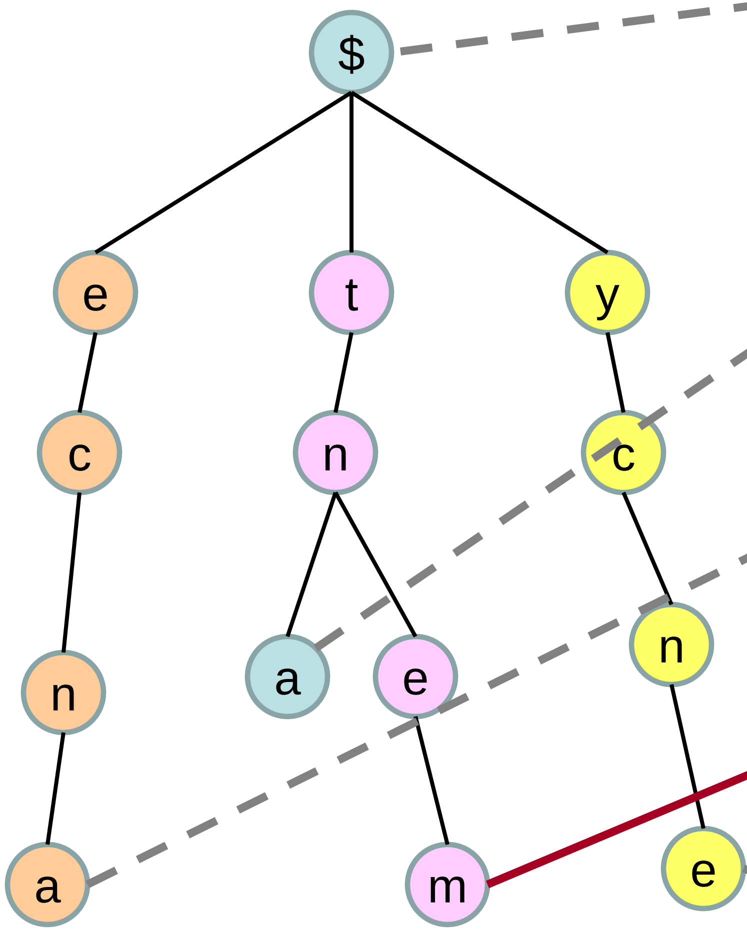
Base-1	Cat-1	EUI-1	Base-2	Cat-2	EUI-2	Negation	Type	prefix
…	…	…	…	…		…	…	…
care	noun	E0015334	**pre**care	noun	E0611704	O	P	**pre**
care	noun	E0015334	care**less**	adj	E0015344	N	S	None
care	noun	E0015334	care	verb	E0015335	O	Z	None
…	…	…	…	…	…	…	…	…

- SD-Rules
  - Use exceptions to increase precision

EXAMPLE: retirement|noun|retire|verb
RULE: ment$|noun|$|verb
EXCEPTION: apartment|apart;
EXCEPTION: basement|base;
EXCEPTION: department|depart;
…

SD-Rules (Trie)

- retirement|noun => retire|verb



EXAMPLE: retire|verb|retirement|noun
RULE: $|verb|ment$|noun
EXCEPTION: apart|apartment;
…

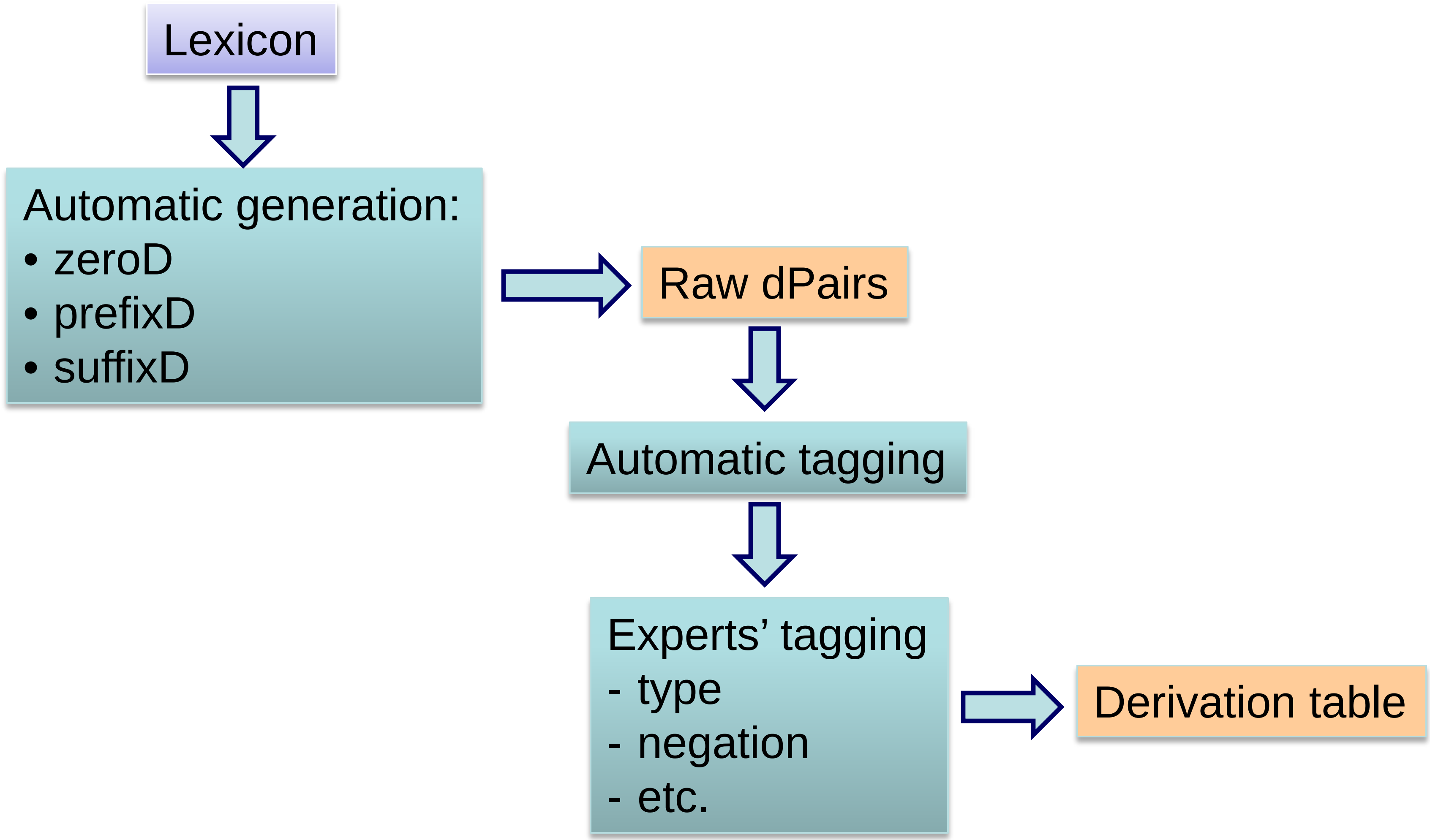
EXAMPLE: relaxant|adj|relax|verb
RULE: ant$|adj|$|verb
EXCEPTION: important|import;
…

EXAMPLE: conformant|adj|conformance|noun
RULE: ance$|noun|ant$|adj
EXCEPTION: ambulant|ambulance;
…

EXAMPLE: retirement|noun|retire|verb
RULE: ment$|noun|$|verb
EXCEPTION: apartment|apart;
…

EXAMPLE: fluent|adj|fluency|noun
RULE: ency$|noun|ent$|adj
EXCEPTION: emergency|emergent;
…

Facts Generation



SD-Fact Data

- Original SD generating rules in SD-Facts process:

No.	Rules to generate Raw SD-Pairs	Raw Retrieved	Valid Relevant	Invalid Irrelevant
1	$\|adj\|ness$|noun	2723	2723	0
2	ability$\|noun\|able$|adj	1278	1278	0
3	ization$\|noun\|ize$|verb	1215	1215	0
4	osis$\|noun\|otic$	366	366	0
5	le$\|adj\|ly$|adv	326	326	0
	…	…	…	…
71	ious$\|adj\|y$|noun	43	31	12
72	ant$\|adj\|ate$|verb	109	70	39
73	$\|noun\|ist$|noun	332	208	124
	…	…	…	…
93	ia$\|noun\|ian$|noun	136	1	135
94	a$\|noun\|an$|noun	273	1	272
95	gram$\|noun\|graphy$|noun	358	0	358
96	gram$\|noun\|graphic$|adj	228	0	228
97	$\|verb\|ably$|adv	57	0	57

SD-Rules Optimization

- Objective:
  To find an optimized set of SD-Rules to reach best performance (precision and recall)
  - to have high precision (95%)
  - to cover more derivations (recall) that are not in Lexicon

- Assumption:
  Use Lexicon as the testing corpus by assuming Lexicon is a representable subset of general English

Step 1 - Normalize

- Remove duplicates
  - Unify bi-directional SD-Rules (alphabetic order sorting)

- Remove overlap (child rules)
  - Example:

  magic|noun|E0038555|magical|adj|E0038557

  - $|noun|al$|adj|2013|ORG_RULE|PARENT
  - ic$|noun|ical$|adj|2013|ORG_RULE|CHILD

- Normalize 97 to 87 SD-Rules

Step 1 - Normalize

- Remove Child-Rules:

Parent-rules (9)	**Child-rules (10)**
$\|adj\|ity$|noun|2013|ORG_RULE|PARENT	ic$\|adj\|icity$|noun|2013|ORG_RULE|CHILD
$\|noun\|al$|adj|2013|ORG_RULE|PARENT	ic$\|noun\|ical$|adj|2013|ORG_RULE|CHILD
a$\|noun\|an$|adj|2013|ORG_RULE|PARENT	ia$\|noun\|ian$|adj|2013|ORG_RULE|CHILD
a$\|noun\|an$|noun|2013|ORG_RULE|PARENT	ia$\|noun\|ian$|noun|2013|ORG_RULE|CHILD
a$\|noun\|ar$|adj|2013|ORG_RULE|PARENT	ula$\|noun\|ular$|adj|2013|ORG_RULE|CHILD
ance$\|noun\|ant$|adj|2013|ORG_RULE|PARENT	iance$\|noun\|iant$|adj|2013|ORG_RULE|CHILD
ation$\|noun\|e$|verb|2013|ORG_RULE|PARENT	ization$\|noun\|ize$|verb|2013|ORG_RULE|CHILD
ency$\|noun\|ent$|adj|2013|ORG_RULE|PARENT	iency$\|noun\|ient$|adj|2013|ORG_RULE|CHILD
sis$\|noun\|tic$|adj|2013|ORG_RULE|PARENT	esis$\|noun\|etic$|adj|2013|ORG_RULE|CHILD
	osis$\|noun\|otic$|adj|2013|ORG_RULE|CHILD

Step 2 – Performance

- A good SD-Rule:
  has high precision and high frequency

- A good set of SD-Rules:
  includes better SD-Rules to reach better system performance for:
  - higher system precision (> 95%)
  - higher system recall

  - more SD-Rules (for better coverage)

Step 2 – System Performance

- Sort all SD-Rules by:
  - precision (= valid No. / raw No.)
  - raw No. (frequency).
  - alphabetic order of SD-Rules

- System performance:
  - System precision (cumulative):
    P = relevant, retrieved / retrieved
  - System recall:
    R = relevant, retrieved / relevant

  - More SD-Rules (for tie-breaker)

Step 3 – Optimization

- The optimal set has the best system performance

- Parent-Child SD-Rules
  Compare system performance of Parents (9) to Child SD-Rules (10)

- Add New SD-Rules:
  - from nomD
  - from original Facts
  - form suggestions

Step 3.1 – Optimization

- Evaluate Parent-Child-Grandchild Rules:
  - Only case 2 provides better results while replacing parent rule by child rule
  - Case 2.3 has the best results among case 2

ID	Parent-Rule	Candidate Child-Rules	Rule No.	Precision	Cutoff SD-Rule	Sys P	Sys R
0	Parent-rule only (Baseline)	No child-Rule	60	73.68%	a$\|noun\|iasis$|noun	95.76%	91.84%
1.1	$\|adj\|ity$|noun	c$\|adj\|city$|noun l$\|adj\|lity$|noun	61	73.68%	a$\|noun\|iasis$|noun	95.77%	90.81%
1.2	$\|adj\|ity$|noun	ic$\|adj\|icity$|noun l$\|adj\|lity$|noun	61	73.68%	a$\|noun\|iasis$|noun	95.77%	90.81%
2.1	$\|noun\|al$|adj	n$\|noun\|nal$|adj	64	62.65%	n$\|noun\|nal$|adj	95.10%	94.16%
2.2	$\|noun\|al$|adj	on$\|noun\|onal$|adj	64	62.65%	n$\|noun\|nal$|adj	95.17%	94.09%
2.3	$\|noun\|al$|adj	ion$\|noun\|ional$|adj	65	60.66%	ar$\|adj\|e$|noun	95.01%	94.30%
2.4	$\|noun\|al$|adj	tion$\|noun\|tional$|adj	65	60.66%	ar$\|adj\|e$|noun	95.04%	94.06%
3.1	a$\|noun\|an$|adj	No candidate child-rule found					
…	…	…	…	…	…	…	…
8.1	ency$\|noun\|ent$|adj	No candidate child-rule found					
9.1	sis$\|noun\|tic$|adj	esis$\|noun\|etic$|adj osis$\|noun\|otic$|adj ysis$\|noun\|ytic$|adj	62	73.68%	a$\|noun\|iasis$|noun	95.75%	91.59%

3.1 Optimization Example

- Sorted SD-Rules of case 2.3:

No.	Rule Precision	Raw Retrieved	Valid Relevant	Invalid Irrelevant	SD-Rule
1	100.00%	2723	2723	0	$\|adj\|ness$|noun
2	100.00%	1278	1278	0	ability$\|noun\|able$|adj
3	100.00%	326	326	0	le$\|adj\|ly$|adv
	…	…	…	…	…
64	62.65%	332	208	124	$\|noun\|ist$|noun
65	60.66%	183	111	72	ar$\|adj\|e$|noun
66	58.08%	582	338	244	al$\|adj\|e$|noun
	…	…	…	…	…
84	0.37%	273	1	272	a$\|noun\|an$|noun
85	0.00%	358	0	358	gram$\|noun\|graphy$|noun
86	0.00%	228	0	228	gram$\|noun\|graphic$|adj
87	0.00%	57	0	57	$\|verb\|ably$|adv

3.1 Optimization Example

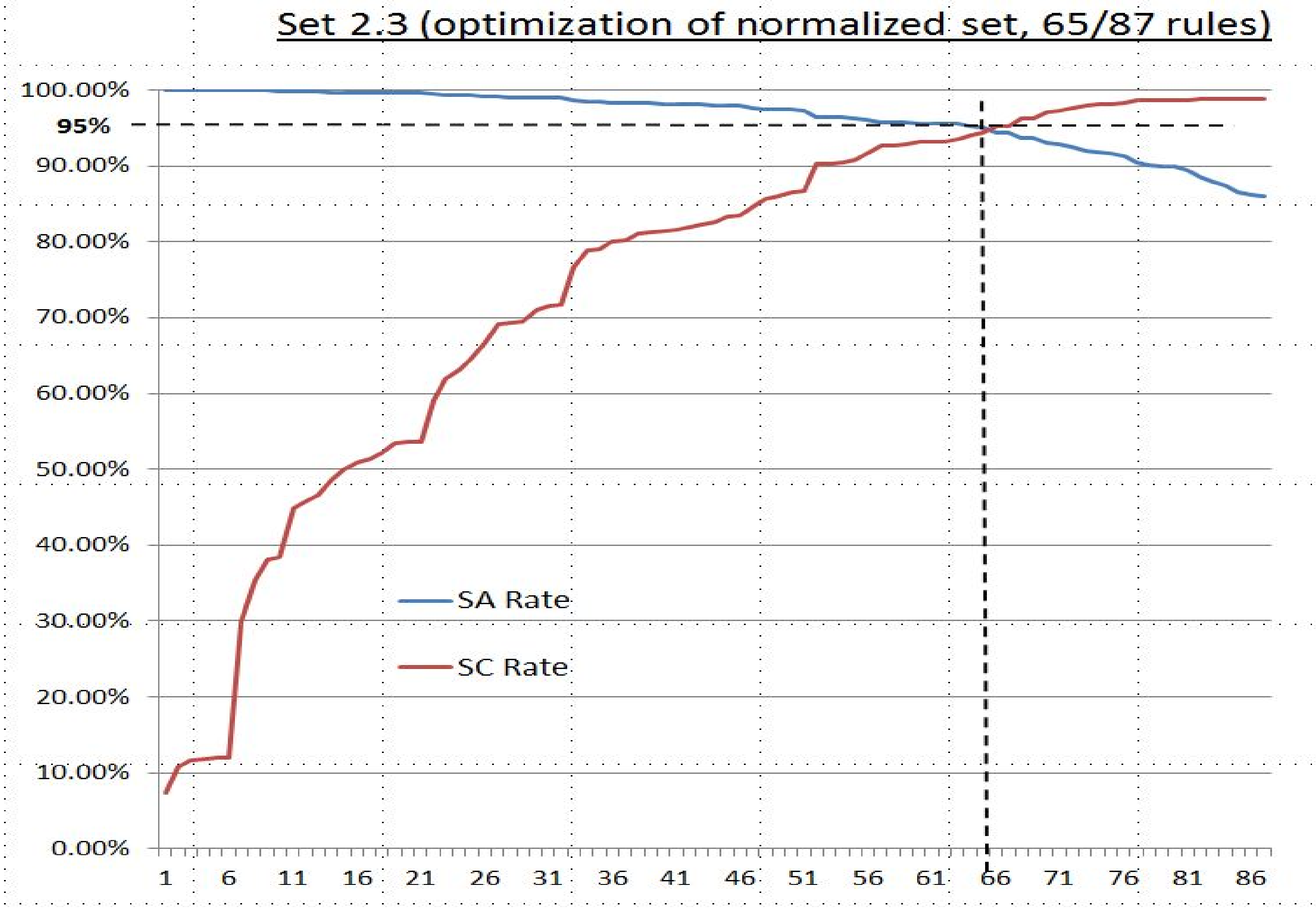
- Sorted SD-Rules of case 2.3:

No.	Rule Precision	Raw	Valid	InV.	SD-Rule	Accum Total	Accum Valid	System Precision	System Recall
1	100.00%	2723	2723	0	$\|adj\|ness$|noun	2723	2723	100.00%	7.33%
2	100.00%	1278	1278	0	ability$\|noun\|able$|adj	4001	4001	100.00%	10.77%
3	100.00%	326	326	0	le$\|adj\|ly$|adv	4327	4327	100.00%	11.65%
	…	…	…	…	…	…	…	…	…
64	62.65%	332	208	124	$\|noun\|ist$|noun	36673	34907	95.18%	94.00%
65	60.66%	183	111	72	ar$\|adj\|e$|noun	36858	35018	**95.01%**	94.30%
66	58.08%	582	338	244	al$\|adj\|e$|noun	37438	35356	94.44%	95.21%
	…	…	…	…	…	…	…	…	…
84	0.37%	273	1	272	a$\|noun\|an$|noun	42732	36673	87.33%	98.75%
85	0.00%	358	0	358	gram$\|noun\|graphy$|noun	43090	36673	86.60%	98.75%
86	0.00%	228	0	228	gram$\|noun\|graphic$|adj	43318	36673	86.13%	98.75%
87	0.00%	57	0	57	$\|verb\|ably$|adv	43375	36673	86.02%	98.75%

(37136)

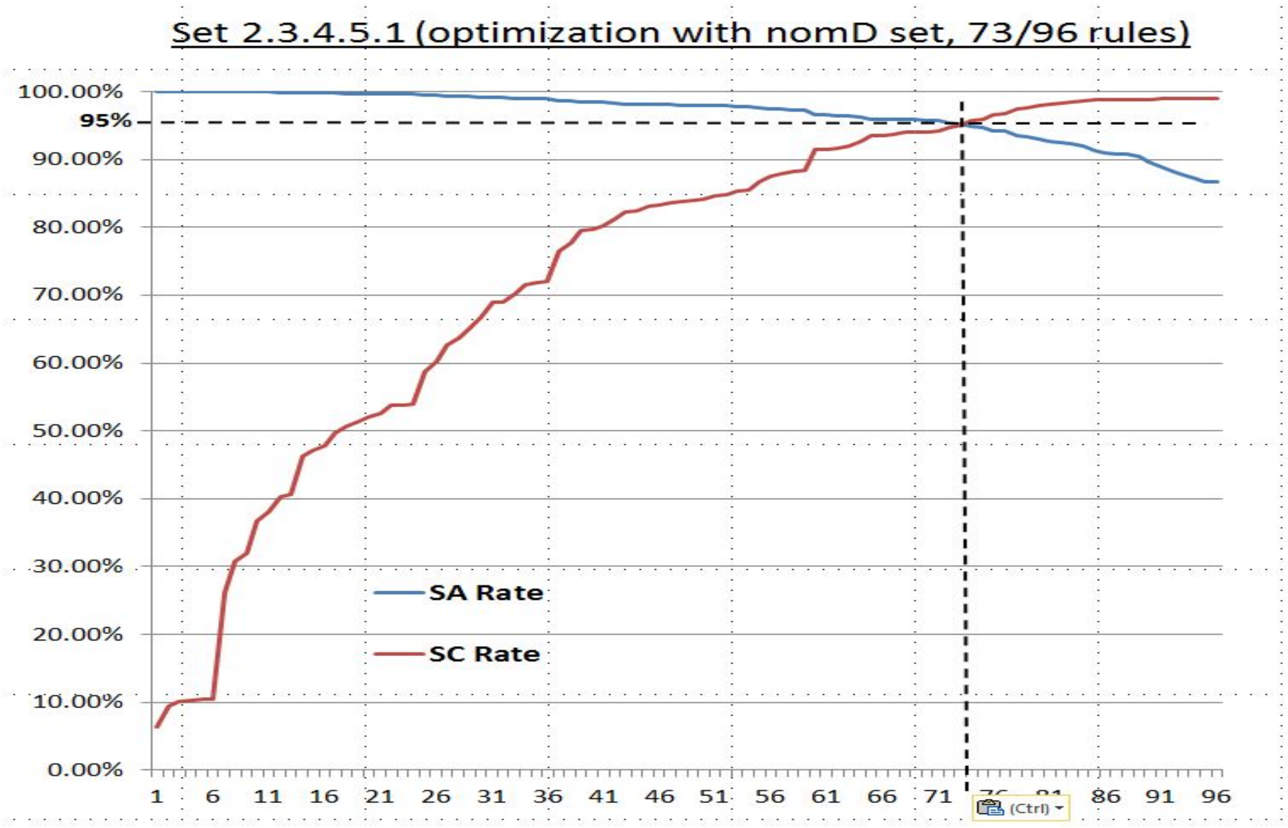
3.1 Optimization Results

- Best set includes 65 rules with S.P. of 95.01% and S.R. of 94.30%



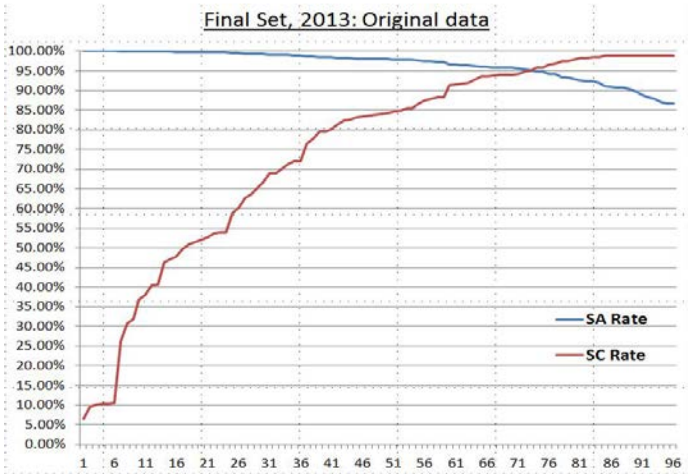
3.2 Enhancement – Add More Rules

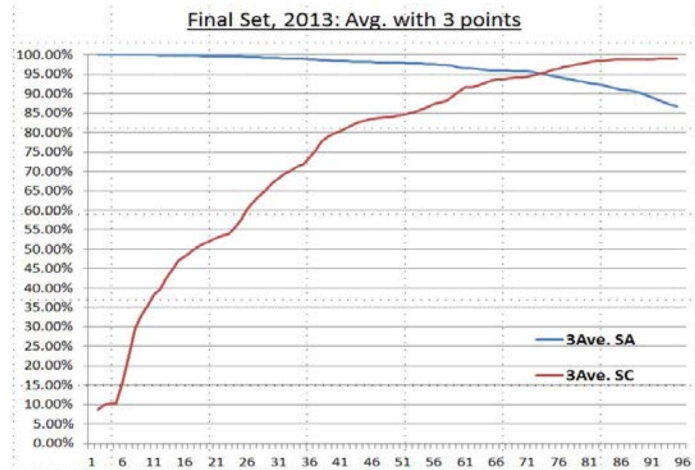
- Use same method to evaluate/add new SD-Rules
  - From nomD (4), 1 is the parent rule
  - From factD (5)
  - From others' suggestions (1)
- Final best set includes 73 rules with S.P. of 95.30% and S.R. of 95.01%

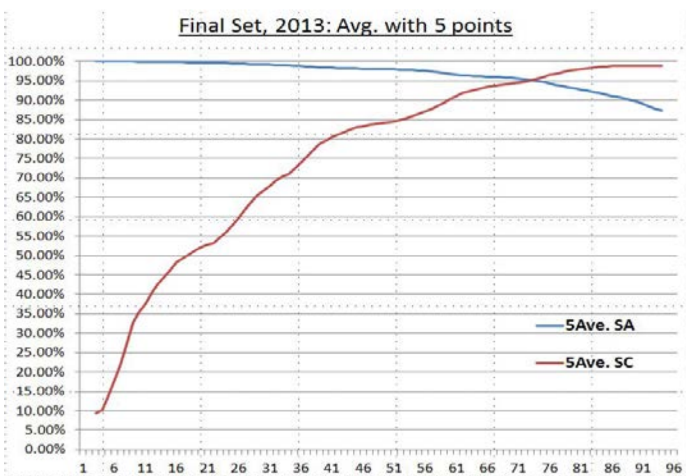


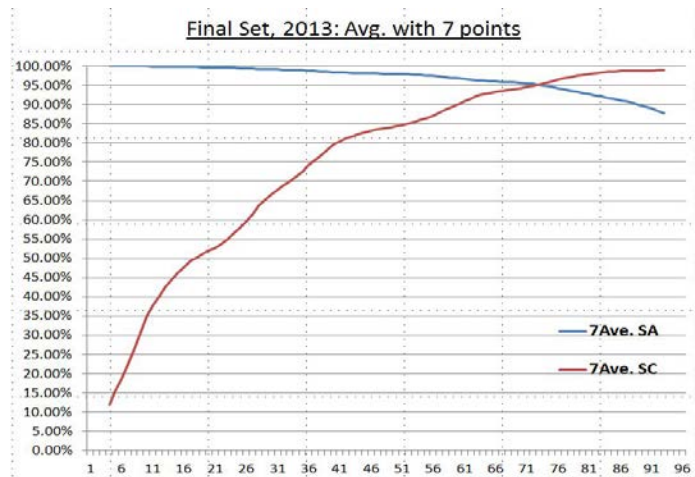
Result - Noise Reduction

- Smoothing algorithm – simple moving average of 3, 5, 7 window size
- The intersections are all around 95% for all cases
- Confirm our optimized goal of 95% S.A. is a good choice





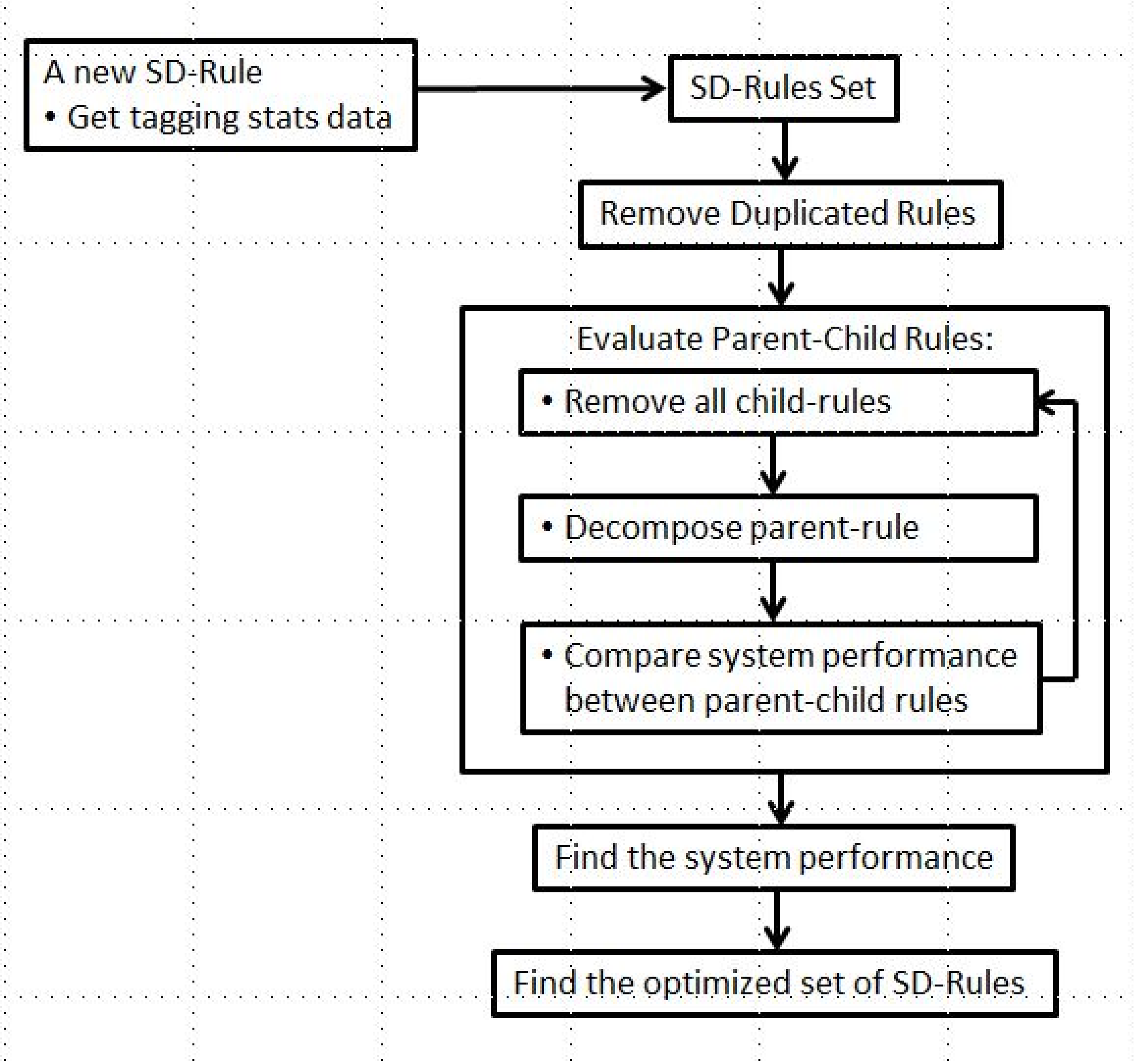




Optimization Summary

- Sort by:
  - precision (= valid No. /raw No.)
  - raw No. (frequency)
  - alphabetic order of SD-Rules (remove duplications)
- Find performance:
  - precision (cumulative): above 95%
  - recall: coverage
- Evaluate related Parent-Child Rules:
  - remove all child-rules
  - decompose parent-rules
  - evaluated performance (precision and recall)
- Get the best set of SD-Rules (with best performance - intersection of curves of precision and recall)

Process Summary



Results

- A comprehensive derivational features in Lexical Tools:
  - Type options: prefixD, suffixD, zeroD
  - Negation options
- A maintainable and scalable system for generating derivations with the Lexicon's annual release

- Better precision:
  - in Lexicon: virtual 100%
  - not in Lexicon (SD-Rules set): above 95.30%
- Better recall:
  - in Lexicon: 100% for the candidate SD-Rules
  - not in Lexicon (SD-Rules set): about 95.01%

Future Work

- Annual routine update with lexicon release

- Enhancement:
  - prefixD: work on more prefixes (2014)
  - suffixD: work on more candidate SD-Rules (2014)

- Analysis in prefixD, suffixD and zeroD

- Assumption (from Lexicon to English):
  - Is Lexicon a representable subset of general English?

Questions

- Lexical Systems Group: http://umlslex.nlm.nih.gov
- The SPECIALIST NLP Tools: http://specialist.nlm.nih.gov