Lexical Tools for UMLS Developers

November 4, 2001
Allen C. Browne, Guy Divita, Chris Lu
Lister Hill National Center for Biomedical Communications
National Library of Medicine
CONCEPT SEARCH

Term:
hand-foot-mouth diseases

Restrict to:
ALL SOURCES

Matching criteria:
Normalized string index
Normalized word index
Approximate matching
Word index
Left truncation
Right truncation
Lexical Tools for UMLS Developers

- The SPECIALIST lexicon – Browne
- The lexical tools – Divita/Lu
- Coffee Break – 10:00 - 10:30
- Lexical tools cont. – Divita/Lu
Lexical tools

SPECIALIST LEXICON

Text processing
The SPECIALIST Lexicon

• A syntactic lexicon
• Biomedical and general English
• Over 160,000 records
The SPECIALIST Lexicon

• General English:
• 10,000 most frequent words from the American Heritage word frequency list
• 2,000 words used by Longman’s Dictionary of Contemporary English
• Verbs and adjectives identified by heuristics
Lexicon Growth

Lexical items
Inflected forms

Counts: 0, 50000, 100000, 150000, 200000, 250000, 300000

George A. Miller

The Science of Words
1991
The SPECIALIST Lexicon

- Morphology
  - Inflection
  - Derivation
- Orthography
  - Spelling variants
- Syntax
  - Complementation for verbs, nouns, and adjectives
Morphology

• Inflectional
  – nucleus -- nuclei
  – cauterize, cauterizes, cauterized, cauterizing
  – red, redder reddest

• Derivational
  – laryngeal -- larynx
  – transport -- transportation
Orthography

Spelling Variation

- align -- aline
- Grave’s disease -- Graves’s disease -- Graves’ disease
- anesthetize -- anesthetise
- esophagus -- oesophagus
British and American Spelling

• Criticise -- criticize
• naturalise -- naturalize
• centre -- center
• foetus -- fetus
Try This Test

Which of the variant spellings below do you, accept as standard American English? After making your choices, consult the pages of this book to see what the dictionaries say. You are due for some astonishments.

Syntax -- Verb Complements

• **Intran**
  - I’ll treat.

• **tran=np**
  - He treated the patient.

• **ditran=np,pphr(with,np)**
  - She treated the patient with the drug.
Syntax -- Verb Complements

{base=treat
 entry=E0061964
   cat=verb
   variants=reg
   intran
   tran=np
   tran=pphr(with,np)
   tran=pphr(of,np)
   ditran=np,pphr(to,np)
   ditran=np,pphr(with,np)
   ditran=np,pphr(for,np)
   cplxtran=np,advbl
   nominalization=treatment|noun|E0061968
}


The 2002 SPECIALIST Lexicon

Number of lexical items

- noun
- adjective
- verb
- adverb
- preposition
- pronoun
- conjunction
- determiner
- modal
- auxilliary
- auxilliary
- compl
square square square square square square square square square
village fair and root
the circle
Lexicon Unit Records

{base=Kaposi's sarcoma
 spelling_variant=K aposi sarcoma
 entry=E0003576
 cat=noun
 variants=uncount
 variants=reg
 variants=glreg
}

{base=chronic
 entry=E0016869
 cat=adj
 variants=inv
 position=attrib(1)
 position=pred
 stative
}

{base=aspirate
 entry=E0010803
 cat=verb
 variants=reg
 tran=np
 nominalization=aspiration|noun|E0010804
}

{base=in
 entry=E0033870
 cat=prep
}
Noun Variants

{base=K aposi's sarcoma
 spelling_variant=K aposi sarcoma
 entry=E0003576
   cat=noun
   variants=uncount
   variants=reg
   variants=glreg
}

• K aposi’s sarcoma
• K aposi’s sarcomas
• K aposi’s sarcomata
• K aposi sarcoma
• K aposi sarcomas
• K aposi sarcomata
Regular Nouns

The plural suffix is s.
y becomes ie following a consonant before s.
e is inserted before s if the base ends in s, z, x, ch, or s
# Regular Nouns

<table>
<thead>
<tr>
<th>Base ends with</th>
<th>Plural ends with</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cy</td>
<td>Cies</td>
<td>fly: flies</td>
</tr>
<tr>
<td>-s</td>
<td>-ses</td>
<td>illness: illnesses</td>
</tr>
<tr>
<td>-z</td>
<td>-zes</td>
<td>waltz: waltzes</td>
</tr>
<tr>
<td>-x</td>
<td>-xes</td>
<td>box: boxes</td>
</tr>
<tr>
<td>-ch</td>
<td>-ches</td>
<td>match: matches</td>
</tr>
<tr>
<td>-sh</td>
<td>-shes</td>
<td>splash: splashes</td>
</tr>
<tr>
<td>X</td>
<td>Xs</td>
<td>book: books</td>
</tr>
<tr>
<td>singular ends with:</td>
<td>plural ends with:</td>
<td>Examples</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>-us</td>
<td>-i</td>
<td>focus/foci</td>
</tr>
<tr>
<td>-ma</td>
<td>-mata</td>
<td>trauma/traumata</td>
</tr>
<tr>
<td>-a</td>
<td>-ae</td>
<td>larva/larvae</td>
</tr>
<tr>
<td>-um</td>
<td>-a</td>
<td>ilium/ilia</td>
</tr>
<tr>
<td>-on</td>
<td>-a</td>
<td>taxon/taxa</td>
</tr>
<tr>
<td>-sis</td>
<td>-ses</td>
<td>analysis/analyses</td>
</tr>
<tr>
<td>-is</td>
<td>-ides</td>
<td>cystis/cystides</td>
</tr>
<tr>
<td>-men</td>
<td>-mina</td>
<td>foramen/foramina</td>
</tr>
<tr>
<td>-ex</td>
<td>-ices</td>
<td>index/indices</td>
</tr>
<tr>
<td>-x</td>
<td>-ces</td>
<td>matrix/matrices</td>
</tr>
</tbody>
</table>
Uncount Nouns
(abstract or mass)

{base=smallpox
entry=E0056359
  cat=noun
  variants=uncount
}
{base=potassium
entry=E0049387
  cat=noun
  variants=uncount
}

• * a smallpox
• * two smallpoxes
• much smallpox
• * a potassium
• * two potassiums
• much potassium
Fixed Plural Nouns

{base=police
tenry=E0048616
cat=noun
variants=plur
}

{base=scissors
tenry=E0054633
cat=noun
variants=plur
}
Irregular Nouns

{base=corpus
  entry=E0019113
  cat=noun
  variants=irreg|corpora|
  variants=reg
}

{base=larynx
  entry=E0036919
  cat=noun
  variants=irreg|larynges|
  variants=reg
}

Regular Verbs

- The third person present tense suffix is s.
  - y becomes ie following a consonant before s.
  - e is inserted between z, x, ch, or sh and s.
- The past tense suffix is ed.
  - y becomes ie following a consonant before ed.
  - Final e is deleted before ed.
Regular Verbs

- dismiss: dismisses, dismissed, dismissing
- agree: agrees; agreed; agreeing
- dry: dries, dried, drying
Regular Doubling Verbs

- End in a CVC pattern
- Double the final consonant before ed and ing.
- Are otherwise regular
- variants=regd
- e.g. control: controls, controlled, controlling
Irregular Verbs

{base=dive
cat=verb
    variants=reg
    variants=irreg|dives|dove|dove|diving|
intran
intran;part(in)
...
}
"There's the place we sightsaw yesterday."
Regular Adjectives and Adverbs

- The comparative suffix is *er*.
- The superlative suffix is *est*.
  - *y* become *ie* after a consonant before *er* or *est*.
  - Final *e* is deleted before *er* or *est*.
- e.g. green: greener, greenest
Regular Doubling Adjectives and Adverbs

- CVC final pattern
- Final consonant is doubled before ed or est.
- Otherwise regular
- e.g. red: redder, reddest
Ancillary Data Bases

- Synonymy
  - sm.db
- Derivation
  - dm.db, dm.rules
- Inflection
  - im.rules
- Neoclassical compounds
  - nc.db
Derivational Facts and Rules

dm.facts

treatment|noun|treat|verb
prohibition|noun|prohibitive|adj
cell lineage|noun|cell line|noun
photochemotherapeutic|adj|photochemotherapy|noun
pharmacotherapeutic|adj|pharmacotherapy|noun
Derivational Facts and Rules

dm.rules

# e.g. alienation|alienate
ation$|noun|ate|verb
 ration|rate; station|state;
Inflectional Facts and Rules

im.rules

# Noun rules (glreg)
us$|noun|singular|i$|noun|plural
antus|anti;
ma$|noun|singular|mata$|noun|plural
a$|noun|singular|ae$|noun|plural
um$|noun|singular|a$|noun|plural
on$|noun|singular|a$|noun|plural
sis$|noun|singular|ses$|noun|plural
is$|noun|singular|ides$|noun|plural
men$|noun|singular|mina$|noun|plural
ex$|noun|singular|ices$|noun|plural
x$|noun|singular|ces$|noun|plural
Neoclassical compounds

nc.db

abdomin(o)|abdomen|root
ab|away from|prefix
acanth(o)|prickle|root
acar(o)|mite|root
acetabul(o)|acetabulum|root
ad|towards|prefix
agogue|inducing|terminal
albumin(o)|albumin|root
sis|condition|terminal
stomy|surgical opening|terminal
Synonyms

sm.db

alar|adj|wing|noun
amygdaline|adj|tonsil|noun
articuläre|adj|joint|noun
bulbar|adj|medulla oblongata|noun
fununcular|adj|boil|noun
genicular|adj|knee|noun
hepatocellular|adj|liver cells|noun
lazar|adj|leprosy|noun
lenticular|adj|crystalline lens|noun
ypsiliform|adj|upsiloid|adj
wolfram|noun|tungsten|noun
double vision|noun|diplopia|noun
Relational Tables

- One line records
- Pipe separated Fields -- "|"
- Keyed to EUI
- LRAGR matches forms to EUIs
- Word index: L R W D
Relational Tables

- LRA GR - Agreement
- LR CMP - Complements
- LR FIL - Files
- LR FLD - Fields
- LR MOD - Modification
- LR NOM - Nominalization
- LRP RN - Pronouns
- LRP RP - Properties
- LR SPL - Spelling
- LR TRM - Trademarks
- LR W D - Word index
LRAGR

Agreement and Inflection

- EUI - Entry ID
- STR - Inflected form
- SCA - Syntactic category
- AGR - agreement information
- BAS - Base form (morphological)
- CIT - Citation form (base=)
Kaposi sarcoma
Kaposi sarcomas
Kaposi sarcomata
Kaposi's sarcoma
Kaposi's sarcomas
Kaposi's sarcomata
Lexical tools

SPECIALIST LEXICON

Text processing
Lexical Tools

- **Wordind** -- breaks strings into words
  - Produces the Metathesaurus word indexes (M RX W)
- **LVG** -- performs various lexical transformations
- **NORM** -- a selection of LVG transformations,
  - Used for Metathesaurus indexing
  - Produces the Metathesaurus Normalized word and string indexes (M RX NW & M RX NS)
  - Used to access those indexes
Normalization

- 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
- Hodgkin's disease, NOS
- Disease hodgkin
- Disease, Hodgkins
- Diseases, Hodgkins
- Hodgkins Diseases
- Hodgkins disease
- hodgkin's disease
- Disease; Hodgkins
- Disease, Hodgkin
Lexical Tools
The Lexical Tools

- Introduction
- Norm/WordInd/Lvg
- Details, Details and More Details ...
- Installation
- Embedding This into Your Application
- Using The Lexical Tools with The Metathesaurus
- Building an Index Using The Lexical Tools
- Plans for 2003 and Beyond
Lexical Tools: Introduction

• These tools:
  – Includes Norm, WordInd, and Lvg
  – Pure Java based
  – Command line tools
  – Java APIs
Lexical Tools: Introduction

• These tools are good for
  - aggressive text pattern matching
  - making word, term, phrase indexes
  - matching queries with indexed entries
  - increasing recall and/or precision
Lexical Tools: Introduction

• Characteristics of all the command line tools
  - take input from the screen or a file
  - put their results to the screen or a file
  - Interpret fielded text
    • Can be told which fields contain what type of information
Lexical Tools: Norm

Lexical Tools Documentation
- Release Notes
- Installation
- Repository List
- MySql DB
- Configuration Setup

Norm (Java)

Installation Prog
- Norm
- LuiNorm
- WordInd
- Lvg
Lexical Tools: Norm

- Norm abstracts away from:
  - case
  - Punctuation
  - word order
  - possessive forms
  - inflectional variation
Lexical Tools: Norm

- remove genitives
- replace punctuation with spaces
- remove stop words
- lowercase
- uninflect each word
- word order sort

Hodgkin's Diseases, NOS
Lexical Tools: Norm

- remove genitives
- replace punctuation with spaces
- remove stop words
- lowercase
- uninflect each word
- word order sort

Hodgkin's Diseases, NOS
Lexical Tools: NORM

- remove genitives
- replace punctuation with spaces
- remove stop words
- lowercase
- uninflect each word
- word order sort

Hodgkin's Diseases, NOS

Hodgkin's Diseases, NOS

Hodgkin Diseases, NOS

Hodgkin Diseases

hodgkin diseases
Lexical Tools: Norm

- remove genitives
- replace punctuation with spaces
- remove stop words
- lowercase
- uninflect each word
- word order sort
<table>
<thead>
<tr>
<th>Down’s Syndrome</th>
<th>down syndrome</th>
</tr>
</thead>
<tbody>
<tr>
<td>A cetolyses</td>
<td>acetolysis</td>
</tr>
<tr>
<td>Lung cancer</td>
<td>cancer lung</td>
</tr>
</tbody>
</table>
Lexical Tools: Norm

> norm
Paget’s disease-scapula
Paget's disease-scapula | disease paget scapula
Scapula, Paget Disease
Scapula, Paget Disease | disease paget scapula
Lexical Tools: WordInd

Lexical Tools Documentation
- Release Notes
- Installation
- Repository List
- MySql DB
- Configuration Setup

Installation Program
- Norm
- LuiNorm
- WordInd
- Lvg

WordInd Java
Lexical Tools: WordInd

• Breaks words into tokens
• Passes other fields to output, untouched
• Lowercases
• Removes white space and punctuation
Lexical Tools: wordInd

Useful command line options for wordInd

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-t[:Num]</td>
<td>Defines what field to tokenize</td>
</tr>
<tr>
<td>-f[:Num[:Num]]</td>
<td>Defines what fields get passed through</td>
</tr>
</tbody>
</table>
Lexical Tools: WordInd

> wordInd -t:7 -F:1:6

C0185495|ENG|P|L0223844|PF|S0298948|Denis-Browne splint strapping|3|

C0185495|S0298948|denis
C0185495|S0298948|browne
C0185495|S0298948|splint
C0185495|S0298948|strapping
Lexical Tools: Lvg

Lexical Tools Documentation
- Release Notes
- Installation
- Repository List
- MySql DB
- Configuration Setup

Lvg

Java

Installation

Prog
## Lexical Tools: Flow Components

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Return known acronyms</td>
</tr>
<tr>
<td>a</td>
<td>Return known acronym expansions</td>
</tr>
<tr>
<td>B</td>
<td>Uninflect words in a term</td>
</tr>
<tr>
<td>b</td>
<td>Uninflect a term</td>
</tr>
<tr>
<td>C</td>
<td>Canonicalize</td>
</tr>
<tr>
<td>Ct</td>
<td>Retrieve the citation term</td>
</tr>
</tbody>
</table>
## Lexical Tools: Flow Components

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td><strong>Tokenize a term into &quot;words&quot;</strong></td>
</tr>
<tr>
<td>ca</td>
<td><strong>Tokenize, keep everything</strong></td>
</tr>
<tr>
<td>ch</td>
<td><strong>Tokenize without breaking hyphens</strong></td>
</tr>
<tr>
<td>d</td>
<td><strong>Generate derivational variants</strong></td>
</tr>
<tr>
<td>dc~N</td>
<td><strong>Generate derivational variants with specifying output categories</strong></td>
</tr>
</tbody>
</table>
# Lexical Tools: Flow Components

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Tool</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td><strong>Retrieve the unique E UI for a term</strong></td>
<td></td>
</tr>
<tr>
<td>f</td>
<td><strong>Filter output to contain only forms from lexicon</strong></td>
<td></td>
</tr>
<tr>
<td>Gn</td>
<td><strong>Generate known fruitful variants</strong></td>
<td></td>
</tr>
<tr>
<td>g</td>
<td><strong>Remove genitive</strong></td>
<td></td>
</tr>
<tr>
<td>i</td>
<td><strong>Generate inflectional variants</strong></td>
<td></td>
</tr>
<tr>
<td>ici~Cats+Infls</td>
<td><strong>Generate inflectional variants, by Categories and inflections</strong></td>
<td></td>
</tr>
</tbody>
</table>
## Lexical Tools: Flow Components

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td><strong>Retrieve category and inflection for a term</strong></td>
</tr>
<tr>
<td>l</td>
<td><strong>Lowercase</strong></td>
</tr>
<tr>
<td>N</td>
<td><strong>Normalize the input text in a non-canonical way (Norm)</strong></td>
</tr>
<tr>
<td>o</td>
<td><strong>Replace punctuations with spaces</strong></td>
</tr>
<tr>
<td>p</td>
<td><strong>Strip Punctuation</strong></td>
</tr>
<tr>
<td>q</td>
<td><strong>Strip diacritics</strong></td>
</tr>
<tr>
<td>R</td>
<td><strong>Generate derivational variants, recursively</strong></td>
</tr>
<tr>
<td>r</td>
<td><strong>Generate synonyms, recursively</strong></td>
</tr>
</tbody>
</table>
## Lexical Tools: Flow Components

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>s</td>
<td>Generate known spelling variants</td>
</tr>
<tr>
<td>t</td>
<td>Strip stop words</td>
</tr>
<tr>
<td>u</td>
<td>Uninvert the input phrase around commas</td>
</tr>
<tr>
<td>w</td>
<td>Sort words by order</td>
</tr>
<tr>
<td>y</td>
<td>Generate synonyms</td>
</tr>
</tbody>
</table>
Lexical Tools: Flows

leave → inflect → leaves, leaving, left
Lexical Tools: Flows

> lvg -f:i
leave
leave|leave|128|1|i|1|
leave|leave|128|512|i|1|
leave|leaves|128|8|i|1|
leave|left|1024|64|i|1|
leave|left|1024|32|i|1|
leave|leave|1024|1|i|1|
leave|leave|1024|262144|i|1|
leave|leave|1024|1024|i|1|
leave|leaves|1024|128|i|1|
leave|leaving|1024|16|i|1|
Flow components can be arranged so that the output of one is the input to another.
Lexical Tools: A Serial Flow

> lvg -f:l:q:g:t:p:w
The Gougerot-Sjögren's Syndrome
The Gougerot-Sjögren's Syndrome

> gougertosjogren syndrome|2047|16777215|

> l+q+g+t+p+w|1|
Lexical Tools: Parallel Flows

Multiple flows can be defined

Input term → noOperation → Output term
Input term → Uninflect → synonyms → Output terms
Lexical Tools: Parallel Flows

First Flow

Second Flow

> lvg -f:n -f:B:y

ear

ear|**ear**|2047|1048575|n|1|

ear|**aural**|1|1|B +y|2|

ear|**auricularis**|1|1|B +y|2|

ear|**otic**|1|1|B +y|2|

ear|**otor**|1|1|B +y|2|
Lexical Tools: Fielded Output

> lvg -f:L leaves

leaves leaves 1152 136 L 1

Input Term Output Term Categories Inflections

Flow history Flow Number
Lexical Tools: Fielded Output

Output term
leaves

Categories
noun
verb

Inflections
plural
pres3ps
Lexical Tools: Categories

Category bit vector

Categories

1152

Verb  Pron  Prep  Noun  Modal  Det  Conj  Compl  Aux  Adv  Adj

1  0  0  1  0  0  0  0  0  0  0
# Lexical Tools: Categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Bit Vector Positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adjective</td>
<td>1</td>
</tr>
<tr>
<td>Adverb</td>
<td>2</td>
</tr>
<tr>
<td>Auxiliary</td>
<td>4</td>
</tr>
<tr>
<td>Complement</td>
<td>8</td>
</tr>
<tr>
<td>Conjunction</td>
<td>16</td>
</tr>
<tr>
<td>Determiner</td>
<td>32</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
<th>Bit Vector Positions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modal</td>
<td>64</td>
</tr>
<tr>
<td>Noun</td>
<td>128</td>
</tr>
<tr>
<td>Preposition</td>
<td>256</td>
</tr>
<tr>
<td>Pronoun</td>
<td>512</td>
</tr>
<tr>
<td>Verb</td>
<td>1024</td>
</tr>
</tbody>
</table>
Lexical Tools: Inflections

Inflections
136
# Lexical Tools: Inflections

<table>
<thead>
<tr>
<th>Grammatical Form</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>1</td>
</tr>
<tr>
<td>Comparative</td>
<td>2</td>
</tr>
<tr>
<td>Superlative</td>
<td>4</td>
</tr>
<tr>
<td>Plural</td>
<td>8</td>
</tr>
<tr>
<td>Present Participle</td>
<td>16</td>
</tr>
<tr>
<td>Past</td>
<td>32</td>
</tr>
<tr>
<td>Past Participle</td>
<td>64</td>
</tr>
<tr>
<td>Present 3rd Person Singular</td>
<td>128</td>
</tr>
<tr>
<td>Positive</td>
<td>256</td>
</tr>
<tr>
<td>Singular</td>
<td>512</td>
</tr>
<tr>
<td>Infinitive</td>
<td>1024</td>
</tr>
<tr>
<td>Pres 123p</td>
<td>2048</td>
</tr>
<tr>
<td>Past Neg</td>
<td>4096</td>
</tr>
<tr>
<td>pres123pN eg</td>
<td>8192</td>
</tr>
<tr>
<td>Pres1s</td>
<td>16384</td>
</tr>
<tr>
<td>past1p23pN eg</td>
<td>32768</td>
</tr>
<tr>
<td>past1s3sN eg</td>
<td>65536</td>
</tr>
<tr>
<td>pres1p23p</td>
<td>131072</td>
</tr>
<tr>
<td>pres1p23pN eg</td>
<td>524288</td>
</tr>
<tr>
<td>past1s3s</td>
<td>1048576</td>
</tr>
<tr>
<td>pres</td>
<td>2097152</td>
</tr>
<tr>
<td>pres3sN eg</td>
<td>4194304</td>
</tr>
<tr>
<td>presN eg</td>
<td>8388608</td>
</tr>
<tr>
<td>all</td>
<td>16777215</td>
</tr>
<tr>
<td>Input term</td>
<td>Output term</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>leaves</td>
<td>leaves</td>
</tr>
</tbody>
</table>
Lexical Tools: Fielded Input

Input term
leaves

Categories
noun

Inflections
Lexical Tools: Fielded Input

> lvg -f:L -t:1 -cf:2

leaves | 128
leaves | leaves | 128 | 8 | L | 1
Lexical Tools: Command Line Syntax

- Hierarchical structure

- Option
  - Option Attribute
    - 1st Attribute Parameter
    - 2nd Attribute Parameter
    - Nth Attribute Parameter
Lexical Tools: Command Line Syntax

-\texttt{f:L :ici~128+ALL L :w}
**Lexical Tools: Post Flow Options**

<table>
<thead>
<tr>
<th>SC</th>
<th>Show category names</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI</td>
<td>Show inflection names</td>
</tr>
<tr>
<td>ccgi</td>
<td>Mark the end of the set of variants returned</td>
</tr>
<tr>
<td>F:Int[Int]</td>
<td>Specify fields for outputs</td>
</tr>
<tr>
<td>ti</td>
<td>Display the only input term in the output when using fielded input</td>
</tr>
<tr>
<td>R:Int</td>
<td>Restrict the number of variants returned</td>
</tr>
</tbody>
</table>
Lexical Tools: Post Flow Options

Show category names
Show inflection names

> lvg -f:L -SC -SI

phosphoprotein
phosphoprotein | phosphoprotein | <noun> | <base+singular> | L | 1 |

sclerosing
sclerosing | sclerosing | <adj+verb> | <base+presPart+positive> | L | 1 |
Lexical Tools: Post Flow Options

Mark the end of the set of variants returned

> lvg -f:L -ccgi
behavior
behavior|behavior|128|513|L|1|
__THE_END__

Mark the end of processing
Lexical Tools: Post Flow Options

Specify fields for outputs

Display only the 8th and 6th field from the output

> lvg -f:u -t:7 -F:8:6

C0035440|ENG|S|L0035434|VW|S0003894|Rheumatic carditis, acute

acute Rheumatic carditis| S0003894
Lexical Tools: Post Flow Options

Display only the input term field when using fielded input

> lvg -f:u -t:7 -ti

C0035440 | S0003894 | **Rheumatic carditis, acute**

**Rheumatic carditis, acute** | acute Rheumatic carditis | 2047 | 16777215 | u | 1 |
Lexical Tools: Post Flow Options

Restrict the number of variants returned

> lvg -f:i -R:2
foo
foo | foo | 128 | 1 | i | 1 |
foo | foos | 128 | 8 | i | 1 |

Note: Dangerous! Do not try this at home!

Note: The unrestricted output would have produced 12 rows otherwise

Limit the number of output terms to 2
Lexical Tools: Post Flow Options

<table>
<thead>
<tr>
<th>EC:Long</th>
<th>Display variants: exclude categories specified</th>
</tr>
</thead>
<tbody>
<tr>
<td>EI:Long</td>
<td>Display variants: exclude inflections specified</td>
</tr>
<tr>
<td>DC:Long</td>
<td>Display variants that only contain the categories specified</td>
</tr>
<tr>
<td>DI:Long</td>
<td>Display variants that only contain the inflections specified</td>
</tr>
</tbody>
</table>
Lexical Tools: Post Flow Options

Display variants: exclude categories specified

> lvg -f:i -EC:1919
sleep
sleep | sleep | 128 | 1 | i | 1
sleep | sleep | 128 | 512 | i | 1

Display variants, but exclude all terms other than nouns
Lexical Tools: Post Flow Options

Display variants exclude inflection specified

Display variants, but exclude base forms

> lvg -f:i -EC:1919 -EI:1
sleep
sleep | sleep | 128 | 512 | i | 1
Lexical Tools: Post Flow Options

Display variants contain category specified

```
> lvg -f:i -DC:128
sleep
sleep | sleep | 128 | 1 | i | 1
sleep | sleep | 128 | 512 | i | 1
```

Display variants, but only include nouns in the output
Display variants contain inflection specified

> lvg -f:i -DI:255

sleep
sleep|sleep|128|1|i|1|
sleep|sleep|1024|1|i|1|
sleep|slept|1024|64|i|1|
sleep|slept|1024|32|i|1|
sleep|sleeps|1024|128|i|1|
sleep|sleeping|1024|16|i|1|

Display variants, but only include “simplified” inflections
Lexical Tools: Post Flow Options

**CR:o** Combine record by output term

**CR:oc** Combine record by output term and category

**CR:oi** Combine record by output term and inflection

**St:o** Sort outputs by terms in an alphabetical order

**St:oc** Sort outputs by term and category

**St:oci** Sort outputs by term, category, and inflection
Lexical Tools: Post Flow Options

Combine record by output term

> lvg -f:i -CR:o

sleep

sleep | sleep | 1152 | 263681 | i | 1 |
sleep | slept | 1024 | 96 | i | 1 |
sleep | sleeps | 1024 | 128 | i | 1 |
sleep | sleeping | 1024 | 16 | i | 1 |

Note: this is a noun+verb
Lexical Tools: Post Flow Options

**Combine record by output term and category**

```plaintext
> lvg -f:i  -CR:oc
sleep
sleep  sleep  128  513  i  1 |
sleep  sleep  1024  263169  i  1 |
sleep  slept  1024  96  i  1 |
sleep  sleeps  1024  128  i  1 |
sleep  sleeping  1024  16  i  1 |
```

Combine records by term and category.

Note: this is both the base+singular inflections combined.
Lexical Tools: Post Flow Options

Combine record by output term and inflection

Note: This is an adj+adv+noun

lvg -f:b -CR:oi
left
left | left | 131 | 1 | b | 1 |
left | leave | 1024 | 1 | b | 1 |
Lexical Tools: Post Flow Options

Sort outputs by terms in an alphabetical order

>lvg -f:i -St:o

<table>
<thead>
<tr>
<th>Term</th>
<th>Frequency</th>
<th>Offset</th>
<th>Length</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>see</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>32</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>1</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>262144</td>
<td>1</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>1024</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>1024</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>16</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>64</td>
<td>1</td>
<td>i</td>
</tr>
<tr>
<td>see</td>
<td>1024</td>
<td>128</td>
<td>1</td>
<td>i</td>
</tr>
</tbody>
</table>
Lexical Tools: Post Flow Options

Sort outputs by term and category

```bash
> lvg -f:i -St:oc
left
...
left | left | 1 | 1 | i | 1 |
left | left | 1 | 256 | i | 1 |
left | left | 2 | 1 | i | 1 |
left | left | 2 | 256 | i | 1 |
left | left | 128 | 1 | i | 1 |
left | left | 128 | 512 | i | 1 |
left | left | 128 | 8 | i | 1 |
left | left | 1024 | 64 | i | 1 |
left | left | 1024 | 32 | i | 1 |
left | lefts | 128 | 8 | i | 1 |
```

Sort by the output term and category
Lexical Tools: Post Flow Options

Sort outputs by term, category, and inflection

> lvg -f:i -St:oci

<table>
<thead>
<tr>
<th>term</th>
<th>category</th>
<th>inflection</th>
<th>frequency</th>
<th>score</th>
<th>rank</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>see</td>
<td></td>
<td></td>
<td>1024</td>
<td>32</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td></td>
<td></td>
<td>1024</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td></td>
<td></td>
<td>1024</td>
<td>1024</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td></td>
<td></td>
<td>1024</td>
<td>262144</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td>seeing</td>
<td></td>
<td>1024</td>
<td>16</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td>seen</td>
<td></td>
<td>1024</td>
<td>64</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>see</td>
<td>sees</td>
<td></td>
<td>1024</td>
<td>128</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Sort by the output term, category and inflection
Lexical Tools: Why Base+Singular

• Base form for Nouns:
  - Singular form
    • There are exceptions:
      - Police
        !vg -f:i -SC -SI -DC:128
        police
        police|police|<noun>|<base>|i|1|
        police|police|<noun>|<plural>|i|1|
**Lexical Tools: Global Behaviors**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>i:filename</code></td>
<td>Define input file name</td>
</tr>
<tr>
<td><code>o:filename</code></td>
<td>Define output file name</td>
</tr>
<tr>
<td><code>x:filename</code></td>
<td>Loading an alternative configuration file</td>
</tr>
<tr>
<td><code>p</code></td>
<td>Interactive prompt</td>
</tr>
<tr>
<td><code>m</code></td>
<td>Print extra information of flow mutations</td>
</tr>
<tr>
<td><code>s:Str</code></td>
<td>Defines a field separator.</td>
</tr>
</tbody>
</table>
Lexical Tools: Global Behaviors

**Interactive prompt**

```
> lvg -f:s -CR:oc -p

- Please input a term (type "Ctl-d" to Quit) >

  color

  color|color|128|513|s|1|

  color|color|1024|263169|s|1|

  color|colour|128|513|s|1|

  color|colour|1024|263169|s|1|

- Please input a term (type "Ctl-d" to Quit) >
```
Lexical Tools: Global Behaviors

| K : dlnt | Configure the derivation morphology behavior |
| K : ilnt | Configure the inflection morphology behavior |
### Lexical Tools: Global Behaviors

**Configure the derivation morphology behavior**

<table>
<thead>
<tr>
<th></th>
<th>Restrict the output to those variants which are known to the lexicon (default).</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Restrict the output to those variants which are known to the lexicon, unless none of the variants are found in the lexicon, in which case the entire (rule-generated) list is returned.</td>
</tr>
<tr>
<td>3</td>
<td>No restriction on the output of the morphology. Both facts and rules generated variants are displayed.</td>
</tr>
</tbody>
</table>
Lexical Tools: Global Behaviors

Configure the derivation morphology behavior

 restrict the derivations to terms that are known to the Lexicon

> lvg -f:d -kd:1 -m

breath

breath | breather | 128 | 1 | d | 1 | RULE | $ | verb | base | er$ | noun | base |
breath | breathy | 1 | 1 | d | 1 | RULE | $ | noun | base | y$ | adj | base |
breath | breathless | 1 | 1 | d | 1 | FACT | breath | 128 | breathless | 1 |
## Lexical Tools: Global Behaviors

**Configure the inflection morphology behavior**

<table>
<thead>
<tr>
<th></th>
<th>Restrict the output to those variants which are known to the lexicon.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Restrict the output to those variants which are known to the lexicon, unless none of the variants are found in the lexicon, in which case the entire (rule-generated) list is returned (Default).</td>
</tr>
<tr>
<td>3</td>
<td>No restriction on the output of the morphology. Both facts and rules generated variants are displayed</td>
</tr>
</tbody>
</table>
Lexical Tools: Global Behaviors

Configure the inflection morphology behavior

> lvg -f:i -m -ki:3 -R:5 -F:1:2:7:3:4 -SC -SI

Amish

Amish | Amish | FACT | <noun> | <base>

Amish | Amish | FACT | <noun> | <plural>

Amish | Amishs | RULE | <noun> | <plural>

Amish | Amishes | RULE | <noun> | <plural>

Amish | Amishs | RULE | <verb> | <pres>

Return all inflected variants, both fact and rule generated
Lexical Tools: **No Operation**

- **f:n**  
  Copies the input term to the output with no transformation

```plaintext
> lvg -f:n -f:d -f:y -SC -SI
force
force | **force** | <all> | <all> | n | 1 |
force | forcefully | <adv> | <base> | d | 2 |
force | forceful | <adj> | <base> | d | 2 |
force | forcible | <adj> | <base> | d | 2 |
force | dynamic | <adj> | <base> | y | 3 |
```
Lexical Tools: **Inflect**

- **Generate inflectional variants**

> lvg -f:i SC SI

West Nile Virus

West Nile Virus | **West Nile virus** | <noun> | <base> | i | 1 |

West Nile Virus | **West Nile virus** | <noun> | <singular> | i | 1 |

West Nile Virus | **West Nile viruses** | <noun> | <plural> | i | 1 |
Lexical Tools: Inflect

Generate inflections, filter by cat and/or inflection

> lvg -f:ici~128+ALL -SC -SI

bioassay

bioassay | bioassay | <noun> | <base> | ici | 1 |
bioassay | bioassay | <noun> | <singular> | ici | 1 |
bioassay | bioassays | <noun> | <plural> | ici | 1 |
Lexical Tools: Uninflect by Term

Uninflect by term

> lvg -f:b -SC -SI

left atria

left atria | left atrium | <noun> | <base> | b | 1 |
Lexical Tools: Derivations

\textbf{-f:d Generate derivations}

\begin{verbatim}
> lvg -f:d -SC -SI
diagnostic
diagnostic | \textbf{diagnosis} | <noun> | <base> | d | 1 |
diagnostic | \textbf{diagnostics} | <noun> | <base> | d | 1 |
diagnostic | \textbf{diagnose} | <verb> | <base> | d | 1 |
diagnostic | \textbf{diagnostical} | <adj> | <base> | d | 1 |
\end{verbatim}
Lexical Tools: **Derivations**

- \textbf{-f:dc}\textasciitilde{cats}  \textbf{G}enerate derivations, filter by category

> lvg \textbf{-f:dc}\textasciitilde{129} -SC -SI
reduce
reduce | \textbf{reduction} | <noun> | <base> | d | 1 |
reduce | \textbf{reducible} | <adj> | <base> | d | 1 |
Lexical Tools: Synonyms

-f:y  Generate synonyms

> lvg -f:y -SC -SI

kidney

kidney | nephric | <adj> | <base> | y | 1 |
kidney | nephritic | <adj> | <base> | y | 1 |
kidney | renal | <adj> | <base> | y | 1 |
Lexical Tools: **Normalize** (norm)

- **f:N**

Remove stop words, then remove genitives, then replace punctuation with spaces, then lowercase, then uninflect each word, then take each of the uninflected words, then word order sort.

```
> lvg -f:N

Syndrome, Dry Eyes

Syndrome, Dry Eyes|**dry eye syndrome**|2047|1|g+o+t+l+B+w|1|
```
Lexical Tools: Installation

• Requirements
  • 1.6 gigabytes of space
  • Solaris/NT/Linux
  • Tar, gzip or WinZip
  • Minimum of 36 MB Memory

• Java JREs included
Lexical Tools: Installation

> ./install/bin/install_solaris_sparc.sh

> \install\bin\install_win
Lexical Tools: Installation

Java(TM) 2 runtime environment, Standard Edition, v 1.3.1_01

Software License Agreement

Please read the following License Agreement. Press the PAGE DOWN key to see the rest of the agreement.

Do you accept all the terms of the preceding License Agreement? If you choose No, Setup will
choose To install Java Runtime Environment, you must accept this agreement.

[Buttons: Back, Yes, No]
Lexical Tools: Installation

Welcome to the Java Lexical Tools Installation!
-------------------------------------------------------------------
Please read the installationNotes.html prior to invoking this script.

This script will configure the _LVG_DIR_/data/config/lvg.properties file.

This script will create configured lvg, norm, luiNorm, and wordind scripts in the _LVG_DIR_ directory.

This script, as an option, will configure and load the lexical tools into a pre-existing MySql database. This option is only available if you already have a MySql database running.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
The Java Lexical tools use or (may use) the following third party software packages:

- Instant DB from Enhydra.org (see http://instantdb.enhydra.org)
  This package is covered under an Enhydra Public License, which allows for redistribution and use with a "use notice".

- The installation script uses some GNU CYGWIN programs that have been distributed along with this script. These commands are only used when this installation is on a Windows platform.
  The commands that have been distributed include tar, gunzip, rm, find, hosthame, tee, and cmp.
  The CYGWIN package is licensed under the GNU Public License. The entire CYGWIN package, along with the sources to these commands can be found at http://sources.redhat.com/cygwin/

Enhydra is a trademark of Lutris Technologies, Inc.
Lexical Tools: Installation

+----------------------Note---------------------------+
|Although we do not support this feature, and it     |
|is very likely that we will change the underlying   |
|implementation, we have found it useful for our     |
|applications to use an existing database system     |
|other than IDB. We often use Mysql as the underlying |
|database for our applications. Since we needed a    |
|way to get the lexical tools data into MySql, we    |
|figured that we would incorporate the MYSQL loader  |
|into this install.                                  |
|                                                      |
|This option requires that an existing mysql database|
|(version 3.23 or higher) have already been installed,|
|that you have database root access privileges,      |
|that the database has enough room to load this      |
|data, and that the database is running now.         |
+-----------------------------------------------------+

+-------------------Question-------------------+
| Do you want to have the lexical tools use |    |
| your mysql database [y/n] [n]? n  |    |
Lexical Tools: Installation

+-------------------------------------------------------------------------------------+
| Verifying the installation ... | |
+-------------------------------------------------------------------------------------+

Enhydra InstantDB - Version 3.26

The Initial Developer of the Original Code is Lutris Technologies Inc. Portions created by Lutris are Copyright (C) 1997-2001 Lutris Technologies, Inc. All Rights Reserved.

Database sample is shutting down...

Database sample shutdown complete.
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Lexical Tools: Installation

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Congratulations!
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

This script has completed configuring the java lexical tools. You may invoke these tools from a command line. These tools are found in the __LVG_DIR__/bin directory. You can add this __LVG_DIR__/bin path to your $PATH environment variable. This would enable you to find and run these tools from any location. In UNIX, this would be done by adding this path to your ~/.cshrc or ~/.profile startup script.

In Windows, this would be done by appending this path to the PATH variable from the control panel/System/Environment variables menus.

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The lexical tools are ready to be used!
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Lexical Tools: Embedding These Tools into Your Application

- Classpath
- NormApi()
- LvgCmdApi()
- LexItem
- LvgLexItemApi()
Lexical Tools: Embedding Norm into Your Application
Lexical Tools: Embedding These Tools into Your Application

CLASSPATH = ${CLASSPATH}:
${LVG_DIR}:
${LVG_DIR}/classes/lvg2002.jar:
${LVG_DIR}/classes/IDB/jta-spec1_0_1.jar:
${LVG_DIR}/classes/IDB/idb.jar:
${LVG_DIR}/classes/jdbcDrivers/mm.mysql-2.0.6

If you are using IDB

If you are using MySQL
Lexical Tools:
Embedding Norm into Your Application

```java
import Lvg.Api.*;

NormApi normalize = new NormApi();
String input2Norm = null;
Vector outputFromNorm = null;
```
Lexical Tools:
Embedding Norm into Your Application

```java
while ( (input2Norm = stdIn.readLine() ) != null ) {
    outputFromNorm = normalize.Mutate(input2Norm);
    for ( int i = 0; i < outputFromNorm.size(); i++ ) {
        System.out.println((String) outputFromNorm.get(i));
    }
}
normalize.CleanUp();
```
Lexical Tools:
Embedding Lvg into Your Application

Your application

Input term → LvgCmd Api → Output terms

↓
import Lvg.A pi.*;

String input2Lvg = null;
Vector outputFromLvg = null;

Lexical Tools:
Embedding Lvg into Your Application
Lexical Tools: Embedding Lvg into Your Application

```java
while ( (input2Lvg = stdIn.readLine() ) != null ) {
    outputFromLvg = lvgApi.Mutate(input2Lvg);
    for ( int i = 0; i < outputFromLvg.size(); i++ ) {
        System.out.println((String) outputFromLvg.get(i));
    }
}
lvgApi.CleanUp();
```
Lexical Tools: Embedding Lvg into Your Application
Lexical Tools:
Embedding Lvg into Your Application

Your application

Input term

LvgLexItem Api

f:g:o:t:l:i

LexItems
Lexical Tools:
The LexItem Class
import Lvg.Api.*;
import Lvg.Lib.*;

LvgCmdApi lvgApi = new LvgLexItemApi("-f:g:o:t:l:i");
String input2Lvg = null;
Vector outputFromLvg = null;
LexItem aLexItem = null;

Lexical Tools:
Embedding Lvg into Your Application
while ((input2Lvg = stdIn.readLine()) != null) {
    outputFromLvg = lvgApi.MutateLexItem(input2Lvg);
    for (int i = 0; i < outputFromLvg.size(); i++) {
        aLexItem = (LexItem) outputFromLvg.get(i);
        System.out.println(aLexItem.GetSourceTerm() + "|" +
                            aLexItem.GetTargetTerm() + "|" +
                            aLexItem.GetTargetCategory().GetName() + "|" +
                            aLexItem.GetTargetInflection().GetName() + "|" +
                            aLexItem.GetTargetInflection().GetValue());
    }
}
lvgApi.Cleanup();
## Lexical Tools:
### Java API Documentation

<table>
<thead>
<tr>
<th>Packages</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Lvg.Api</strong></td>
<td>This package provides APIs</td>
</tr>
<tr>
<td><strong>Lvg.CmdLineSyntax</strong></td>
<td>Provides Java classes necessary to create a command line system.</td>
</tr>
<tr>
<td><strong>Lvg.Db</strong></td>
<td>Provides a higher level interface to LVG database.</td>
</tr>
<tr>
<td><strong>Lvg.Flows</strong></td>
<td>This package provides API of all Lvg flow components.</td>
</tr>
<tr>
<td><strong>Lvg.Lib</strong></td>
<td>Contains LVG general library classes of BitMaskBase, Category, Gender, Inflection, Configuration, CombineRecords, OutputFilter, Flow, GlobalBehavior, LexItem, LexItemComparator</td>
</tr>
<tr>
<td><strong>Lvg.Trie</strong></td>
<td>Provides the classes necessary to generate inflections, uninfections, and derivations using LVG rules tries</td>
</tr>
<tr>
<td><strong>Lvg.Util</strong></td>
<td>Contains LVG general utility classes of comparators, Bit operations, Case, In, Out, Strip operations, token operation, Char, Str, and Word.</td>
</tr>
</tbody>
</table>
# Lexical Resources

![UMLS Knowledge Source Server](image)

## Other UMLRS Resources

<table>
<thead>
<tr>
<th>Knowledge Sources</th>
<th>Source files for the Metathesaurus and Semantic Network and the SPECIALIST Lexicon.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Source Server API</td>
<td>The UMLS Knowledge Source Server API and command line interface.</td>
</tr>
<tr>
<td>Lexical Tools</td>
<td>The Lexical Tools are utilities which manipulate lexical data in order to abstract away from a various kinds of lexical variation: inflection, inversion, alphabetic case, etc.</td>
</tr>
</tbody>
</table>
References


McCray AT, Browne AC. Discovering the modifiers in a terminology data set. Proc AMIA Symp. 1998;:780-4


Using The Lexical Tools with The Metathesaurus

Metathesaurus

English

Strings

→ norm

Normalized string index

MRXNS.ENG

WordInd

Normalized word index

MRXNW.ENG
Using The Lexical Tools with The Metathesaurus

**Query**

- **norm**
- **Normalized string index**
- **Normalized word index**

**Metathesaurus Concepts**

- Concepts that match
- The normalized query
Using The Lexical Tools with The Metathesaurus

Dry Eyes Syndrome

Query \(\xrightarrow{\text{norm}}\) Normed Term \(\xrightarrow{}\) dry eye syndrome
Using The Lexical Tools with The Metathesaurus

Normed term

ENG dry eye syndrome C0013238|L 0013238|S0004019
ENG dry eye syndrome C0013238|L 0013238|S0035652
ENG dry eye syndrome C0013238|L 0013238|S0090228
ENG dry eye syndrome C0013238|L 0013238|S0090454
ENG dry eye syndrome C0013238|L 0013238|S0220550
ENG dry eye syndrome C0013238|L 0013238|S0368350
ENG dry eye syndrome C0013238|L 0013238|S1459074
Using The Lexical Tools with The Metathesaurus

<table>
<thead>
<tr>
<th>C0013238</th>
<th>ENG</th>
<th>P</th>
<th>L0013238</th>
<th>PF</th>
<th>S0035652</th>
<th>Dry Eye Syndromes</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>VS</td>
<td>S0004019</td>
<td>Dry eye syndrome</td>
</tr>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>VS</td>
<td>S0368350</td>
<td>Dry Eye Syndrome</td>
</tr>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>VS</td>
<td>S1459074</td>
<td>dry eye syndrome</td>
</tr>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>WS</td>
<td>S0090228</td>
<td>Syndrome, Dry Eye</td>
</tr>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>WS</td>
<td>S0220550</td>
<td>Dry, eye syndrome</td>
</tr>
<tr>
<td>C0013238</td>
<td>ENG</td>
<td>P</td>
<td>L0013238</td>
<td>VW</td>
<td>S0090454</td>
<td>Syndromes, Dry Eye</td>
</tr>
</tbody>
</table>
Building an Index Using The Lexical Tools

• Can we build a tool that increases recall?
• Can we build a tool that increases precision?
Building an Index Using The Lexical Tools

• Can we build a tool that increases precision?
  – Case
  – Constrain by part of speech
  – Filter to the lexicon
Building an Index Using The Lexical Tools

• Can we a tool that increases recall?
  - Include
    synonyms
    derivations
    acronyms and their expansions
    spelling variants
Building an Index Using The Lexical Tools

Metathesaurus Strings → index → Wordindex

word index
Building an Index Using The Lexical Tools

Query → Transformed query → Word index

SUIS

Metathesaurus Concepts that match The transformed query

Metathesaurus Concepts
Lexical Tools: 2003 and Beyond

• Performance issues addressed
• Graphic User Interface
  – Lexical GUI tool
  – On-line Web based tool
• Enhance flow components
  – Norm improved
• New flow components
  – Nominalize, Complementation
• Additional functionalities
  – XML output option
Lexical Tools for UMLS Developers

November 4, 2001
Allen C. Browne, Guy Divita, Chris Lu
Lister Hill National Center for Biomedical Communications
National Library of Medicine

Email: umlslex.nlm.nih.gov
Appendix

NormExample.java
LvgExampleEasy.java
LvgExampleHarder.java
LvgExampleEvenHarder.java