The SPECIALIST NLP Tools

Dr. Chris J. Lu

The Lexical Systems Group

NLM. LHNCBC. CGSB

Nov., 2011
NLP Tools

The SPECIALIST NLP Tools

The SPECIALIST NLP Tools

LexBuild

The SPECIALIST LEXICON

LexCheck

LexAccess

2VI3 Numbers

Lexical Tools

LVG-Derivations

Text Tools

NLP Projects

Table of Contents

• Introduction
• Lexical Variant Generation (LVG)
  ▪ Derivational Variant Generation
• Lexical Tools
  ▪ Norm
  ▪ Applications
• NLP Tools
• Questions
Lexicon To Lexical Tools

• How to use Lexicon (lexical records)?
• Lexical record(s) / Lexicon:
  ▪ Text
  ▪ Tables
• LexCheck package
  ▪ XML
  ▪ APIs: Java object(s)
Example: Spelling Variant

- Example: color & colour
- Lexical record in text & Java API:

```java
{base=color
 spelling_variant=colour
 entry=E0017902
   cat=noun
   variants=uncount
   variants=reg
 }
```

**Step 1: Convert to Java Object**
- LexRecord ToJavaObjFromText(String text)
- Vector&lt;LexRecord&gt; ToJavaObjsFromText(String text)
- Vector&lt;LexRecord&gt; ToJavaObjsFromTextFile(String inFile)

**Step 2: Retrieve information from LexRecord Java Object**
- Vector&lt;String&gt; GetSpellingVars()
- String GetBase()
- String GetCategory()

- LRSPL table:

<table>
<thead>
<tr>
<th>EUI</th>
<th>Spelling Variant</th>
<th>Base form</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>E0017902</td>
<td>colour</td>
<td>color</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Example: Inflectional Variant

- Lexical record in text & Java API:

```java
{base=color
  spelling_variant=colour
  entry=E0017902
  cat=noun
  variants=uncount
  variants=reg
}
```

- LRAGR table:

<table>
<thead>
<tr>
<th>EUI</th>
<th>Infl Var</th>
<th>Category</th>
<th>Agreement</th>
<th>Citation Form</th>
<th>Base Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>E0017902</td>
<td>color</td>
<td>noun</td>
<td>count(sing)</td>
<td>color</td>
<td>color</td>
</tr>
<tr>
<td>E0017902</td>
<td>color</td>
<td>noun</td>
<td>uncount(sing)</td>
<td>color</td>
<td>color</td>
</tr>
<tr>
<td>E0017902</td>
<td>colors</td>
<td>noun</td>
<td>count(plur)</td>
<td>color</td>
<td>color</td>
</tr>
<tr>
<td>E0017902</td>
<td>colour</td>
<td>noun</td>
<td>count(sing)</td>
<td>color</td>
<td>colour</td>
</tr>
<tr>
<td>E0017902</td>
<td>colour</td>
<td>noun</td>
<td>uncount(sing)</td>
<td>color</td>
<td>colour</td>
</tr>
<tr>
<td>E0017902</td>
<td>colours</td>
<td>noun</td>
<td>count(plur)</td>
<td>color</td>
<td>colour</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Lexical Variations (Lexicon)

- Spelling variants (-f:s):
  - color|colour (noun|E0017902, verb|E0017903)
  - grey|gray (adj|E0030394, noun|E0030395, verb|E0030396)
  - heart burn|heart-burn|heartburn (noun|E0030961)
  - hemostasis|haemostasis (noun|E0030684)

- Inflectional/uninflectional variants (-f:l, -f:b, -f:B):
  - heart burn (noun|E0030961)
  - color|colors (noun|E0017902)
  - color|colored|colored|colors|coloring (verb|E0017903)
  - see|sees|saw|seen|seeing (verb|E0055007)
  - saw|saws|sawed|sawn|sawing (verb|E0054444)

  - ER|emergency room
  - ER|enhancement ratio
  - ER|eye research
  - 20+ known acronyms …

- Nominalization (-f:nom):
  - active|adj|activity|noun
  - active|adj|activeness|noun

- ProperNoun (-f:fp):
  - Clinton
  - Virginia
  - University of Virginia
Basic Variations

- Lowercase (-f:l):
  - AIDS|aids
  - ÀÁÂÄÄ ÊÉÊ ÈÓÔÔÔ Ô ÙÚÙÜ|àáâãäåèéêëìíîïòóôõöøùúûü
- Strip punctuation (-f:o, -f:p, -f:P)
  - St. John's|St Johns
- Strip stopwords (-f:t)
  - Remove "of", "and", "with", "for", "nos", "to", "in", "by", "on", "the", "(non mesh)", etc.
  - Academy of Physical Medicine|Academy Physical Medicine
- Remove genitive (-f:g):
  - Down's Syndrome|Down Syndrome
- Remove parenthetical plural form of (s), (es), (ies) (-f:rs)
  - Burn(s);skin|Burn;skin
  - 9(s)-erythromycylamine|9(s)-erythromycylamine
- Strip ambiguity tags (-f:T)
  - cold <1>|cold
- Sort words (-f:w)
  - Cancer, Lung|Cancer Lung
  - Lung Cancer|Cancer Lung
- Word size filter (-f:ws)
  - Academy of Physical Medicine|Academy Physical Medicine
- …
Others Variations

- **Derivational variants (-f:d, -f:R):**
  - gene|noun|genetic|adj
  - gene|noun|genic|adj
  - hyperuricemic|adj|hyperuricemia|noun
  - hyperplastic|adj|hyperplasia|noun

- **Synonyms (-f:y):**
  - otitis|ear inflammation (C0029877)
  - kidney|renal| (C0022646)
  - eartburn|pyrosis|brash (C0018834)

- **Canonical Form (-f:C):** used in LuiNorm for Lui assignment
  - color|color
  - colour|color
  - colored|color
  - coloued|color

- ...
Complicated Variations

- **ASCII Conversion** (-f:q, -f:q0, -f:q1, … -f:q8):
  - resumé|resume
  - spælsau|spaelsau
  - \( \frac{\%}{8} \)|5/8
  - “Quote” |"Quote"
  - α-Best™|alpha-Best
  - …

- **Norm (-f:N):**
  - Hodgkin's diseases, NOS|disease hodgkin
  - proofread|proof read
  - proof-read|proof read
  - proof read|proof read
  - left|left
  - left|leave

- **LuiNorm (-f:N3):**
  - left|leaf

- **AntiNorm (Use for Approximate match in Lexicon):**
  - Abrami disease|Abrami's disease
  - Abrami disease|Abrami's diseases

- …
Lexical Variant Generation (LVG)
Lexical Tools - LVG

- A suite of text utilities
Lexical Tools - LVG

- A suite of text utilities take the given input
Lexical Tools - LVG

- A suite of text utilities that generate, mutate, and filter out lexical variants from the given input
LVG - 2012

• 62 flow components
• 37 options
  – input filter options (3)
  – global behavior options (12)
  – flow specific options (2)
  – output filter options (20)
Flow Components

leave ➔ inflect ➔ leaves, leaving, left
shell> 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|
> lvg –f:i
leave

Fielded Output

Input Term

Output Term

Categories

Inflections

Flow history

Flow Number
A Serial Flow

- Flow components can be arranged so that the output of one is the input to another.
A Serial Flow - Example

shell> lvg –f:l:q:g:t:p:w
The Gougerot-Sjögren's Syndrome
The Gougerot-Sjögren's Syndrome |
gougerotsjogren syndrome |
2047|16777215|l+q+g+t+p+w|1|
• Multiple flows can be defined
### Parallel Flows - Example

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

| 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 |
```
Input Filter Options

Input term

Output terms

Take field 7 from the input

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

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

acute Rheumatic carditis|S0003894
Global Behavior Options

> lvg -f:L -f:E

_change separator to “\”_

_“otitis”_

_“otitis\otitis\128\513\L\1”_

_“otitis\E0044452\128\513\E\2”_
Output Filter Options

> lvg -f:L | -SC -SI

Show the category and inflection names

hot

hot|hot|<adj+verb>|<base+positive+infinitive+pres1p23p>|L|1|
Derivational Variants

• **Words** related by a derivational process
  - Derivational process: suffix and prefix
  - Used to create new words based on existing words
  - Meaning change
  - Category change

• Focus on relatedness (no direction)
Derivations Application

- hyperuricemic|adj, E0317343, no CUI
- hyperuricemia|noun, E0032862, is a UMLS Metathesaurus term (C0740394)
Derivational Network

kind|noun

kind|adj

kindly|adv

kindliness|noun

unkind|adj

unkindly|adv

unkindliness|noun

kindness|noun

unkindness|noun
Derivational Pair

- Kindness | Noun
- Unkindness | Noun
- Kind | Noun
- Unkind | Adjective
- Kindly | Adverb
- Unkindly | Adverb
- Kindliness | Noun
- Unkindliness | Noun
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
Suffix Derivation (SD) Pair

kind|noun

kind|adj

kindly|adv

kindliness|noun

unkind|adj

unkindly|adv

unkindliness|noun
Prefix Derivation (PD) Pair

kind|noun

kind|adj

kindly|adv

kindliness|noun

unkind|adj

unkindly|adv

unkindliness|noun

kind|noun

unkindness|noun

unkindness|noun
Zero Derivation (ZD) Pair

kind|noun

kind|adj

kindly|adv

kindliness|noun

unkindness|noun

unkind|adj

unkindly|adv

unkindliness|noun
Derivational Analysis (Tagging)

- Performed by linguistic experts
- Is complicated when more than one affix involved
  - look at usage of all related words
  - peel off the derivational affixes
  - check if they are valid words
  - determine the order of derivation
  - multi-option-al, pseudo-hyper-para-thyroid-ism

![Diagram of derivational analysis]
Derivational Pair & Tag

- Format: base 1|category 1|base 2|category 2|Tag
- Examples:
  - kindness|noun|kind|adj|yes
  - unkindly|adv|unkindliness|noun|yes
  - kindness|noun|kindly|adj|no
  - kindness|noun|unkindness|noun|no
Derivational Pair & Tag

- Format: base 1|category 1|base 2|category 2|Tag
- Examples:
  - kindness|noun|kind|adj|yes
  - unkindly|adv|unkindliness|noun|yes
  - kindness|noun|kindly|adj|no
  - kindness|noun|unkindness|noun|no
Derivational Pair & Tag

- Format: base 1|category 1;base 2|category 2<Tag
- Examples:
  - kindness|noun|kind|adj|yes
  - unkindly|adv|unkindliness|noun|yes
  - kindness|noun|kindly|adj|no
  - kindness|noun|unkindness|noun|no
Derivational Pair & Tag

- Format: base 1|category 1|base 2|category 2|Tag
- Examples:
  - kindness|noun|kind|adj|yes
  - unkindly|adv|unkindliness|noun|yes
  - kindness|noun|kindly|adj|no
  - kindness|noun|unkindness|noun|no
Derivational Flows in LVG

• Direct derivation generation (-f:d)
  ▪ All valid derivational pairs associated with the node
  ▪ Example:
    4 derivational variants of kind|adj are found:
      kind|noun, kindness|noun, kindly|adv, and unkind|adj

• Recursive derivation generation (-f:R)
  ▪ Entire derivational network
  ▪ Also provides the distance (number of derivational pairs involved). For example, 2 for kindness|noun and kindly|adv
Derivational Flow

• Facts
  ▪ 4,559 derivational pairs (2011)

<table>
<thead>
<tr>
<th>Base 1</th>
<th>Category 1</th>
<th>Base 2</th>
<th>Category 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>treatment</td>
<td>noun</td>
<td>treat</td>
<td>noun</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

• Rules
  ▪ 97 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;
...
• retirement |noun=> retire|verb
SD-Rules Filters

- **Exception filter**
  - Exclude exceptions for the rules
  - Implemented in the Trie
  - depart|verb|department|noun

- **Word length filter**
  - Exclude short word
  - Default (min.) value is 3
  - moment|noun|mo|verb

- **Stem length filter**
  - stem length = word length – suffix length
  - Default (min.) value is 3
  - lament|noun|la|verb

- **Domain filter**
  - Exclude words not in Lexicon
  - color|verb|colorment|noun
Derivations - Evaluation

• Facts
  - 4,559 derivational pairs (2011)
  - Maintenance: collecting, validating, and tagging
  - Has not grown proportionally with Lexicon …
  - Prefix derivation & zero derivation (conversion) ?

• Rules
  - 97 SD-Rules
  - High frequency?
  - High precision?
  - Prefix derivation & zero derivation rules?
Challenges

• Facts
  ▪ More coverage: include zeroD, prefixD, suffixD
  ▪ Grows proportionally with Lexicon
  ▪ Higher precision

• Rules
  ▪ Evaluate frequency and precision for SD-Rules
  ▪ Include PD-Rules?
  ▪ Include ZD-Rules?
Zero Derivation (ZD)

- Also called conversion or functional shift
- assigns an existing word to a new syntactic category without any concomitant change in form
- ZD Pairs:
  - kind|noun|kind|adj|yes
  - flex|noun|flex|verb|yes
  - round|adj|round|prep|no
ZD Process

- Retrieve base forms (citation & spelling variants) and category information from Lexicon
- Raw ZD pairs: all words with multiple categories
- Filter programs:
  - Min. Word length (< 2):
    - Example: a|noun|a|det|no
      - a|noun: abbreviations for 50+ nouns, such as abortion, acid, adult, …
  - Exclude abbreviations and acronyms
    - Example: AAIR|noun|AAIR|adj |no
      - AAIR|noun: age-adjusted incidence rate
      - AAIR|adj: rate-adaptive atrial
- Final tagging
ZD Results

- Raw ZD pairs: 18,400
- Filtered: 10.52%
- Tagged: 89.48% (recorded for future release)
- No ZD-Rules
  - Valid rate: 80.14%
  - Invalid: 1,718

<table>
<thead>
<tr>
<th></th>
<th>ZD Counts</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw</td>
<td>18,400</td>
<td>100.00%</td>
</tr>
<tr>
<td>Filtered</td>
<td>1,935</td>
<td>10.52%</td>
</tr>
<tr>
<td>Tag - Invalid</td>
<td>1,718</td>
<td>9.34%</td>
</tr>
<tr>
<td>Tag - Valid</td>
<td>14,747</td>
<td>80.14%</td>
</tr>
</tbody>
</table>
Prefix Derivation (PD)

• Placed at the beginning of a base word to form another word
• Three patterns:
  ▪ **prefix**: significant | adj nonsignificant | adj
  ▪ **prefix and a dash**: significant | adj non-significant | adj
  ▪ **prefix and a space**: significant | adj non-significant | adj
• PD pairs:
  ▪ unkind | adj | kind | adj | yes
  ▪ kindness | noun | unkindness | noun | no
  ▪ unplug | verb | plug | noun | no
  ▪ touchable | adj | untouchable | adj | yes
  ▪ touchable | adj | untouchable | noun | yes
PD Process

- Collects common derivational prefixes (143)
- Retrieve all base forms from Lexicon
- Raw PD pairs: match three prefix patterns
  - prefix: nonsignificant|adj|significant|adj
  - prefix and a dash: non-significant|adj|significant|adj
  - prefix and a space: non significant|adj|significant|adj
- Final tagging
  - Tag the most frequent and user requested prefixes
PD Results

- Raw PD pairs: 115,139
- Tagged: 74.98%
  recorded for future release
- No PD-Rules
  ▪ Avg. 65.67% valid rate
  ▪ Max. 80.31% valid rate
- No Category filter:
  ▪ 24.54% of valid PD pairs changes category
    - fog|noun|antifog|adj|yes
- No acronym or abbreviation filter:
  ▪ 0.83% of valid PD pairs are acronyms or abbreviations
    - MDR|noun|antiMDR|adj|yes
      MDR, acronym for “multidrug resistance”
Suffix Derivation (SD)

- Also called a postfix or ending
- Placed after the stem of a word to form another word
- Several hundreds of derivational suffixes
- Collects common derivational suffixes (200)
- SD Pairs:
  - kind|adj|kindness|noun
  - kind|adj|kindly|adv
SD Process - Nominalization

- The process of producing a noun from a verb or an adjective via the derivational suffix
- Coded in Lexicon
- A type of suffix derivation
- Bi-directional

```plaintext
{base=locate
drop=entry=E0037939
cat=verb
variants=reg
tran=np
link=advbl
cplxtran=np,advbl
nominalization=location | noun | E0037940}
```

```plaintext
{base=location
drop=entry=E0037940
cat=noun
variants=reg
variants=uncount
compl=pphr(of,np)
compl=pphr(by,np)
nominalization_of=locate | verb | E0037939}
```
SD Process - ND

- Raw ND pairs: retrieve all nominalization information from Lexicon
- Filters:
  - Pattern filter: exclude invalid SD for verb particle ND
    - **Pattern-1**: baseParticle|noun|base|verb => backup|noun|back|verb
    - **Pattern-2**: base-Particle|noun|base|verb => cut-through|noun|cut|verb
    - **Pattern-3**: inflParticle|noun|base|verb => grownup|nou|grow|verb
    - **Pattern-4**: infl-Particle|noun|base|verb => salting-in|noun|salt|verb
  - **Particle Exception**: “per” => shopper|noun|shop|verb
  - Exception filter: exclude other known SD pairs
    - Examples:
      - face-saving|noun|save|verb
      - decision-making|noun|make|verb
      - merry-making|noun|make|verb
      - lovemaking|noun|make|verb
      - ...

ND (SD) Results

- Raw ND pairs: 14,445
- Filtered: 0.50%
- Valid: 99.50 % ND pairs (program generated)

<table>
<thead>
<tr>
<th>ND Pairs</th>
<th>Filtered</th>
<th>Valid</th>
</tr>
</thead>
<tbody>
<tr>
<td>14,445</td>
<td>72</td>
<td>14,373</td>
</tr>
<tr>
<td>100%</td>
<td>0.50%</td>
<td>99.50%</td>
</tr>
</tbody>
</table>
ND to SD-Rules

- Identified SD-Rules from Valid ND pairs
  - 496 possible rules are found
  - `location|noun|locate|verb` => `ion$|noun|e$|verb`
- Further analysis

<table>
<thead>
<tr>
<th>Derivation Suffix Rules</th>
<th>Example</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>ation$</td>
<td>noun</td>
<td>ate$</td>
</tr>
<tr>
<td>sion$</td>
<td>noun</td>
<td>se$</td>
</tr>
<tr>
<td>ution$</td>
<td>noun</td>
<td>ute$</td>
</tr>
<tr>
<td>etion$</td>
<td>noun</td>
<td>ete$</td>
</tr>
<tr>
<td>otion$</td>
<td>noun</td>
<td>ote$</td>
</tr>
<tr>
<td>ition$</td>
<td>noun</td>
<td>ite$</td>
</tr>
<tr>
<td>otion$</td>
<td>noun</td>
<td>ote$</td>
</tr>
</tbody>
</table>

- Map with existing SD-Rules in LVG

<table>
<thead>
<tr>
<th>Identified Rules</th>
<th>Rules in Lexical Tools</th>
<th>Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>ness$</td>
<td>noun</td>
<td>$</td>
</tr>
<tr>
<td>ion$</td>
<td>noun</td>
<td>e$</td>
</tr>
<tr>
<td></td>
<td>sion$</td>
<td>noun</td>
</tr>
<tr>
<td></td>
<td>Others ...</td>
<td>70</td>
</tr>
<tr>
<td>ity$</td>
<td>noun</td>
<td>$</td>
</tr>
<tr>
<td></td>
<td>icity$</td>
<td>noun</td>
</tr>
<tr>
<td>ility$</td>
<td>noun</td>
<td>le$</td>
</tr>
<tr>
<td></td>
<td>Others ...</td>
<td>253</td>
</tr>
<tr>
<td>ation$</td>
<td>noun</td>
<td>e$</td>
</tr>
</tbody>
</table>
Final Compile

• Final affix validation program:
  ▪ Affix check: check the first and last (3) characters between base forms of derivational pairs to assure only one affix is involved.
  ▪ Exception filter:
    o able|adj|ability|noun
    o long|adj|length|noun
    o high|adj|height|noun
    o ...
  ▪ Spelling variants
    o dysmaturity|noun|dismature|adj
    o gray|adj|grey|noun
    o haemolysed|adj|hemolyzation|noun
    o ...

• Combine all three lists (ZD, PD, ND)
Final Results

• More coverage (will grow with Lexicon)

<table>
<thead>
<tr>
<th>2011 Lvg Facts</th>
<th>2012 Lvg Facts</th>
</tr>
</thead>
<tbody>
<tr>
<td>4,559</td>
<td>89,950</td>
</tr>
</tbody>
</table>

Derivation Pairs Distribution

- PrefixD: 66%
- NomD: 17%
- ZeroD: 17%

• Virtually 100% precision rate (Facts)
Future Work

• ZeroD:
  ▪ Rules-based filter: syntactic category and other linguistic knowledge
  ▪ PD-Rules: syntactic category and other linguistic knowledge

• PrefixD:
  ▪ Update prefix list annually
  ▪ Complete tagging processes for all collected prefix
  ▪ Rules-based filter: syntactic category and other linguistic knowledge
  ▪ PD-Rules: syntactic category and other linguistic knowledge

• SuffixD:
  ▪ Develop a thorough validation process for existing SD-Rules by all possible raw SD pairs in the Lexicon
  ▪ Find all exceptions for each SD-Rules in Lexicon
  ▪ Rules-based filter: syntactic category and other linguistic knowledge
  ▪ SD-Rules: syntactic category and other linguistic knowledge
Questions

Lexical Tools

http://SPECIALIST.nlm.nih.gov/lvg
Lexical Tools - Six Tools

Input → Lvg Norm LuiNorm WordIndex ToAScii Fields → Output...
  Output.3
  Output.2
  Output.1
Lexical Tools - Types

• Command line tools
  – lvg (Lexical Variants Generation)
  – norm
  – luiNorm
  – wordInd
  – toAscii
  – fields
• Lexical Gui Tool (lgt)
• Web Tools
• Java API’s
Functions

• Used in nature language processing for:
  – aggressive text pattern matching
  – creating normalized and expanded terms
  – making word, term, phrase indexes
  – matching queries with indexed entries
  – increasing recall and/or precision
Facts

• Release annually
• Free distributed with open source code
• 100% Java (since 2002)
• Run on different platforms
• One complete package
• Documents & supports
Norm

• Composed of 11 Lvg flow components to abstract away from:
  – case
  – punctuation
  – possessive forms
  – inflections
  – spelling variants
  – stop words
  – Diacritics, ligatures & symbols (Unicode to ASCII)
  – word order
<p>| q0: map Unicode symbols to ASCII |
| g: remove genitives |
| rs: remove parenthetic plural forms |
| o: replace punctuation with spaces |
| t: strip stop words |
| l: lowercase |
| B: uninflect each words in a term |
| Ct: retrieve citations |
| q7: Unicode core Norm |
| q8: strip or map non-ASCII char |
| w: sort words by order |</p>
<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>q0</code></td>
<td>map Unicode symbols to ASCII</td>
</tr>
<tr>
<td><code>g</code></td>
<td>remove genitives</td>
</tr>
<tr>
<td><code>rs</code></td>
<td>remove parenthetic plural forms</td>
</tr>
<tr>
<td><code>o</code></td>
<td>replace punctuation with spaces</td>
</tr>
<tr>
<td><code>t</code></td>
<td>strip stop words</td>
</tr>
<tr>
<td><code>l</code></td>
<td>lowercase</td>
</tr>
<tr>
<td><code>B</code></td>
<td>uninflect each words in a term</td>
</tr>
<tr>
<td><code>Ct</code></td>
<td>retrieve citations</td>
</tr>
<tr>
<td><code>q7</code></td>
<td>Unicode core Norm</td>
</tr>
<tr>
<td><code>q8</code></td>
<td>strip or map non-ASCII char</td>
</tr>
<tr>
<td><code>w</code></td>
<td>sort words by order</td>
</tr>
</tbody>
</table>

Hodgkin's Diseases, NOS
<p>| q0: map Unicode symbols to ASCII | Hodgkin's Diseases, NOS |
| g: remove genitives | Hodgkin's Diseases, NOS |
| rs: remove parenthetic plural forms | |
| o: replace punctuation with spaces | |
| t: strip stop words | |
| l: lowercase | |
| B: uninflect each words in a term | |
| Ct: retrieve citations | |
| q7: Unicode core Norm | |
| q8: strip or map non-ASCII char | |
| w: sort words by order | |</p>
<table>
<thead>
<tr>
<th>q0: map Unicode symbols to ASCII</th>
<th>Hodgkin's Diseases, NOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>q7: Unicode core Norm</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>q8: strip or map non-ASCII char</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>B: uninflct each words in a term</td>
<td></td>
</tr>
<tr>
<td>Ct: retrieve citations</td>
<td></td>
</tr>
<tr>
<td>g: remove genitives</td>
<td></td>
</tr>
<tr>
<td>o: replace punctuation with spaces</td>
<td></td>
</tr>
<tr>
<td>t: strip stop words</td>
<td></td>
</tr>
<tr>
<td>I: lowercase</td>
<td></td>
</tr>
<tr>
<td>rs: remove parenthetic plural forms</td>
<td></td>
</tr>
<tr>
<td>w: sort words by order</td>
<td></td>
</tr>
</tbody>
</table>

**Norm**
<table>
<thead>
<tr>
<th>q0: map Unicode symbols to ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>g: remove genitives</td>
</tr>
<tr>
<td>rs: remove parenthetic plural forms</td>
</tr>
<tr>
<td>o: replace punctuation with spaces</td>
</tr>
<tr>
<td>t: strip stop words</td>
</tr>
<tr>
<td>l: lowercase</td>
</tr>
<tr>
<td>B: uninflect each words in a term</td>
</tr>
<tr>
<td>Ct: retrieve citations</td>
</tr>
<tr>
<td>q7: Unicode core Norm</td>
</tr>
<tr>
<td>q8: strip or map non-ASCII char</td>
</tr>
<tr>
<td>w: sort words by order</td>
</tr>
</tbody>
</table>

---

Hodgkin's Diseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases, NOS
<p>| <strong>Norm</strong> |
|-----------------|-----------------|
| q0: map Unicode symbols to ASCII | Hodgkin's Diseases, NOS |
| g: remove genitives | Hodgkin's Diseases, NOS |
| rs: remove parenthetical plural forms | Hodgkin Diseases, NOS |
| <strong>o: replace punctuation with spaces</strong> | Hodgkin Diseases, NOS |
| t: strip stop words | Hodgkin Diseases, NOS |
| l: lowercase | <strong>Hodgkin Diseases, NOS</strong> |
| B: uninflect each words in a term | |
| Ct: retrieve citations | |
| q7: Unicode core Norm | |
| q8: strip or map non-ASCII char | |
| w: sort words by order | |</p>
<table>
<thead>
<tr>
<th>q0: map Unicode symbols to ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>g: remove genitives</td>
</tr>
<tr>
<td>rs: remove parenthetic plural forms</td>
</tr>
<tr>
<td>o: replace punctuation with spaces</td>
</tr>
<tr>
<td>t: strip stop words</td>
</tr>
<tr>
<td>l: lowercase</td>
</tr>
<tr>
<td>B: uninflect each words in a term</td>
</tr>
<tr>
<td>Ct: retrieve citations</td>
</tr>
<tr>
<td>q7: Unicode core Norm</td>
</tr>
<tr>
<td>q8: strip or map non-ASCII char</td>
</tr>
<tr>
<td>w: sort words by order</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hodgkin's Diseases, NOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>Hodgkin Diseases NOS</td>
</tr>
<tr>
<td>Hodgkin Diseases</td>
</tr>
</tbody>
</table>
### Norm

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>q0</td>
<td>map Unicode symbols to ASCII</td>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>g</td>
<td>remove genitives</td>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>rs</td>
<td>remove parenthetic plural forms</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>o</td>
<td>replace punctuation with spaces</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>t</td>
<td>strip stop words</td>
<td>Hodgkin Diseases NOS</td>
</tr>
<tr>
<td>l</td>
<td>lowercase</td>
<td>Hodgkin Diseases</td>
</tr>
<tr>
<td>B</td>
<td>uninflect each words in a term</td>
<td>hodgkin diseases</td>
</tr>
<tr>
<td>Ct</td>
<td>retrieve citations</td>
<td></td>
</tr>
<tr>
<td>q7</td>
<td>Unicode core Norm</td>
<td></td>
</tr>
<tr>
<td>q8</td>
<td>strip or map non-ASCII char</td>
<td></td>
</tr>
<tr>
<td>w</td>
<td>sort words by order</td>
<td></td>
</tr>
<tr>
<td>Operation</td>
<td>Description</td>
<td>Result</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>q0:</td>
<td>map Unicode symbols to ASCII</td>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>g:</td>
<td>remove genitives</td>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>rs:</td>
<td>remove parenthetic plural forms</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>o:</td>
<td>replace punctuation with spaces</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>t:</td>
<td>strip stop words</td>
<td>Hodgkin Diseases NOS</td>
</tr>
<tr>
<td>l:</td>
<td>lowercase</td>
<td>Hodgkin Diseases</td>
</tr>
<tr>
<td>B:</td>
<td>uninflect each words in a term</td>
<td>Hodgkin Disease</td>
</tr>
<tr>
<td>Ct:</td>
<td>retrieve citations</td>
<td>Hodgkin disease</td>
</tr>
<tr>
<td>q7:</td>
<td>Unicode core Norm</td>
<td>Hodgkin disease</td>
</tr>
<tr>
<td>q8:</td>
<td>strip or map non-ASCII char</td>
<td>Hodgkin disease</td>
</tr>
<tr>
<td>w:</td>
<td>sort words by order</td>
<td>Hodgkin disease</td>
</tr>
</tbody>
</table>
Norm

q0: map Unicode symbols to ASCII
q7: Unicode core Norm
q8: strip or map non-ASCII char
w: sort words by order

g: remove genitives
rs: remove parenthetic plural forms
o: replace punctuation with spaces
t: strip stop words
l: lowercase
B: uninflect each words in a term
Ct: retrieve citations

Hodgkin's Diseases, NOS
Hodgkin's Diseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases NOS
Hodgkin Diseases
hodgkin diseases
hodgkin disease
hodgkin disease
<table>
<thead>
<tr>
<th>q0: map Unicode symbols to ASCII</th>
<th>Hodgkin's Diseases, NOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>g: remove genitives</td>
<td>Hodgkin's Diseases, NOS</td>
</tr>
<tr>
<td>rs: remove parenthetic plural forms</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>o: replace punctuation with spaces</td>
<td>Hodgkin Diseases, NOS</td>
</tr>
<tr>
<td>t: strip stop words</td>
<td>Hodgkin Diseases NOS</td>
</tr>
<tr>
<td>l: lowercase</td>
<td>Hodgkin Diseases</td>
</tr>
<tr>
<td>B: uninflect each words in a term</td>
<td>Hodgkin diseases</td>
</tr>
<tr>
<td>Ct: retrieve citations</td>
<td>Hodgkin disease</td>
</tr>
<tr>
<td>q7: Unicode core Norm</td>
<td>Hodgkin disease</td>
</tr>
<tr>
<td>q8: strip or map non-ASCII char</td>
<td></td>
</tr>
<tr>
<td>w: sort words by order</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>q0</td>
<td>Map Unicode symbols to ASCII</td>
</tr>
<tr>
<td>g</td>
<td>Remove genitives</td>
</tr>
<tr>
<td>rs</td>
<td>Remove parenthetic plural forms</td>
</tr>
<tr>
<td>o</td>
<td>Replace punctuation with spaces</td>
</tr>
<tr>
<td>t</td>
<td>Strip stop words</td>
</tr>
<tr>
<td>l</td>
<td>Lowercase</td>
</tr>
<tr>
<td>B</td>
<td>Uninflect each words in a term</td>
</tr>
<tr>
<td>Ct</td>
<td>Retrieve citations</td>
</tr>
<tr>
<td>q7</td>
<td>Unicode core Norm</td>
</tr>
<tr>
<td>q8</td>
<td>Strip or map non-ASCII char</td>
</tr>
<tr>
<td>w</td>
<td>Sort words by order</td>
</tr>
</tbody>
</table>

The process involves several steps to normalize the text, including mapping Unicode symbols to ASCII, removing genitives, removing parenthetic plural forms, replacing punctuation with spaces, stripping stop words, and converting to lowercase. The final result is a normalized form of the input text, in this case, Hodgkin disease.
q0: map Unicode symbols to ASCII
q7: Unicode core Norm
q8: strip or map non-ASCII char
w: sort words by order

g: remove genitives
rs: remove parenthetic plural forms
o: replace punctuation with spaces
t: strip stop words
l: lowercase
B: uninflect each words in a term
Ct: retrieve citations

Hodgkin's Diseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases, NOS
Hodgkin Diseases NOS
Hodgkin Diseases
hodgkin diseases
hodgkin disease
hodgkin disease
hodgkin disease
hodgkin disease
hodgkin disease
disease hodgkin
Norm: Example

- 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
- ...

- disease hodgkin
Normalize and Index

Terms in Corpus

normalize

Index

Indexed Database - Normalized String
Normalize and Index

Input String

Normalized String

SQL Query

Terms in Corpus

normalize

Index

Results

Indexed Database - Normalized String
Questions

NLP & NLP Tools

• Natural Language
  ▪ 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
Information Retrieval

Input String

- Tokenize
- Spelling Check
- Lexical Variants
- POS Tagging
- Semantic Knowledge

NLP System/Database/Search Engine
Core NLP Tasks

• Ex: Web search engine for biomedical information
  ▪ Software:
    o keyword search
      ✓ break inputs into words
      ✓ POS tagging
      ✓ other annotation
    o spelling check
      ✓ suggest correct spelling for misspelled words
    o lexical variants
      ✓ spelling variants, inflectional/uninflectional variants, synonyms, acronyms/abbreviations, expansions, derivational variants, etc.
    o semantic knowledge
      ✓ map text to Metathesaurus concepts
      ✓ Word Sense Disambiguation (WSD)
  ▪ Data:
    o corpus: annotation/tagging
NLP Tools

• Ex: Web search engine for biomedical information
  ▪ **Software:**
    o keyword search
      - break inputs into words *(Text Tools)*
      - POS tagging *(dTagger)*
      - Other annotation *(Visual Tagging Tool, VTT)*
    o spelling check
      - suggest correct spelling for misspelled words *(gSpell)*
    o lexical variants
      - spelling variants, inflectional/uninflectional variants, synonyms, acronyms/abbreviations, expansions, derivational variants, etc. *(Lexical Tools)*
    o semantic knowledge
      - map text to Metathesaurus concepts *(MetaMap, MMTX)*
      - Word Sense Disambiguation *(TC - StWSD)*
  ▪ **Data: corpus**
    o annotation/tagging *(Text Tools, dTagger, VTT, Lexical Tools)*
Core NLP Tools

• Ex: Web Search Engine for biomedical information
  ▪ Software:
    o keyword search
      ➢ break inputs into words (Text Tools)
      ➢ POS tagging (dTagger)
      ➢ Other annotation (Visual Tagging Tool, VTT)
    o spelling check
      ➢ suggest correct spelling for misspelled words (gSpell)
    o lexical variants
      ➢ spelling variants, inflectional/uninflectional variants, synonyms, acronyms/abbreviations, expansions, derivational variants, etc. (Lexical Tools)
    o semantic knowledge
      ➢ map text to Metathesaurus concepts (MetaMap, MMTX)
      ➢ Word Sense Disambiguation (TC - StWSD)
  ▪ Data: corpus
    o annotation/tagging (Text Tools, dTagger, VTT, Lexical Tools)
The SPECIALIST NLP Tools

The SPECIALIST LEXICON

LexBuild

LexCheck

LexAccess

Lexical Tools

Text Tools

Lexical Tool

NLP Projects

SCRT

The SPECIALIST LEXICON
Questions