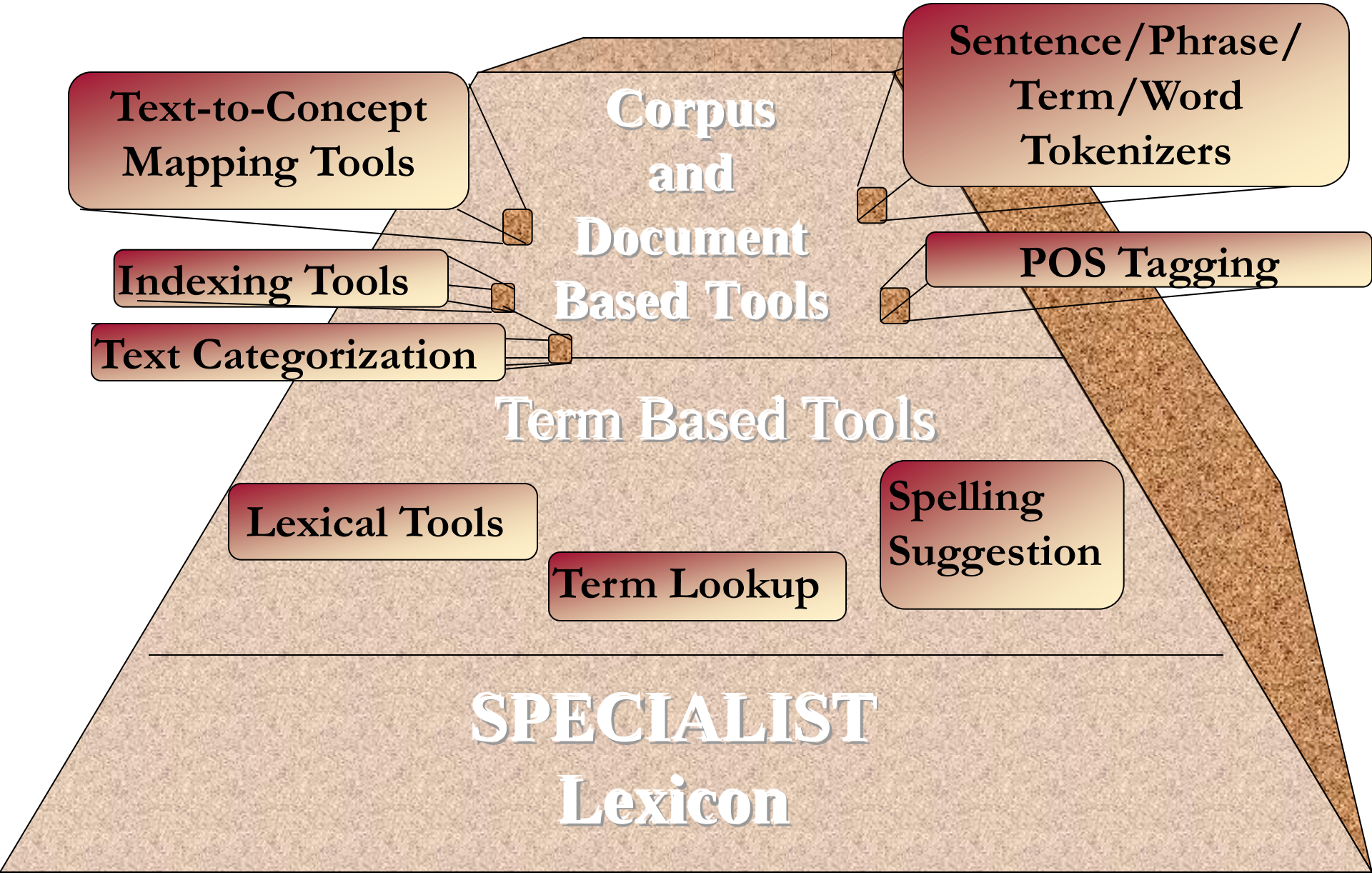
Guy Divita
Chris Lu

THE LISTER HILL NATIONAL CENTER
FOR BIOMEDICAL COMMUNICATIONS
A Research Division of the U.S. National Library of Medicine


Text-to-Concept Mapping Tools
Sentence/Phrase/ Term/Word Tokenizers
Corpus and Document Based Tools
Indexing Tools
POS Tagging
Text Categorization
Term Based Tools
Lexical Tools
Spelling Suggestion
Term Lookup
SPECIALIST Lexicon

SPECIALIST.nlm.nih.gov

About

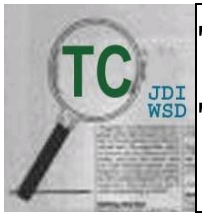
Projects

Documents

The Lexicon

Document Tokenization Tools

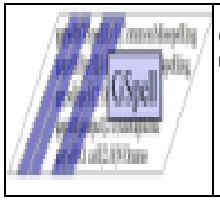
Lexicon Term Lookup



Text Categorization Tool

Term Manipulation Tools

POS Tagger



Spelling Suggestion

MetaMap Transfer (MMTx)
mmtx.nlm.nih.gov

LexAccess Web Tool

JSP, UTF-8, 2007

Home | Web Tool | Documents | Tutorial | Contact Us | Releases | About

Search Options: By Term/Eui | By Base | By Category | Output Format | More | Version | Reset

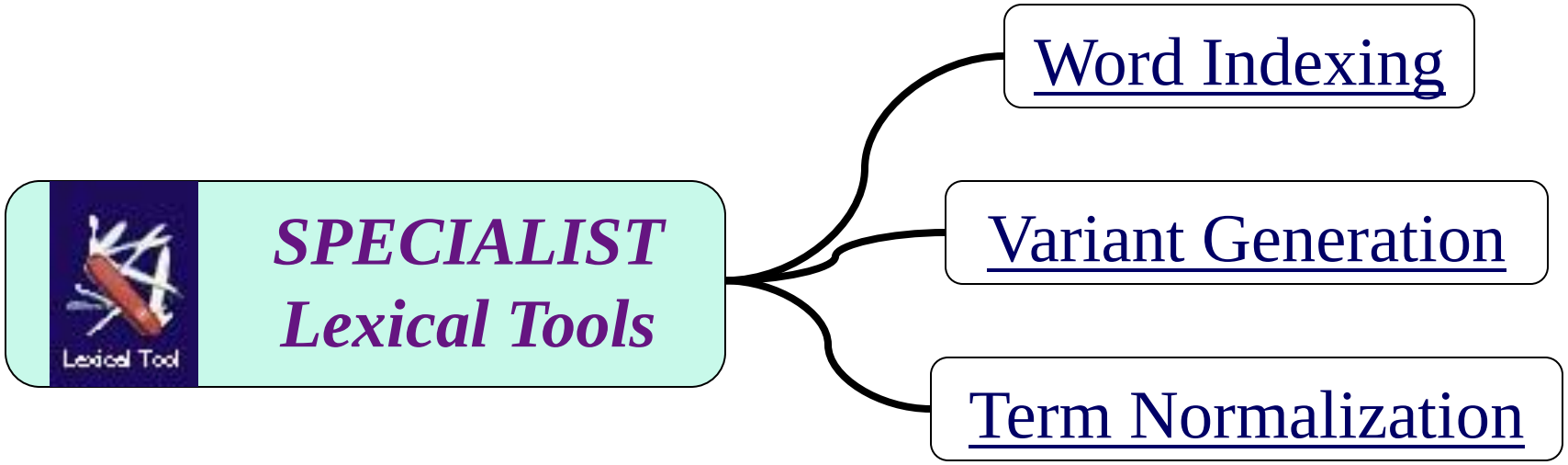
child Search Clear

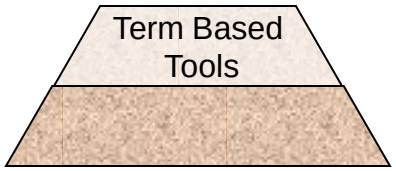
```
{base=child
entry=E0016427
        cat=noun
        variants=irreg|child|children|
}
{base=Child
entry=E0355216
        cat=noun
        variants=reg
        variants=uncount
        proper
}
```

Warning: This record's content has been modified to fit this screen.

Lexicon

SPECIALIST Lexical Tools





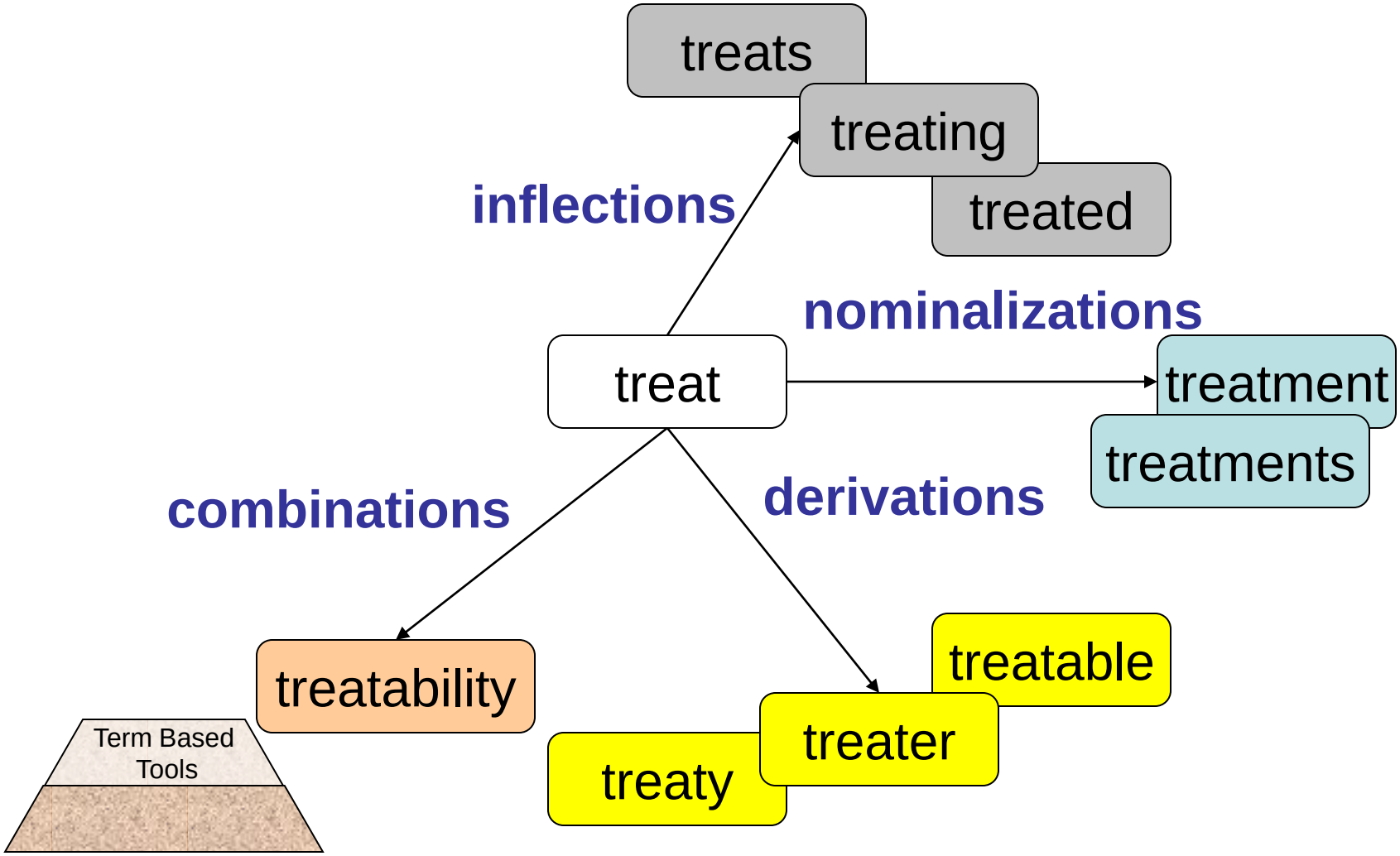
THE LISTER HILL NATIONAL CENTER
FOR BIOMEDICAL COMMUNICATIONS

A Research Division of the U.S. National Library of Medicine

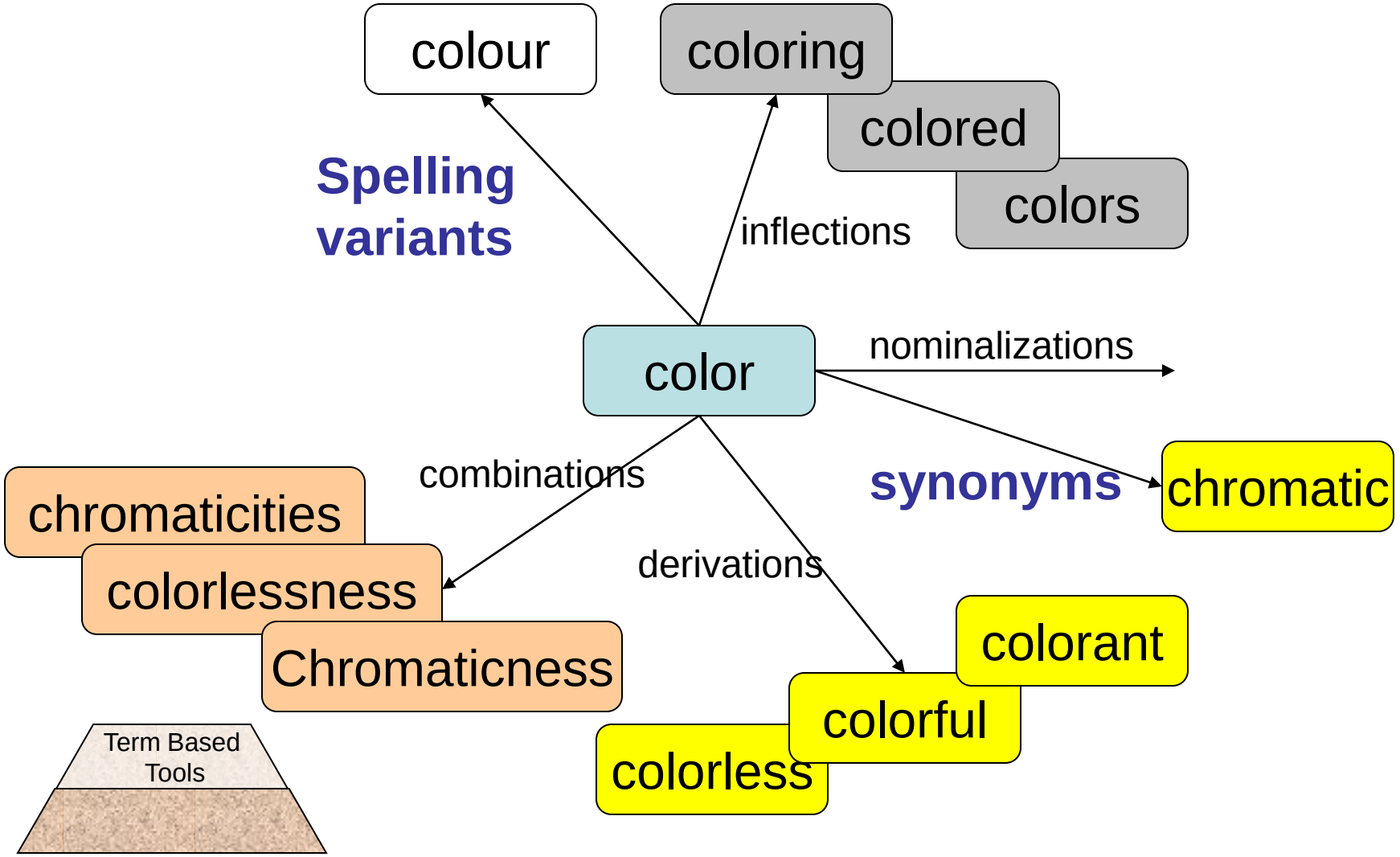
Uses

- Term Transformation
- Query Expansion
- Term normalization
- Building indexes
- Component of controlled vocabulary building tools
- Syntactic parsing
- Component of search engines
- Component of text-to-concept mapping tools
- Component of automated document indexing tools
- Component of text summarization tools
- Component of data-mining tools

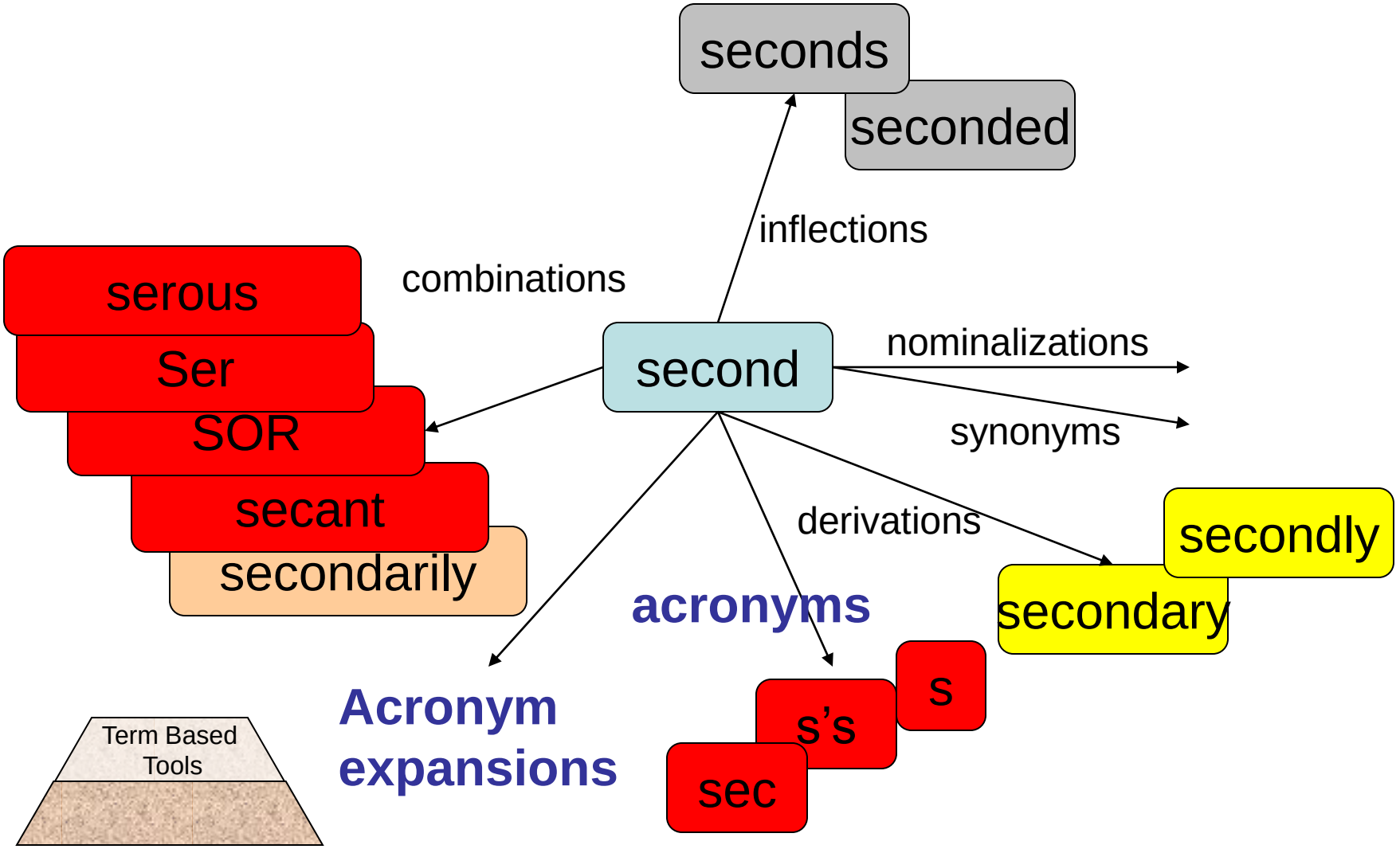
Lexical Variant Generation



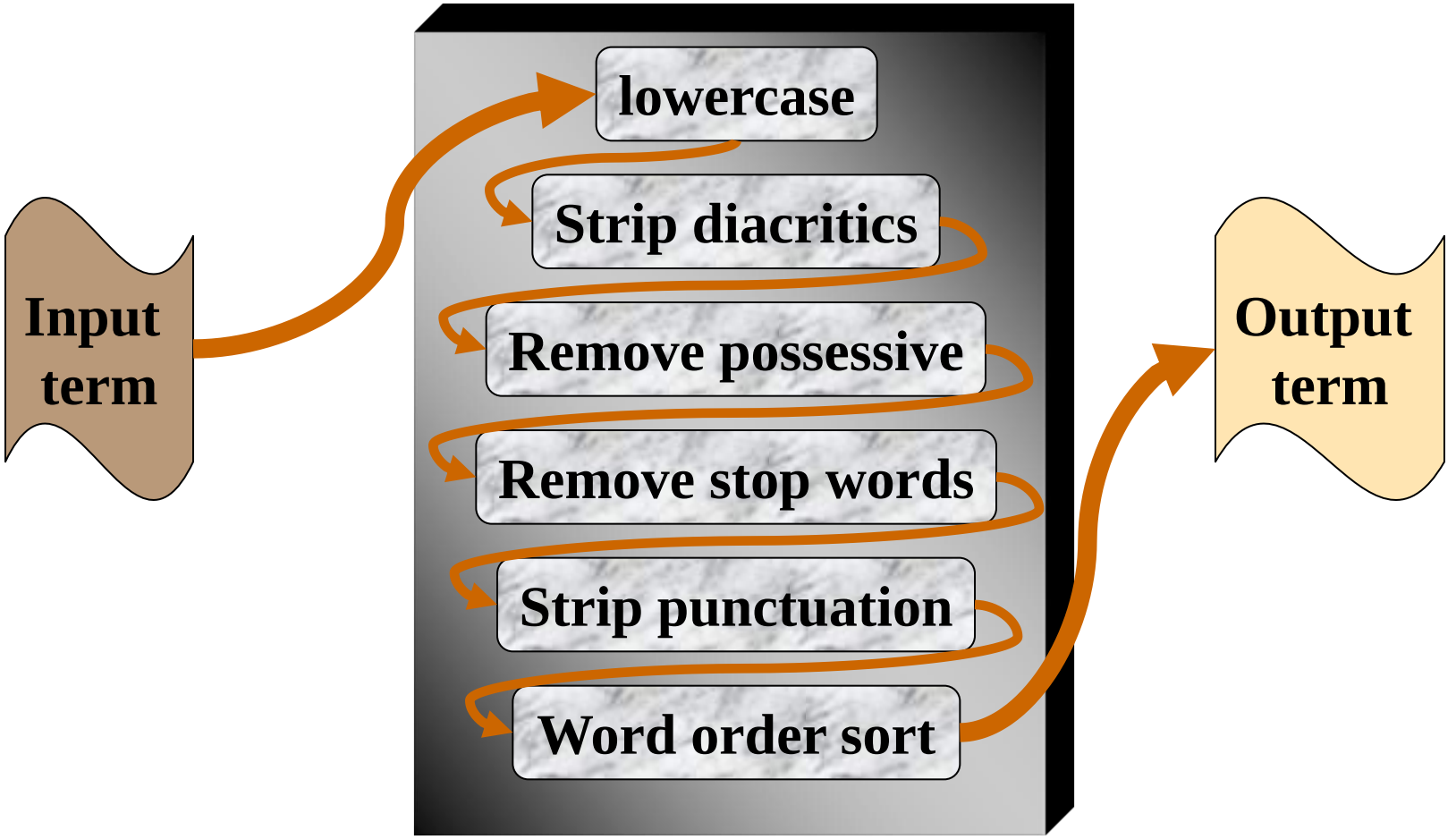
Lexical Variant Generation

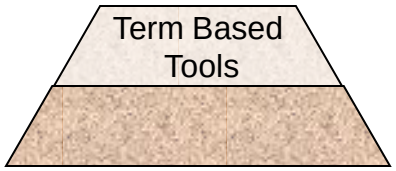


Lexical Variant Generation



Lexical Variant Generation



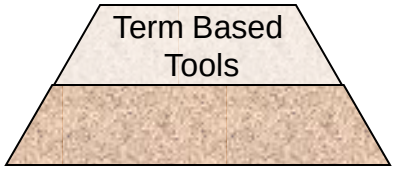


The tools can be arranged so that the output of one is the input to another.

Term Normalization

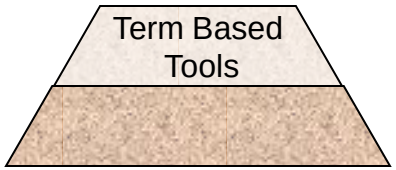
Norm abstracts away from:

- case
- punctuation
- word order
- stop words
- possessive forms
- inflectional variation
- spelling variation
- normalizes diacritics/ligatures/symbols



Term Normalization Examples

- Word order
  - Upper left lobe of lung
  - Left upper lobe of lung
- Possessive Forms
  - Grave's Disease
  - Graves Disease
  - Graves' Disease
- Diacritic/Ligature/Symbol Normalization
  - entrée, anæsthesia, β-blockers, Medline®



Term Normalization Example

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

disease hodgkin

Note: A normalized form is not necessarily itself a readable term. It is a hash.

Normalization web tool

Lgt, a GUI Example

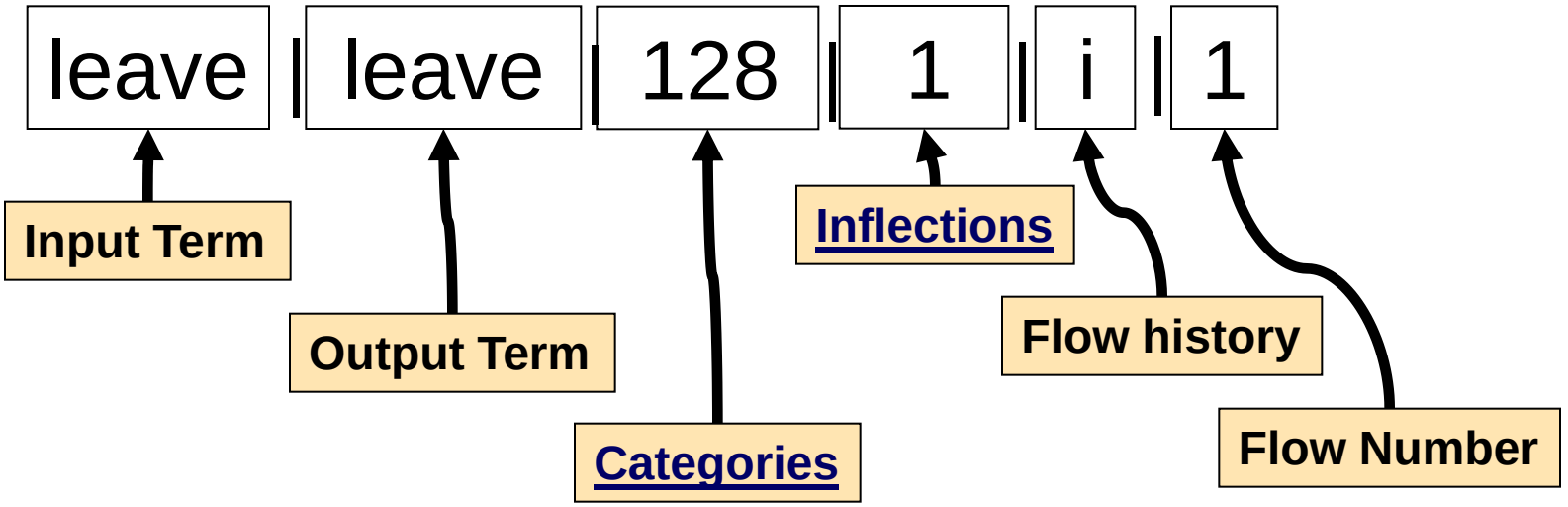
Command Line Example

```
> lvg –f:i –SC -SI
leave
leave|leave|<noun>|<base>|i|1|
leave|leave|<noun>|<singular>|i|
leave|leaves|<noun>|<plural>|i|1|
leave|left|<verb>|<past>|i|1|
leave|left|<verb>|<pastPart>|i|1|
leave|leave|<verb>|<base>|i|1|
leave|leave|<verb>|<pres1p23p>|
leave|leave|<verb>|<infinitive>|i|
leave|leaves|<verb>|<pres3s>|i|1
leave|leaving|<verb>|<presPart>|
```

```
> 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|
leave|leave|1024|1024|i|1
leave|leaves|1024|128|i|1
leave|leaving|1024|16|i|1|
```

Command Line Example: Output Fields Explained

> lvg –f:i
  leave



SPECIALIST Lexical Tools: API's

- **Outline of the needed pieces:**

  - import gov.nih.nlm.nls.lvg.Api.*;

  - NormApi api = new NormApi();

  - Vector<String> out = api.Mutate(inStr);

  - api.CleanUp();

Norm API Example

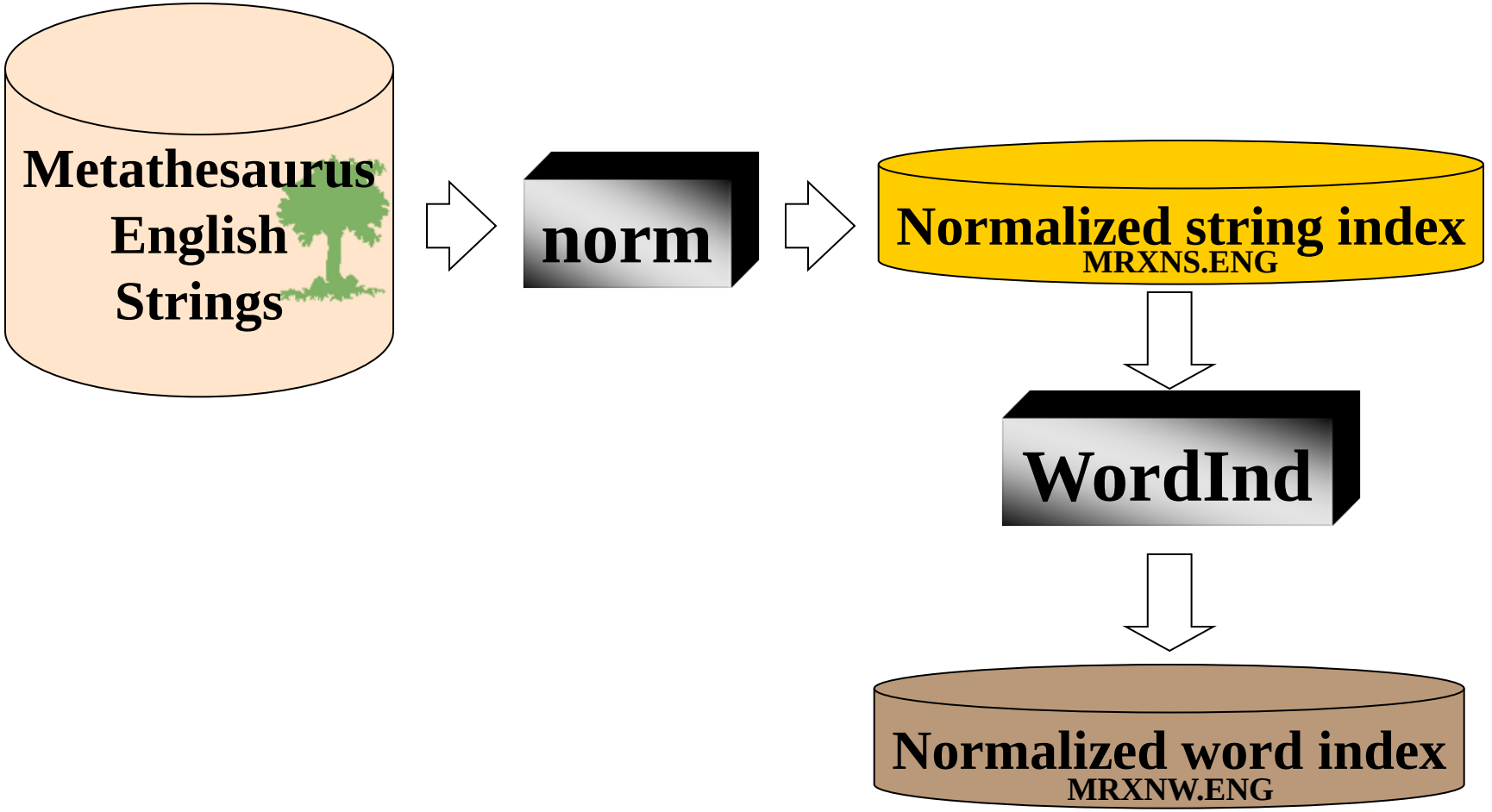
```
import java.util.*;
import gov.nih.nlm.nls.lvg.Api.*;

public class simplestApi
{
        public static void main(String[ ] args)
        {
                NormApi api = new NormApi( );    // instantiate a NormApi object

                try  // Process
                {
                        Vector<String> out = api.Mutate("inputs");
                        for(int i = 0; i < out.size( ); i++) // print out result
                        {
                                System.out.println(out.elementAt(i));
                        }
                }
                catch (Exception e) {  }
                api.CleanUp();       // make sure to clean up
        }
}
```

Norm API Example (2)

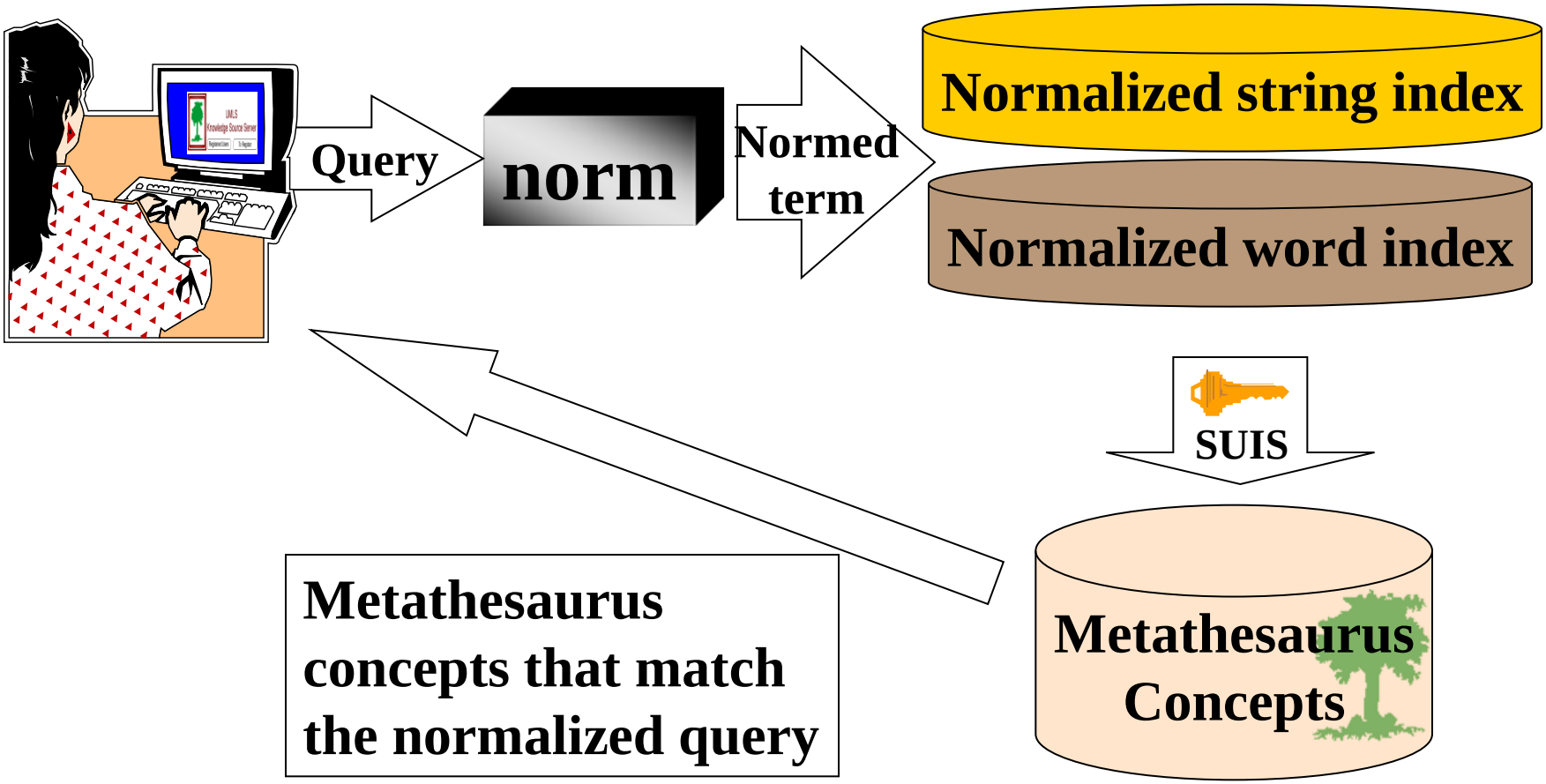
- **To compile and run:**

CLASSPATH = ${CLASSPATH}:
${LVG_DIR}:
${LVG_DIR}/lib/*lvg2007dist.jar:*

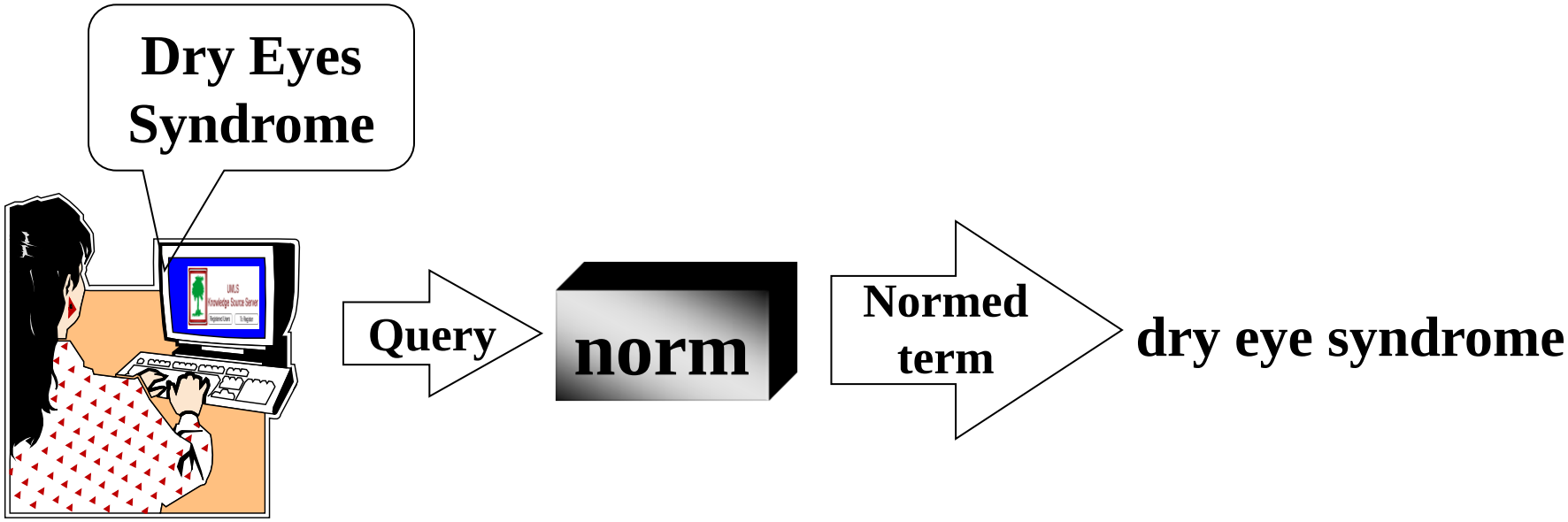
Application



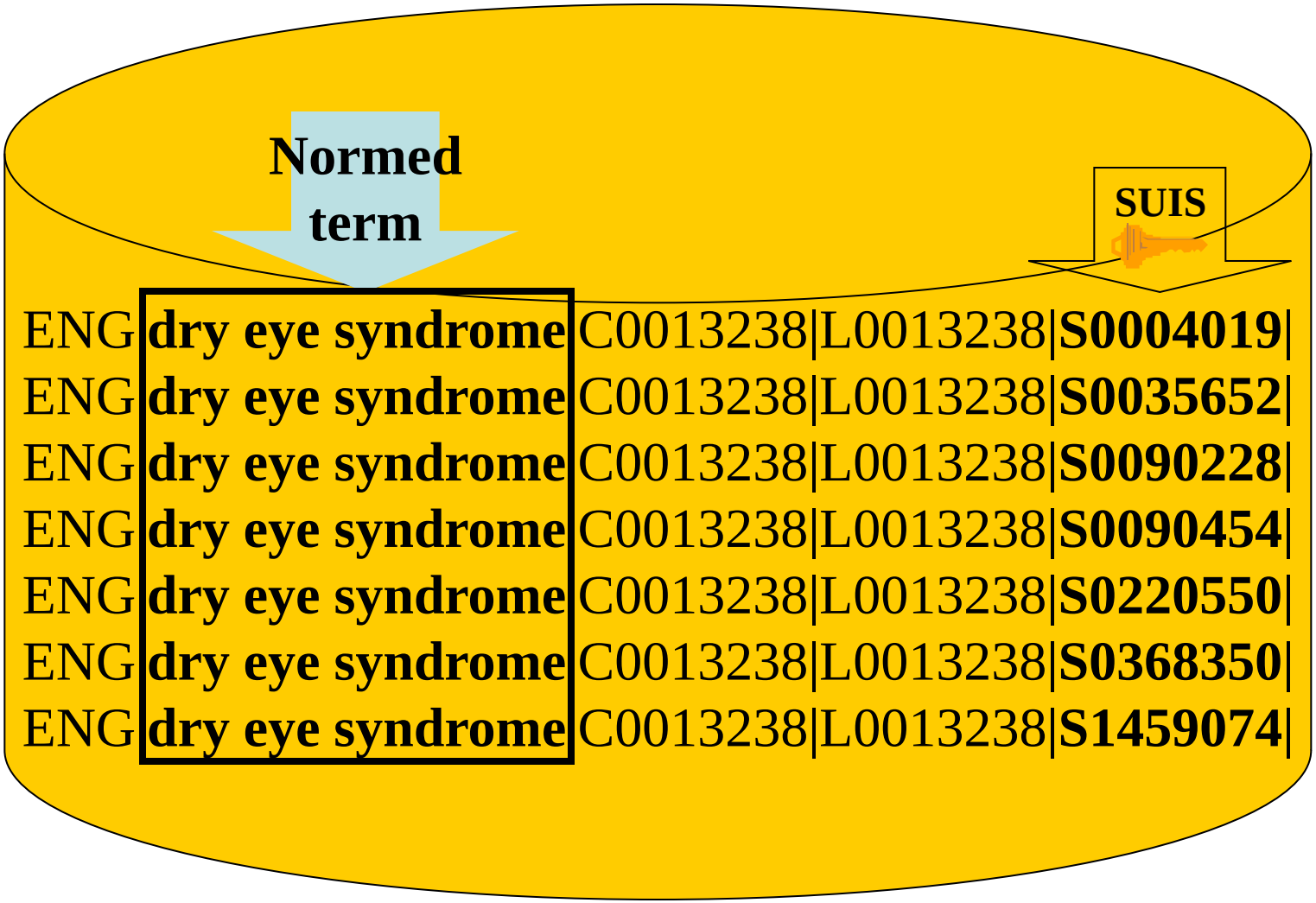
Application



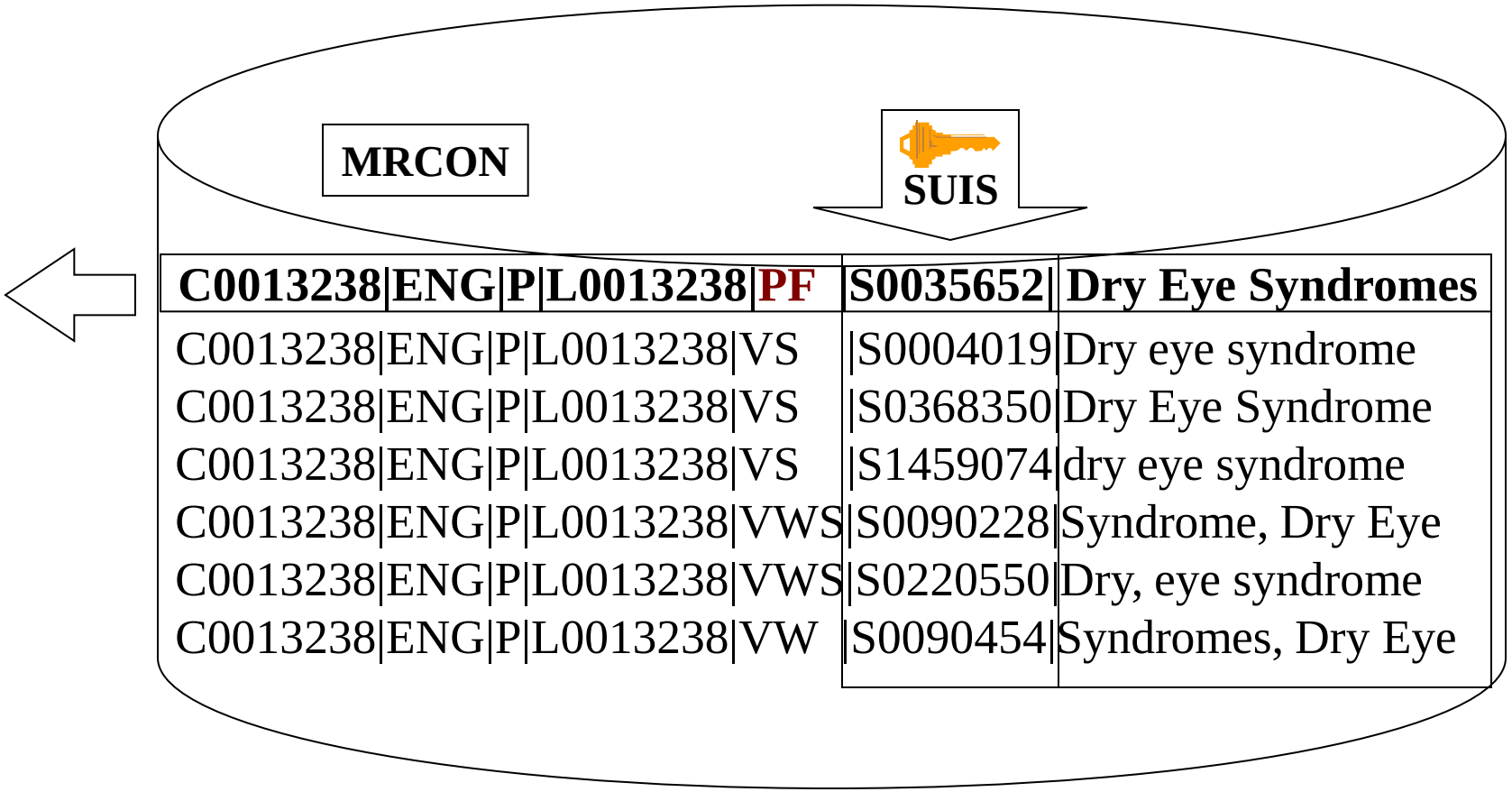
Example



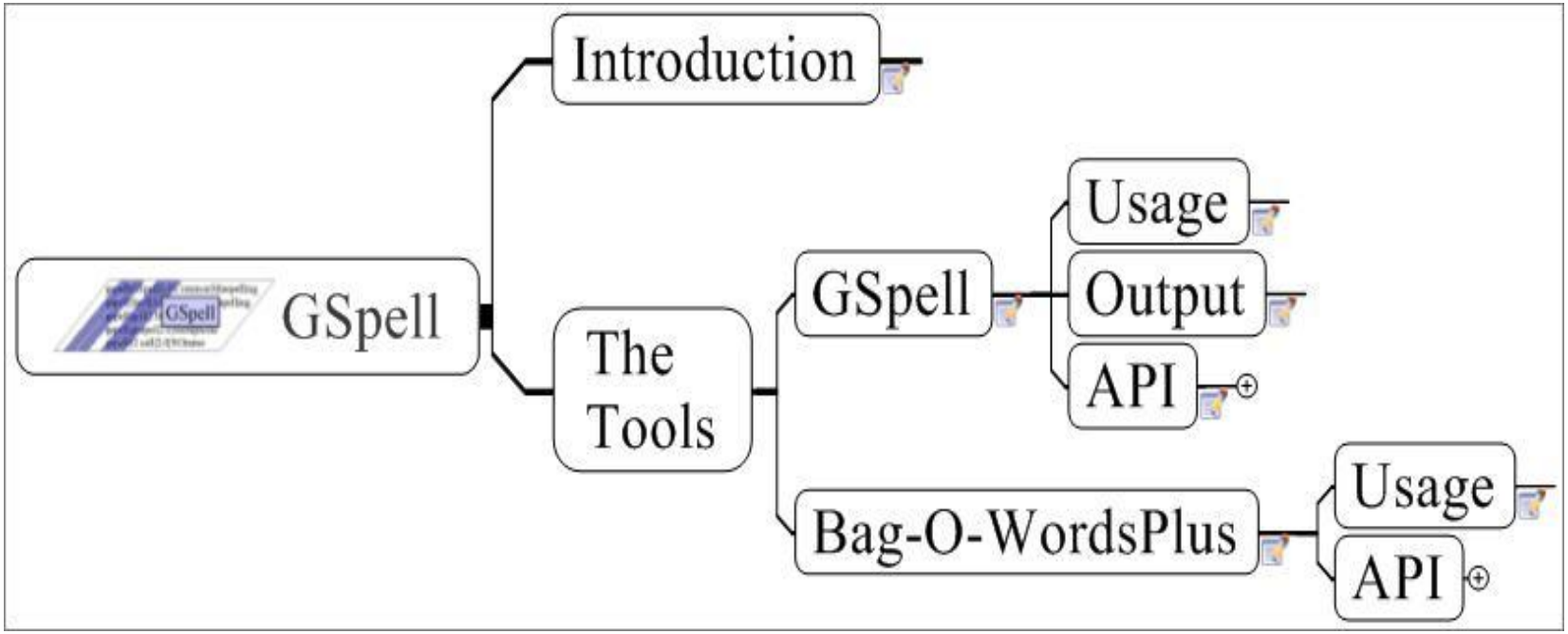
Example (Cont.)

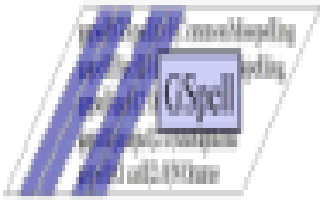


Example (Cont.)



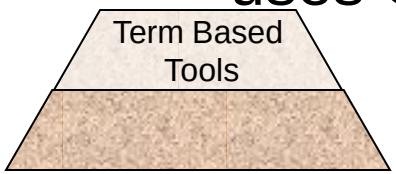
C0013238|ENG|P|L0013238|PF	S0035652|	Dry Eye Syndromes
C0013238|ENG|P|L0013238|VS	|S0004019|	Dry eye syndrome
C0013238|ENG|P|L0013238|VS	|S0368350|	Dry Eye Syndrome
C0013238|ENG|P|L0013238|VS	|S1459074|	dry eye syndrome
C0013238|ENG|P|L0013238|VWS	|S0090228|	Syndrome, Dry Eye
C0013238|ENG|P|L0013238|VWS	|S0220550|	Dry, eye syndrome
C0013238|ENG|P|L0013238|VW	|S0090454|	Syndromes, Dry Eye


GSpell
Introduction
The Tools
GSpell
Usage
Output
API
Bag-O-WordsPlus
Usage
API



Spelling Retrieval Tools

- **GSpell**
  - A term retrieval tool
  - N-gram nearest neighbor algorithm
  - MetaPhone phonetic spelling normalization
  - Homophones
  - Common misspellings
  - Candidates sorted by an edit distance and frequency of occurrence from a corpus
- **BagOWordsPlus**
  - a phrase retrieval tool
  - uses correctly spelled words within the phrase to limit possible candidates
  - uses GSpell only when it has to.



GSpell: Usage

Usage

GSpellFind.[*sh*|*bat*]

--dictionary=*NameOfDictionary*

[--**inputFile**=*Source*] [--**outputFile**=*target*]

[--**truncate**=*N*] [--**considerNCandidates**=*N*]

[--**maxEditDistance**=*N*]

GSpellIndex.[*sh*|*bat*]

--**dictionary**=*NameOfDictionary*

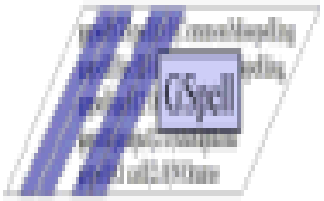
--**inputFile**=*SourceFile*

[--reportTime] [--version][--help]

GSpell: Output

Input Term	Suggestion	Edit Distance	Rank	Method	Message

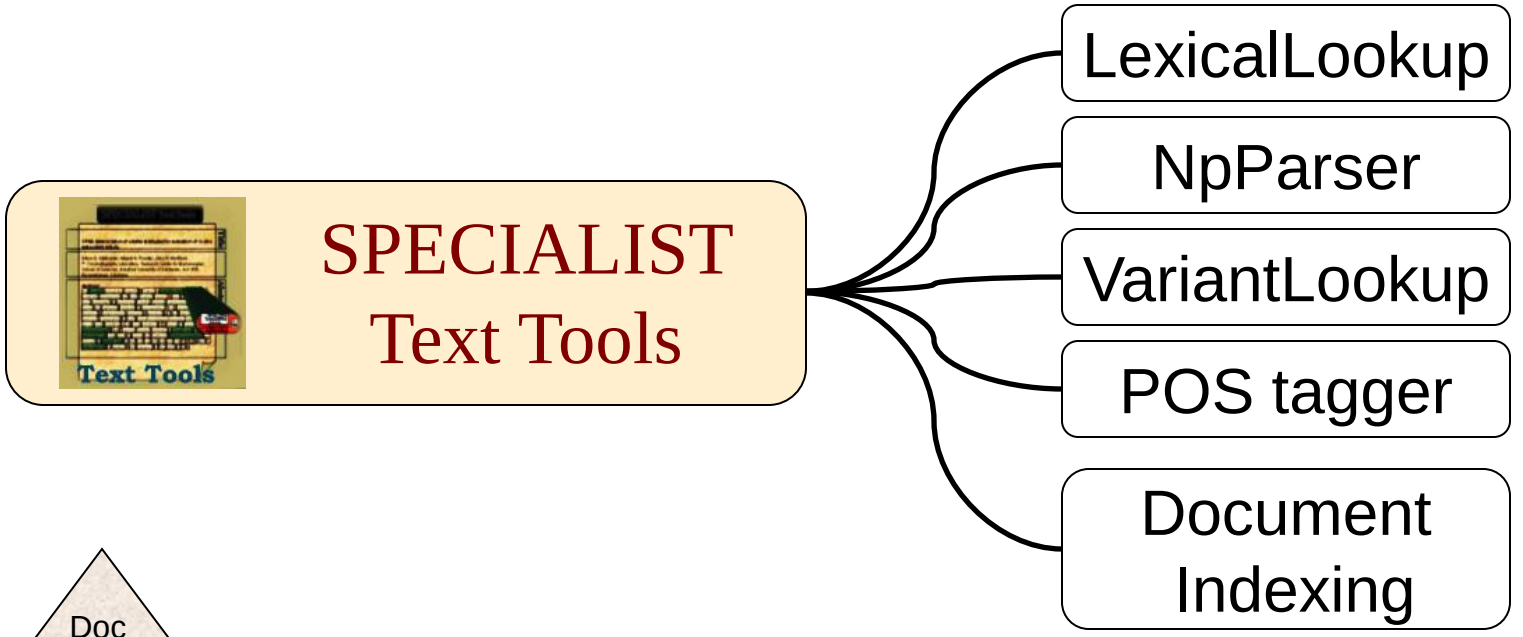
anonomous|**anonymous**|1.0|0.87|NGrams|
anonomous|**allonomous**|2.0|0.58|NGrams|
anonomous|**autonomous**|2.0|0.58|NGrams|
anonomous|**anadromous**|3.0|0.29|NGrams|
anonomous|**analogous**|3.0|0.29|NGrams|
anonomous|**anomalous**|3.0|0.29|NGrams|
anonomous|**anonymously**|3.0|0.29|NGrams|
anonomous|**anonymes**|3.0|0.29|Metaphone|
anonomous|**anonyms**|3.0|0.29|Metaphone|
anonomous|**acoprous**|4.0|0.11|NGrams|

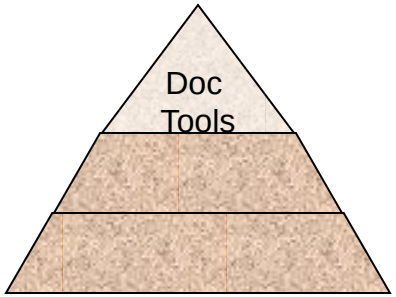


GSpell: API

```
import gov.nih.nlm.nls.gspell.GSpell;       // <-------These come from the gspell.jar
import gov.nih.nlm.nls.gspell.Candidate;

GSpell gspell = new GSpell( _dictionaryName,
                            GSpell.READ_ONLY );
Vector candidates = gspell.find( aTerm );
if ( candidates != null )
  for ( int i = 0; i < candidates.length; i++ )
     System.out.println(candidates[i].toString());
 else
    System.out.println("No Suggestions");

gspell.cleanup();
```

SPECIALIST Text Tools

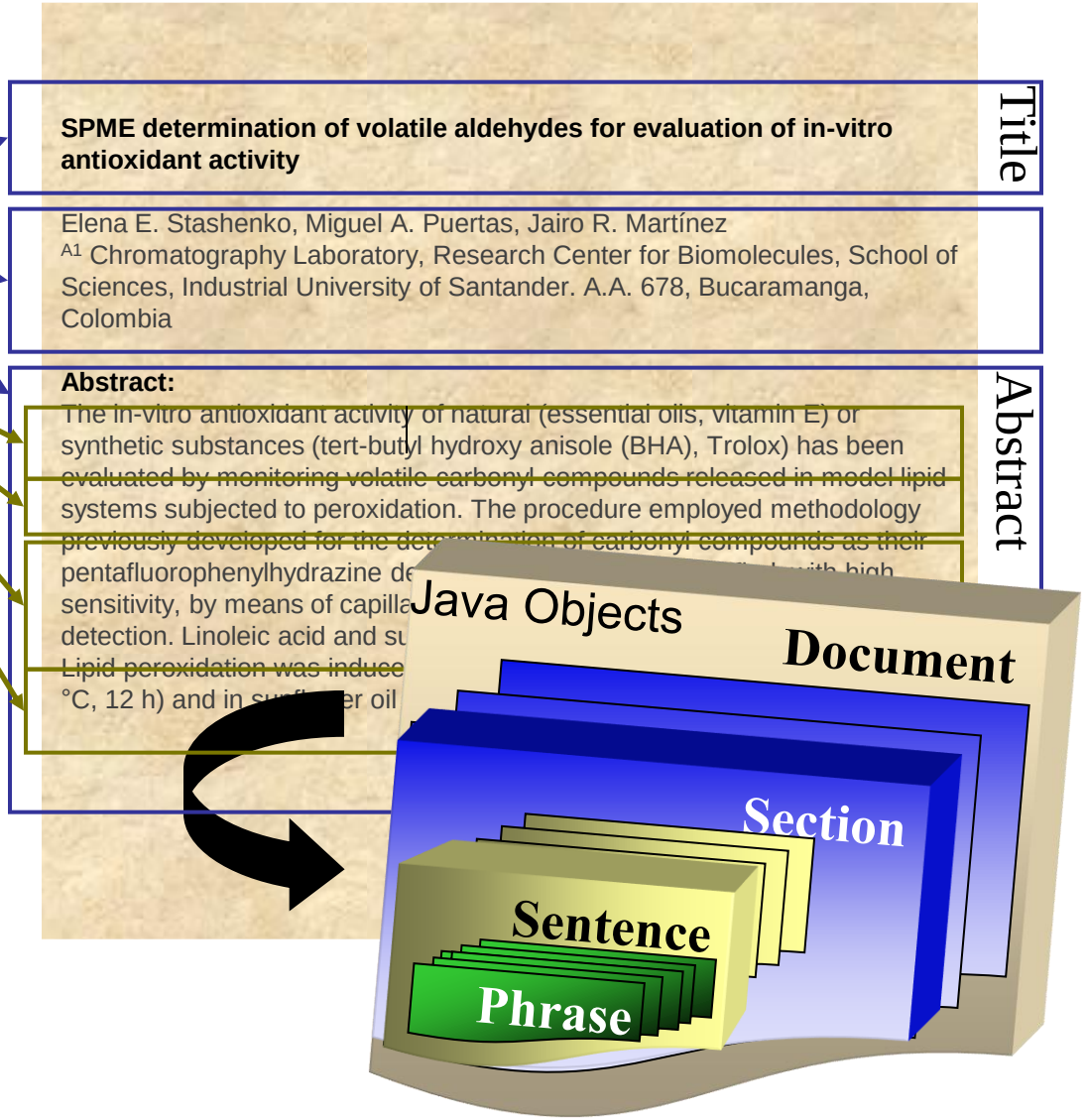




SPECIALIST TextTools

Segments text into:

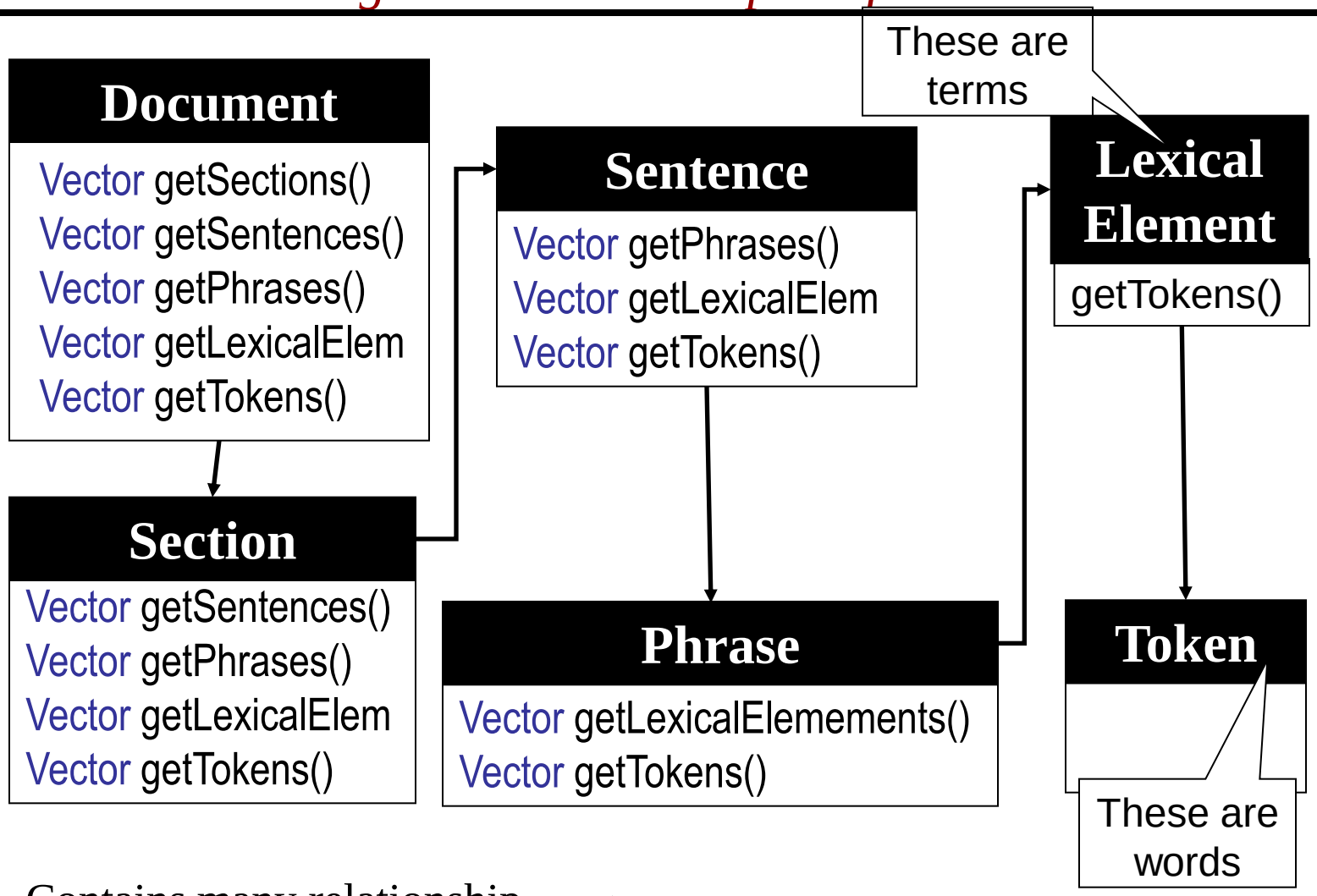
- Sections
- Sentences
- Phrases
- Terms
- Words
- Term variants



Specialist TextTools

Container Classes: Entity Diagram

gov.nih.nlm.nls.nlp.textfeatures



Specialist TextTools

Container Classes: Entity Diagram, more details

gov.nih.nlm.nls.nlp.textfeatures

Collection

Collection()
Collection(GlobalBehavior pSettings)
Collection(StringBuffer pText)
Collection(String pFileName,
GlobalBehavior pSettings)

Vector getDocuments()

Document (continued)

Doccument()
Document(File pFile)
Document(GlobalBehavior pSettings)
Document(String pFileName)

Variant

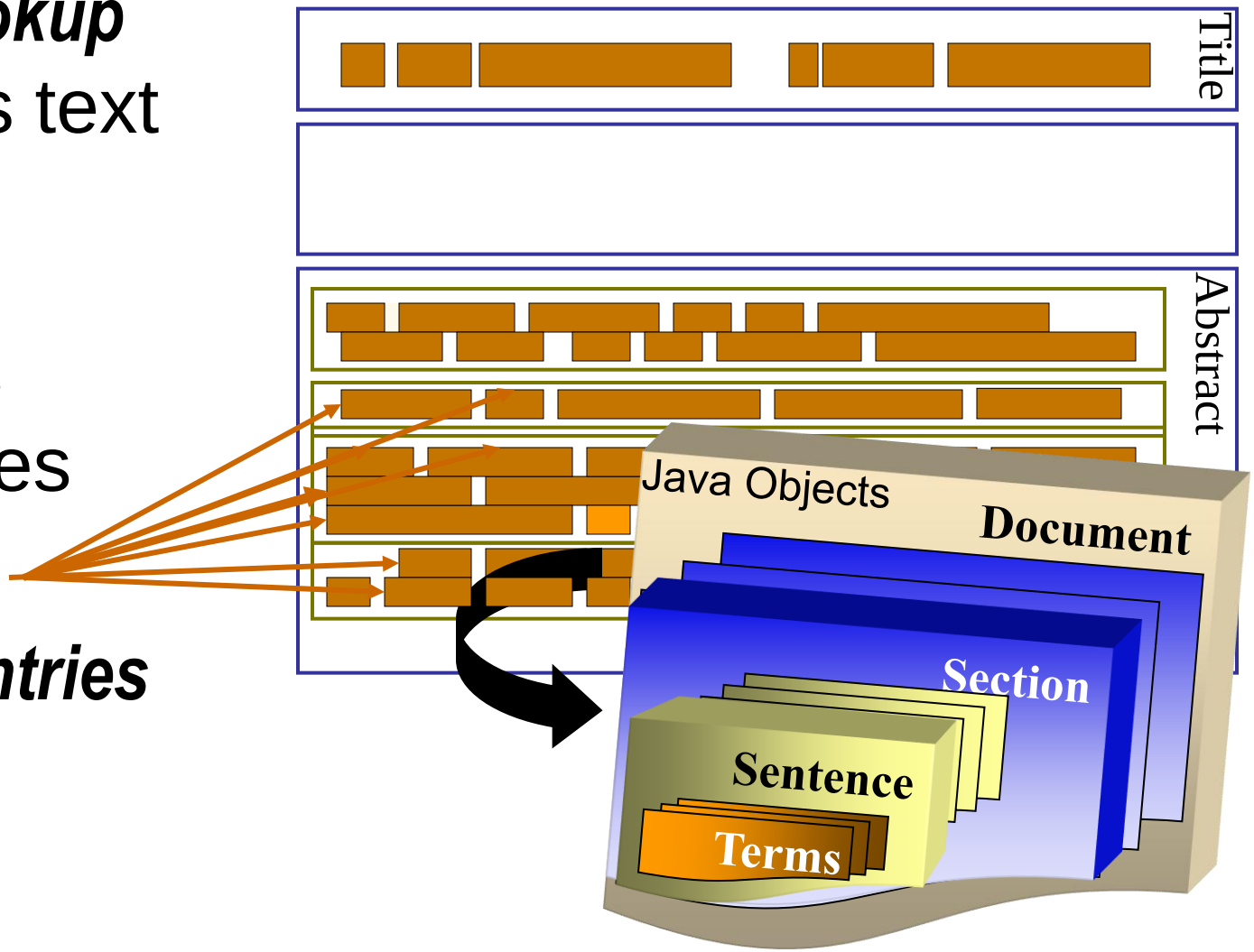
String getTerm()
Vector getTokens()
int getCategories()
int getDistance()
int getHistory()
String getNormalizedTerm()
int getOrigCat()
String getOrigTerm()
LexicalElement getParent()

SPECIALIST TextTools

LexicalLookup

LexicalLookup segments text into

- Sections
- Sentences
- ***Terms***
- ***Lexical Entries***
- Words

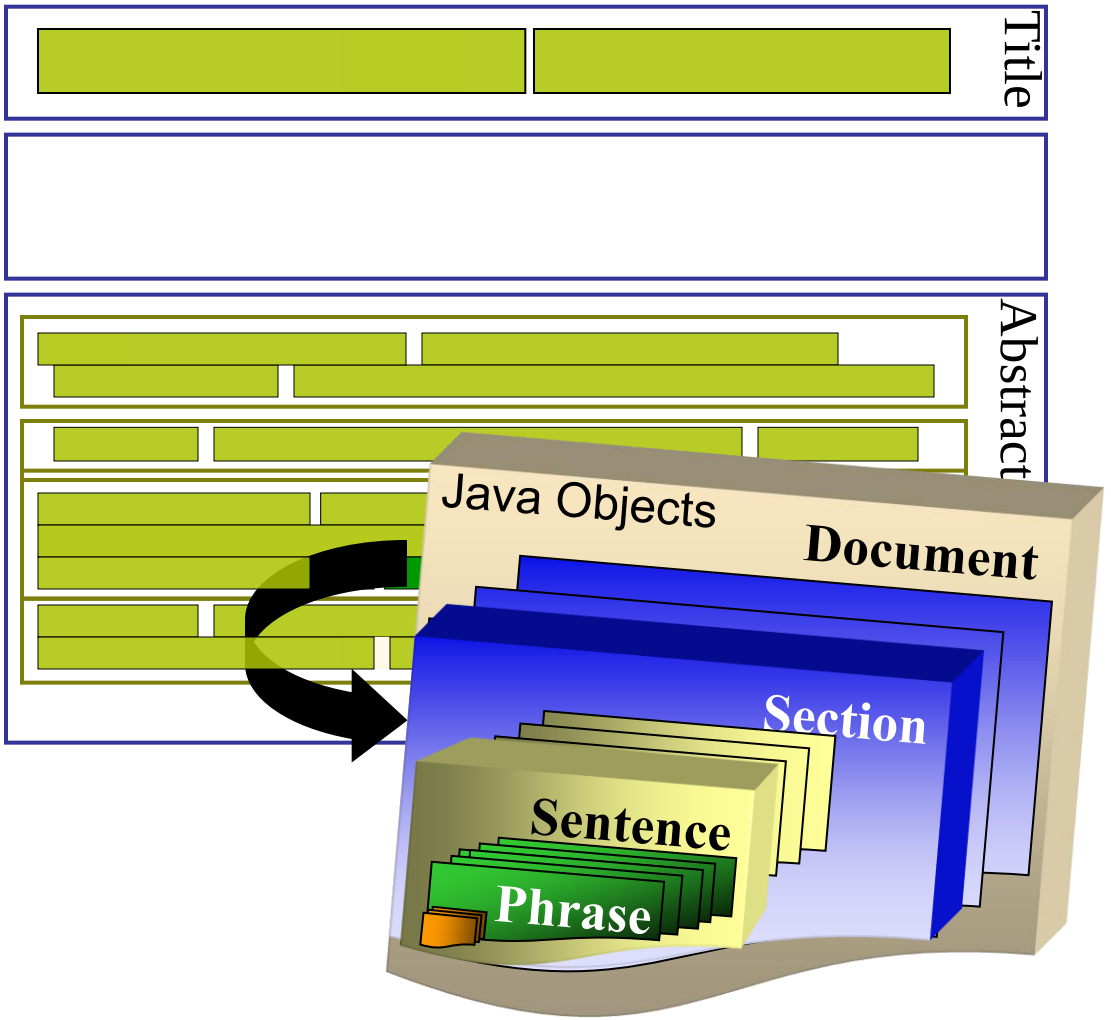


SPECIALIST TextTools

NPParser

Segments text into

- Sections
- Sentences
- ***Noun Phrases***
- Terms
- Words
- Lexical Entries



SPECIALIST TextTools

NPParser Example

NpParser --fileName=PMID14700470.txt

…

Section: 8|Title Words|

Sentence|*0*|*0*|*212*|*The knowledge and expectations of parents about the role of antibiotic treatment in upper respiratory tract infection - a survey among parents attending the primary physician with their sick child*.

Phrase|0|0|12|The knowledge|**knowledge**|2|true|NOUN PHRASE

Lexical Element|0|LEXICON|det|The|0|2|false

Lexical Element|1|LEXICON|noun|knowledge|4|12|true

Phrase|1|14|16|and|and|1|false|CONJUNCTION_PHRASE

Lexical Element|2|LEXICON|conj|and|14|16|false

Phrase|2|18|29|expectations|**expectation**

SPECIALIST TextTools

Classes

gov.nih.nlm.nlp.lexicallookup.**LexicalLookupAPI**

Constructor Summary
LexicalLookupAPI(gov.nih.nlm.nls.utils.GlobalBehavior pSettings)
LexicalLookupAPI(String[] args)

gov.nih.nlm.nlp.parser.Parse.**Parse**

Constructor Summary
Parse(gov.nih.nlm.nls.utils.GlobalBehavior pSettings)
Parse(String[] args)

SPECIALIST TextTools Common Methods

Method Summary	
void	processCollection(Collection pCollection)
void	processDocument(Document pDocument)
void	processSentence(Sentence pSentence)
Sentence	processSentence(String pString)

SPECIALIST TextTools

Special Sauce

gov.nih.nlm.nls.utils

Class GlobalBehavior: Constructor Summary

GlobalBehavior(String pName, String registryFile)

GlobalBehavior(String pName, String registryFile, String[] args)

Just the Name

Properties File Location:

/somepath/textTools_v0.X.X/nls/nlp/config/ NLPRegistry.cfg

Java Classpath requirement:

/somepath/textTools_v0.X.X/nls/nlp/config/

NLPRegistry.cfg: Example Contents:

-015|**--annotationFormat1**|boolean|false|Simple Annotation format

Command line arguments: Example Contents:

--annotationFormat1

SPECIALIST TextTools

Extract Terms from Documents

```
// ==============+ Create a LexicalLookupAPI object +==
LexicalLookupAPI look = new LexicalLookupAPI(args);

// ================================+ Chunk the file +==
Document aDocument = look.processDocument( aFile );

List terms = aDocument.getLexicalElements();
LexicalElement aTerm = null;
// ================+ Print the LexicalElements out +==
for (Iterator i = terms.iterator(); i.hasNext(); ) {
    aTerm = (LexicalElement) i.next();
    System.out.println(aTerm.toPipedString());
}
```

SPECIALIST TextTools

VariantLookup

- This is LVG's fruitful variants index available as an API within the textTools
- Is used to generate variants from noun phrases extracted from documents

gov.nih.nlm.nlp.lexicon.**VariantLookup**

Constructor Summary
VariantLookup(gov.nih.nlm.nls.utils.GlobalBehavior pSettings)

Method Summary
Variant[] find(String pTerm)
Variant[] find(String pTerm, int pCats, int pVarTableType)

SPECIALIST TextTools

taggerClient

- Assigns Parts of Speech (POS) to words in text
- NP parsers need terms with Parts of Speech assigned to determine phrase breaks and head assignment
- Includes LexicalLookup

SPME determination of volatile aldehydes for evaluation of in-vitro antioxidant activity

Elena E. Stashenko, Miguel A. Puertas, Jairo R. Martínez
[A1] Chromatography Laboratory, Research Center for Biomolecules, School of Sciences, Industrial University of Santander. A.A. 678, Bucaramanga, Colombia

Abstract:
The in-vitro antioxidant activity of natural (essential oils, vitamin E) or synthetic substances (tert-butyl hydroxy anisole (BHA), Trolox) has been evaluated by monitoring volatile carbonyl compounds released in model lipid systems subjected to peroxidation. The procedure employed methodology previously developed for the determination of carbonyl compounds as their pentafluorophenylhydrazine derivatives which were quantified, with high sensitivity, by means of capillary gas chromatography with electron-capture detection. Linoleic acid and sunflower oil were used as model lipid systems. Lipid peroxidation was induced in linoleic acid by the Fe^{2+} ion (1 mmol L^{-1}, 37 °C, 12 h) and in sunflower oil by heating in the presence of O_2 (220 °C, 2 h).

Legend: noun adj adv verb conj pron det prep shape

SPECIALIST TextTools TaggerClient

gov.nih.nlm.nlp.taggerservices.**TaggerClientMain**

Constructor Summary
TaggerClientMain(gov.nih.nlm.nls.utils.GlobalBehavior pSettings)
TaggerClientMain(String[] args)

SPECIALIST TextTools

taggerservices

gov.nih.nlm.nlp.taggerservices

Interface TaggerInterface

Method Summary
tag(Sentence pSentence)

Class gov.nih.nlm.nlp.taggerservices.**TaggerFactory**	
static TaggerInterface	build(GlobalBehavior pSettings)

NLPRegistry.cfg: Example Contents:

-043|**--tagger**|String|*medpostskr*|Name of tagger hooked in

MetaMap Transfer (MMTx)

- Extracts UMLS concepts from text
- Java Implementation of MetaMap

Retinoblastoma

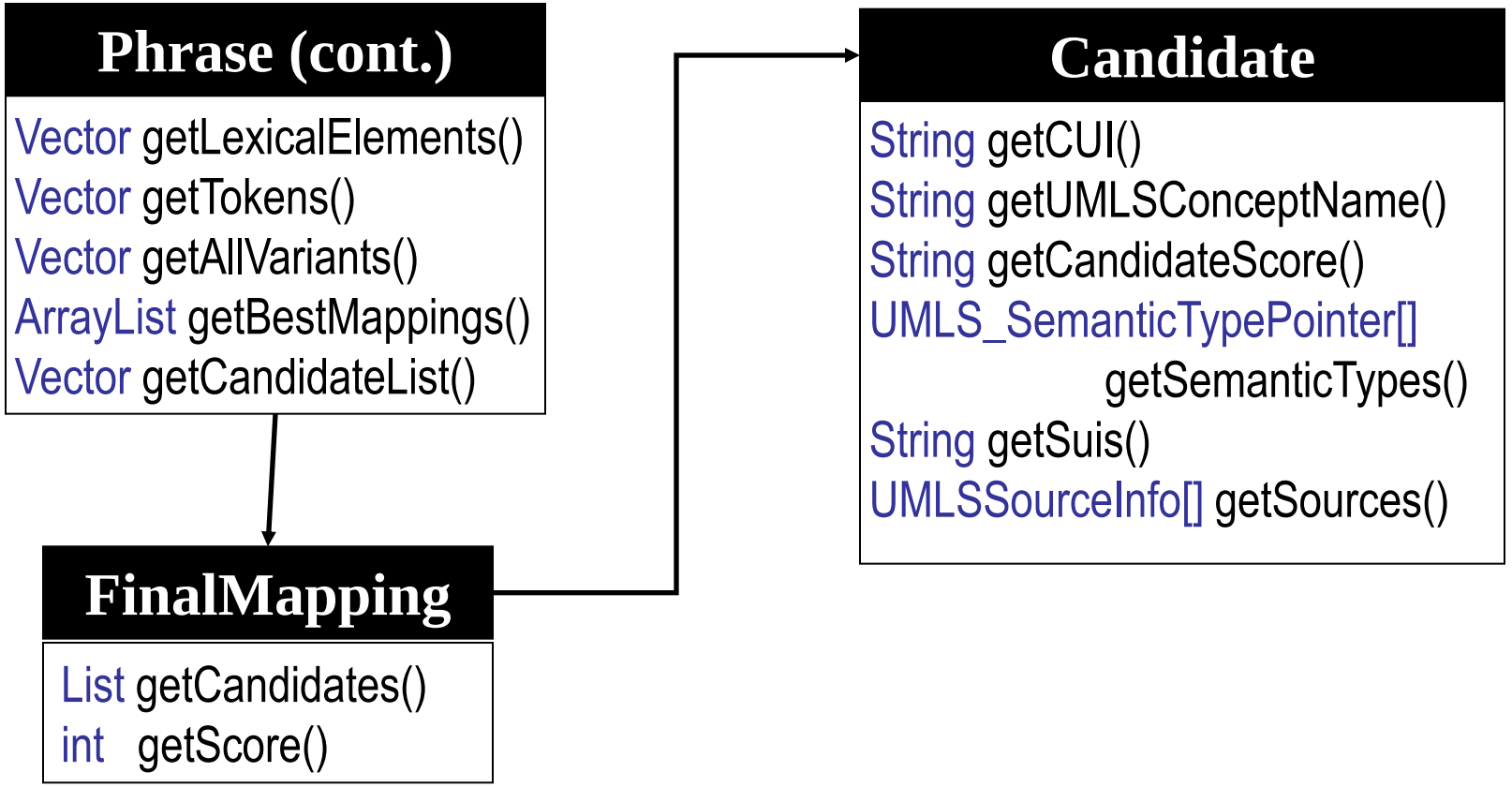
What is retinoblastoma?
Retinoblastoma is a rare type of eye cancer that develops in the retina, which is the part of the eye that detects light and color. Although this disorder can occur at any age, it usually develops in young children.

Meta Mapping (1000):
C0496836
(Malignant neoplasm of eye, unspecified)
[Neoplastic Process]

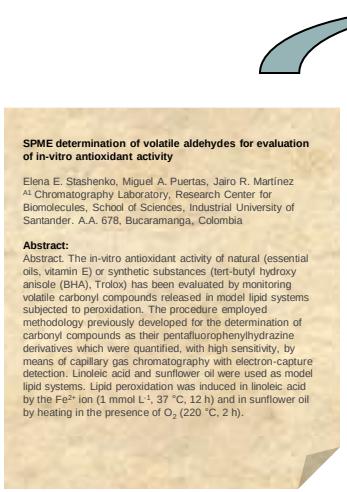
Specialist TextTools

Container Classes: Entity Diagram

gov.nih.nlm.nls.nlp.textfeatures



MetaMap Transfer (MMTx)



SPME determination of volatile aldehydes for evaluation of in-vitro antioxidant activity

Elena E. Stashenko, Miguel A. Puertas, Jairo R. Martínez
[a] Chromatography Laboratory, Research Center for Biomolecules, School of Sciences, Industrial University of Santander. A.A. 678, Bucaramanga, Colombia

Abstract:
Abstract. The in-vitro antioxidant activity of natural (essential oils, vitamin E) or synthetic substances (tert-butyl hydroxy anisole (BHA), Trolox) has been evaluated by monitoring volatile carbonyl compounds released in model lipid systems subjected to peroxidation. The procedure employed methodology previously developed for the determination of carbonyl compounds as their pentafluorophenylhydrazine derivatives which were quantified, with high sensitivity, by means of capillary gas chromatography with electron-capture detection. Linoleic acid and sunflower oil were used as model lipid systems. Lipid peroxidation was induced in linoleic acid by the Fe^{2+} ion (1 mmol L^{-1}, 37 °C, 12 h) and in sunflower oil by heating in the presence of O_2 (220 °C, 2 h).

Tokenization

Section, Sentence, Phrase, Term, Word

↓

Variant Generation

↓

UMLS Concept Retrieval

↓

UMLS Concept to Phrase Evaluation

↓

Final Mapping of Good Candidates to best cover the phrase

MetaMap Transfer (MMTx) Display Mappings

```
// ======================+ Display Phrase and Concepts +==
String displayPhrase( Phrase aPhrase ) throws Exception {

// ================================+ Get the Mappings +==
List finalMappings = aPhrase.getFinalMappings();

if ( finalMappings != null ) {
  Iterator mappingIterator = finalMappings.iterator();
  // ====================+ Iterate through the Mappings +==
  while (mappingIterator.hasNext()) {
   FinalMapping aMapping=(FinalMapping) mappingIterator.next();
     System.out.println( aMapping );
  }
 }
}
```

MetaMap Transfer (MMTx) MMTxAPI

```
// ============================+ Create a MMTxAPI object +==
MMTxAPI mmtx = new MMTxAPI(  );

// ==============================+ Analyze the Sentence +==
Sentence aSentence = mmtx.processSentence("Insomnia is a symptom of a
    sleep disorder" );

Iterator phraseIterator=aSentence.getPhrases().iterator();
// ======================+ Iterate through the Phrases +==
while ( phraseIterator.hasNext() ) {
  Phrase aPhrase = (Phrase) phraseIterator.next();

  System.out.println( displayPhrase( aPhrase) );

}
```

MetaMap Transfer (MMTx) MMTxAPI

Phrase: "non-hodgkin's lymphoma"

Meta Candidates (5)

1000 ***Lymphoma, Non Hodgkin's*** [Neoplastic Process]

861 hodgkin's lymphoma (***HODGKINS DISEASE***) [Neoplastic Process]

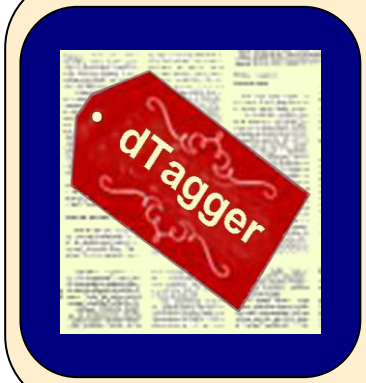
812 Lymphoma (***Germinoblastoma***) [Neoplastic Process]

812 ***Lymphoma*** [Neoplastic Process]

805 NON (***NON Mouse***) [Mammal]

Meta Mapping (1000)

1000 ***Lymphoma, Non Hodgkin's*** [Neoplastic Process]



The SPECIALIST dTagger (A POS Tagger)

specialist.nlm.nih.gov

- Introduction
- Features
- Lexical Lookup
- Handling Unknowns
- Training
- Tagging
- Updating
- Error Analysis
- Work Still to Do

THE LISTER HILL NATIONAL CENTER FOR BIOMEDICAL COMMUNICATIONS

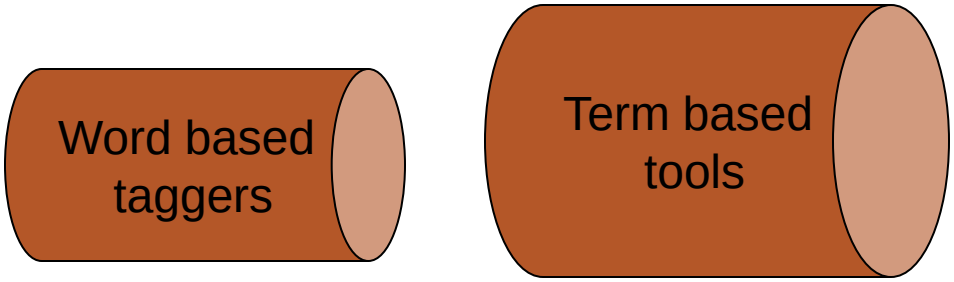
A Research Division of the U.S. National Library of Medicine

NATIONAL LIBRARY OF MEDICINE

Motivation

Why another POS tagger?

- SPECIALIST Lexicon
- Arbitrary tag set
- Supervised and unsupervised training and updating
- True multi-word (term based) tagger
- Generalizable to other languages



Features

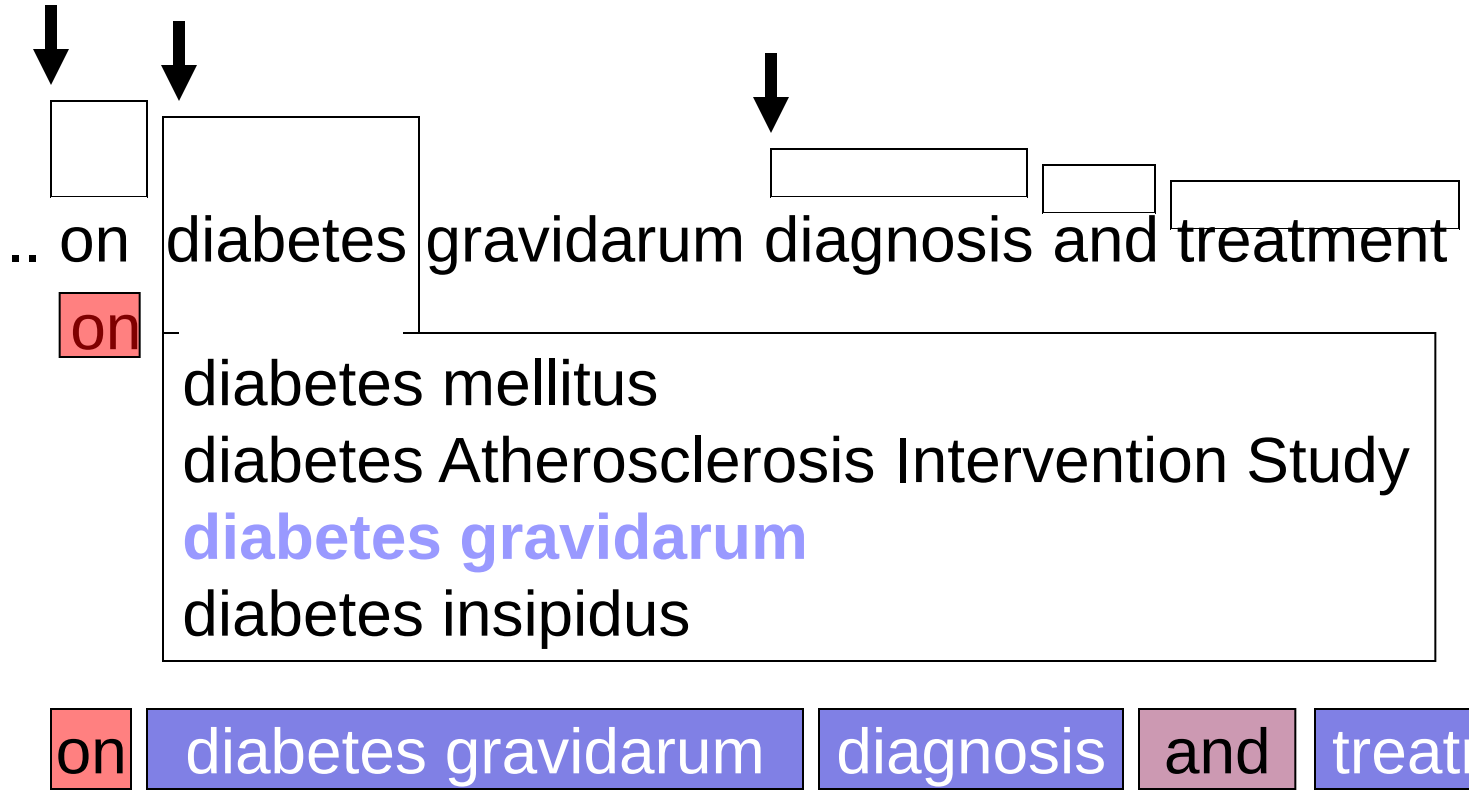
- Tag set specified as a configurable file
  - Just make sure Lexicon/tagset/corpus use the same tags.
- Lexical Information (from .lex files)
  - SPECIAST lexicon
  - Pseudo lexicon (number words, roman numerals)
  - Verbs as adjectives
  - Local lexicon

Features (2)

- Lexical lookup, pattern recognition to tokeninize into terms
- Unknown words handled with several strategies
- Hidden Markov Model, Viterbi Model used for training and tagging
- Probability of correct tag assignment reported back

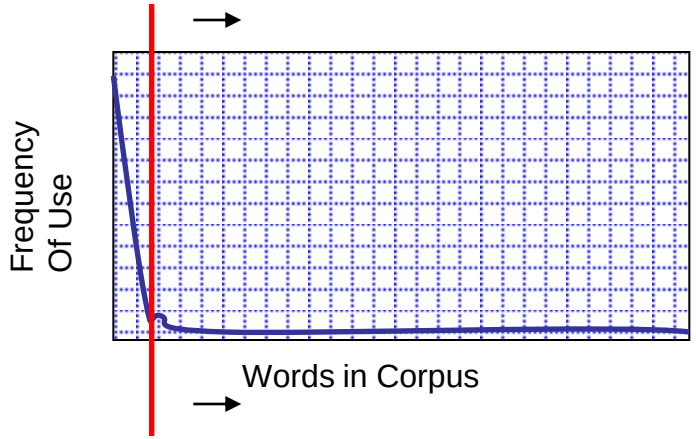
Lexical Lookup

Longest spanning matches from Lexicon



Handling Unknowns

- Overall probability of the next word being an unknown word gathered for (open class) words with low frequency



- Suffix statistics gathered from annotated corpus or from Lexicon (for unsupervised training) for open class terms.

Handling Unknowns: Shape Identification

```
Fifty two|trials|were|identified|which|fulfilled|
wordnum  |noun  |aux |verb      |pron |verb     |

 In  |17 |trials|with|placebo|groups|
 prep|num|noun  |prep|noun   |noun  |

... cure|rate|of  |placebo|preparations|was|30%     |
... verb|noun|prep|noun   |noun        |aux|percent|

... of  |10 weeks      |( |range|4  |to  |24 weeks      |) |
... prep|unitOfMeasure|lp|noun |num|prep|unitOfMeasure|rp|

... myocardial|scintigraphy|(P=0.02)
... adj       |noun        |levelOfSignificance
```

Handling Unknowns: Shape Identification (2)

- Units of Measure
- Level of Significance
- Experiment Size
- Number
- Real Number
- Word Number
- Telephone Number
- Range
- EmailAddress
- Date
- URL
- Fraction
- Percent
- Glob
- Equation
- Address
- Chemical
- Gene Name
- Proper Name
- Name
- Delimiter

*Next or future release

Unsupervised Updating and Training

- Unsupervised Updating

  With a prior model and lexicon and an un-annotated corpus

  Even 10 hand annotated sentences as the initial model gives a big boost

- Unsupervised Training

  With only the lexicon and no prior model

  Suffix statistics gleaned from lexicon

dTagger
Available Programs

- TrainWithTaggedText
- Tag
- UpdateWithUntaggedText
- TrainWithUntaggedText
- MorphologyDiscovery
- AnalizeTaggedCorpus (next release)

Unsupervised Updating and Training

Option	Description	Default Value
--dirName=	Where the input files will be found	the current directory
--fileName=	The input file or input file pattern	standard input
--modelName=	Saved Hidden Markov model name	“default”
--Rev=	Revision number useful for tracking multiple updates to a model	“0”

Unsupervised Updating and Training

Option	Description	Default Value
--corpusName=	This is echoed out in output stats	“MedPost”
--sentenceIDMarker	Sentence ID in the training corpus.	“P”
--overwrite	Overwrite the current rev’s probs when updating	false

Error Analysis

- Verbs tagged as nouns
- Adj/noun and noun/adj’s
- Odd usages
- Human tagging inconsistencies
- Conflicts with what the lexicon says

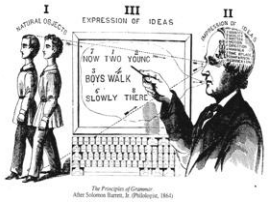
The dTagger Tag Class

gov.nih.nlm.nls.dtagger.**Tag** Constructor Summary
Tag(GlobalBehavior pSettings)

Method Summary	
void	processCollection(Collection pCollection)
void	processDocument(Document pDocument)
void	processSentence(Sentence pSentence)
double	tag(LexicalElement[] pTerms)
double	tag(Sentence pSentence)
LexicalElement[]	tag(String pSentence)

Work In Progress

- Put the latest and greatest on the website (includes textTools w/ dTagger now)
- More evaluation
  - Single vs. multi-word comparison
  - Using different tag sets
- Applied to Spanish to help build Spanish lexical resources
- “fix it, Make it better, make it faster” tasks



Lexical Systems Group

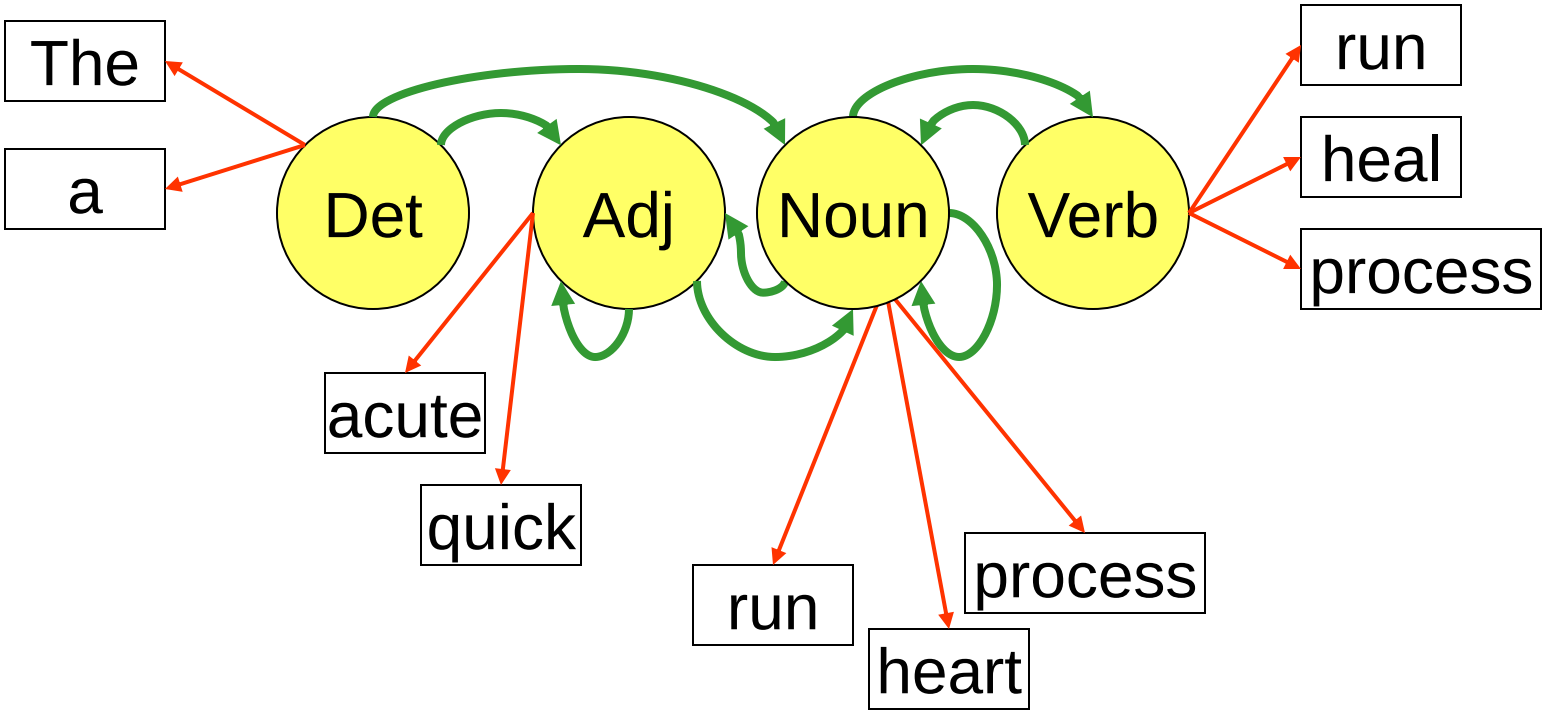
Websites	http://specialist.nlm.nih.gov http://mmtx.nlm.nih.gov
Contacts	Allen Browne browne@nlm.nih.gov Guy Divita divita@nlm.nih.gov Chris Lu lu@nlm.nih.gov Susanne M. Humphrey humphrey@nlm.nih.gov Lexical Systems Group umlslex@nlm.nih.gov

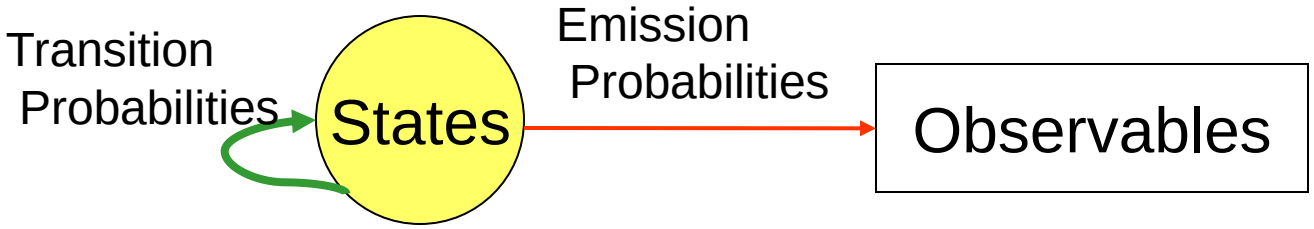
THE LISTER HILL NATIONAL CENTER FOR BIOMEDICAL COMMUNICATIONS

A Research Division of the U.S. National Library of Medicine

This Space to be filled in in Brisbane

Hidden Markov Model for POS Tagging





Training (1)

Hidden Markov Model

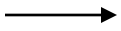
Emission counts gathered from annotated corpus

	Noun	Adj	Adv	Verb	Det	Conj	…
the	0	0	0	0	7406	0	
cause	7	0	0	13	0	0	
...							

Training (2)

Transition counts gathered from annotated corpus

	Noun	Adj	Adv	Verb	Det	Conj	…
Noun	14242	1235	713	3231	3100	424	
Adj	11421	1703	113	11	4	1	
Adv	140	631	285	2014	359	9	
Verb	1273	758	794	97	19	3	
...							



Tagging

- Viterbi Algorithm:

State Probabilities

	Do	viruses	cause	cancer
Noun		97 %	45 %	97 %
Adj				
Adv				
Verb	77 %		52 %	
Aux	23 %			
Unknown		.07 %	.07 %	.07%
…				

```
      |Do   |viruses|cause|cancer|
34.92|verb|noun    |verb |noun  |
```