Speech, NLP and the Web

Pushpak Bhattacharyya
CSE Dept.,
IIT Bombay

Lecture 13, 14, 15: Morphology: English verb group

(lecture 11 was on Classifiers for sentiment analysis by Sagar; lecture hour 12 was for quiz-1)

NLP Architecture

Increased Complexity Of Processing

Discourse and Coreference

Semantics Extraction

Parsing

Chunking

POS tagging

Morphology

Morph Analyser, Lemmatiser, Stemmer

- Morph Analyzer: valid root + features
- Lemmatizer: valid root; no features
- Stemmer: valid root not necessary

Example: Ladies

Morph Analyzer output: lady + ies (+plural)

Lemmatizer: lady

Stemmer: lad/ladi

Various word formation phenomena

- Inflection: *boy→boys*
- Derivation: *boy→boyish (noun→adjective)*
- Foreign word borrowing: *ombrella (italian)→umbrella (English)*
- Acronyms: *UN, WHO*
- Clipping: *Professor→Prof*
- Blending: *Breakfast+Lunch→Brunch*
- Compounding: Air+bus→Airbus

What governs noun's forms

- Mainly: Number, Direct/Obliqueness, Honorific
  - *Number: लड़का (ladakaa)→ लड़के (ladake)*
  - *D/O: ladakoM ne, ladakoM ko, laadakoM se*
    - *Presence of case*
  - *Honorific: (Japanese) Uchida →Uchida_san*

What governs verb's forms

- **GNPTAM**: Gender, Number, Person, Tense, Aspect, Modality
  - G: jaauMgaa (M), jaauMgii (F)
  - N: jaauMgaa (sg), jaaeMge (pl)
  - P: jaauMgaa (1^{st}), jaaoge (2^{nd}), jaaegaa (3^{rd})
  - T: jaauMgaa (fut), jaataa huM (pre)
  - A: jaauMgaa (normal), jaataa rahuMgaa (continuous)
  - M: jaauMgaa (normal), jaa sakuMgaa (ability)

Morphological complexity: Finnish

- *istahtaisinkohan* "I wonder if I should sit down for a while"
- *ist* + "sit", verb stem
- *ahta* + verb derivation morpheme, "to do something for a while"
- *isi* + conditional affix
- *n* + 1st person singular suffix
- *ko* + question particle
- *han* a particle for things like reminder (with declaratives) or "softening" (with questions and imperatives)

Morphological complexity: Telugu

- **Telugu:**

ame <u>padutunnappudoo</u> nenoo panichesanoo

she singing I work

I worked while she was singing.

Morphological complexity: Turkish

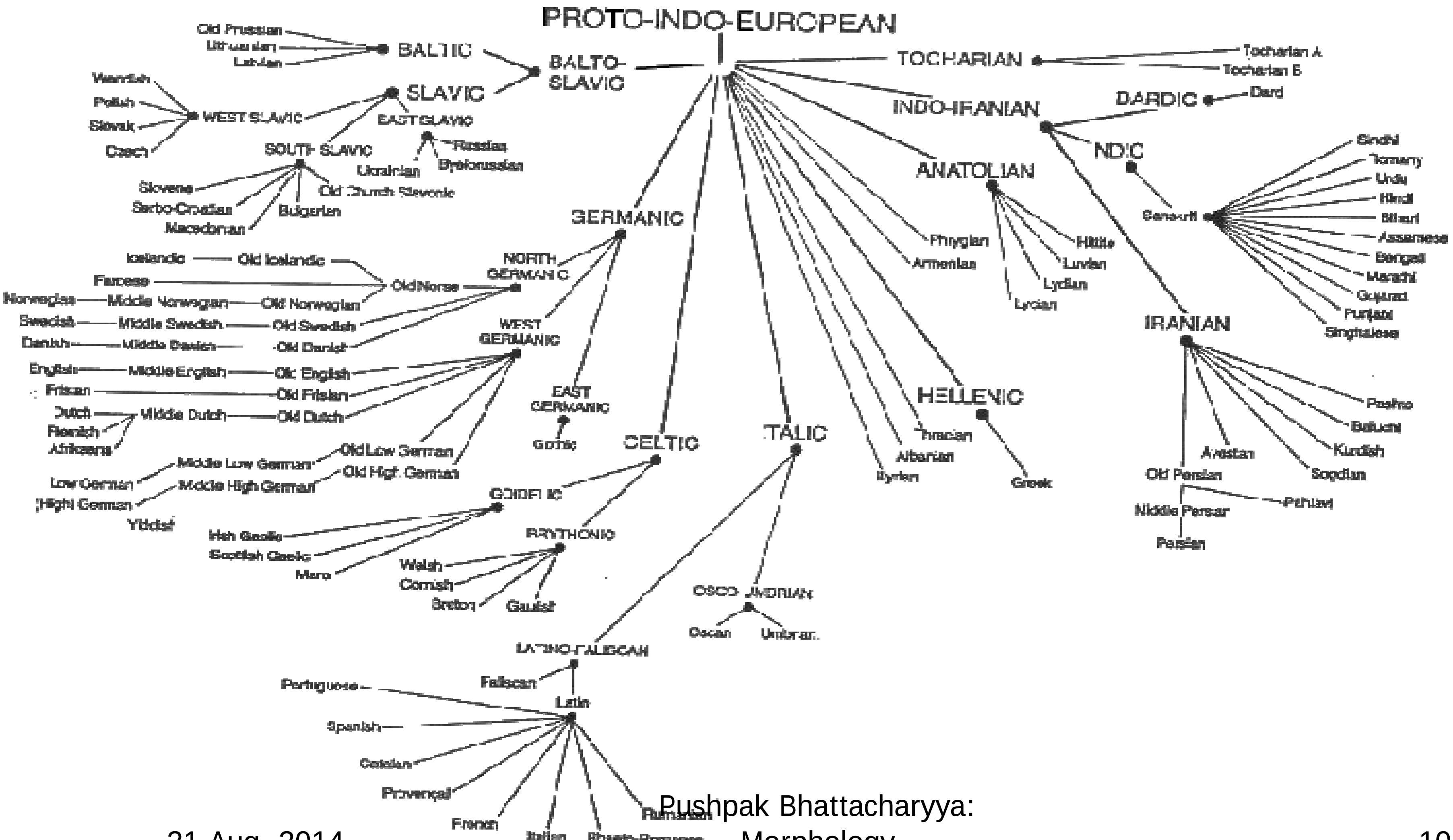
- **Turkish:**

  hazirlanmis plan

  prepare-past plan

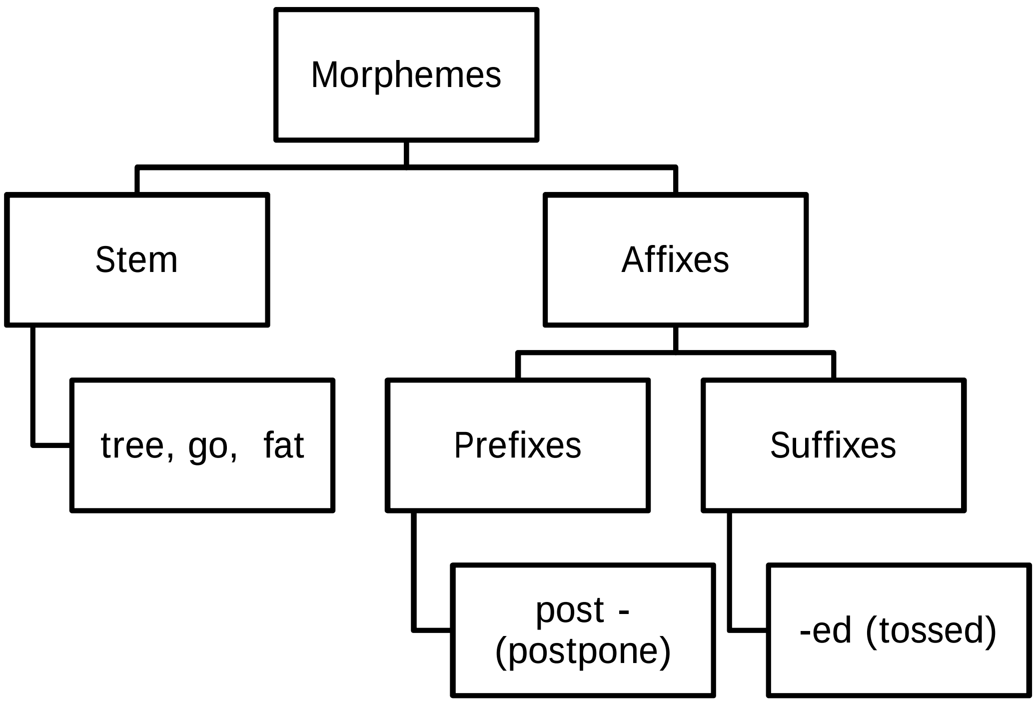
  The plan which has been prepared

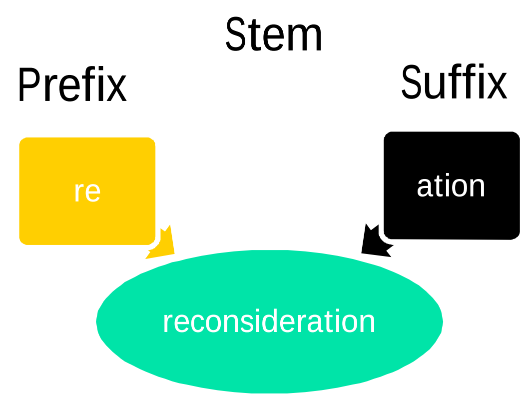
Language Typology



Morphemes

- Smallest meaning bearing units constituting a word





Case of Verbal Inflection

Morphological Form Classes	Regularly Inflected Verbs				Irregularly Inflected Verbs		
Stem	Jump	Parse	Fry	Sob	Eat	Bring	Cut
-s form	Jumps	Parses	Fries	Sobs	Eats	Brings	Cuts
-ing participle	Jumping	Parsing	Frying	Sobbing	Eating	Bringing	Cutting
Past form	Jumped	Parsed	Fried	Sobbed	Ate	Brought	Cut
–ed participle	Jumped	Parsed	Fried	Sobbed	Eaten	Brought	Cut

Forms governed by spelling rules
Idiosyncratic forms

General Features of Words

- They have phonological features
- They carry grammatical information.
- They carry semantic information.

For the word “dog”

IPA: *dɒg*

Grammatical: +*N*, +*sg*, *pl_s*

Semantic: +*animate*, +*mammal* (from lexical resources)

The goal of word level analysis

- ***The basic goal of word level linguistics is to segment and identify all phonemes and morphemes.***
- A phoneme is a minimal distinctive unit of sound of a language: *pin vs. bin*
- A morpheme is a minimal meaningful unit of a language: *play-ed*

Item-and-arrangement vs. Item-and-process

- Item-and-arrangement
  - Affix-driven view
  - Emphasis on the concatenation of affixes.
  - Syntax regulates morphological shapes.
- Item-and-process
  - Stem-driven view
  - Emphasis on the process of modification of the stem.
  - Morphology accumulates syntax.

Item and Arrangement example:

Kridanta processing in Marathi

Ganesh Bhosale, Subodh Kembhavi, Archana Amberkar, Supriya Mhatre, Lata Popale and Pushpak Bhattacharyya, *Processing of Participle (Krudanta) in Marathi*, International Conference on Natural Language Processing (ICON 2011), Chennai, December, 2011.

Kridanta and Taddhita

- Kridantas: verb derived (examples coming)

- Taddhitas: other POS derived
  - ghar → gharvaale

Kridantas can be in multiple POS categories

- **Nouns**

Verb	Noun
वाच {vaach}{read}	वाचणे {vaachaNe}{reading}
उतर {utara}{climb down}	उतरण {utaraN}{downward slope}

- **Adjectives**

Verb	Adjective
चाव {chav}{bite}	चावणारा {chaavaNaara}{one who bites}
खा {khaa} {eat}	खाल्लेले {khallele} {something that is eaten}.

Kridantas derived from verbs
(cont.)

- **<u>Adverbs</u>**

Verb	Adverb
पळ {paL}{run}	पळ**ताना** {paLataanaa}{while running}
बस {bas}{sit}	बस**ून** {basun}{after sitting}

Kridanta Types

***Kridanta* Type**	**Example**	**Aspect**
“णे” *{Ne-Kridanta}*	vaachNyaasaaThee pustak de. (Give me a book for reading.) For reading book give	Perfective
“ला” *{laa-Kridanta}*	Lekh vaachalyaavar saaMgen. (I will tell you that after reading the article.) Article after reading will tell	Perfective
“ताना” *{Taanaa-Kridanta}*	Pustak vaachtaanaa te lakShaat aale. (I noticed it while reading the book.) Book while reading it in mind came	Durative
“लेला” *{Lela-Kridanta}*	kaal vaachlele pustak de. (Give me the book that (I/you) read yesterday.) Yesterday read book give	Perfective
“ऊन”*{Un-Kridanta}*	pustak vaachun parat kar. (Return the book after reading it.) Book after reading back do	Completive
“णारा”*{Nara-Kridanta}*	pustake vaachNaaRyaalaa dnyaan miLte. (The one who reads books, gets knowledge.) Books to the one who reads knowledge gets	Stative
“वे” *{ve-Kridanta}*	he pustak pratyekaane vaachaave. (Everyone should read this book.) This book everyone should read	Inceptive
“ता” *{taa-Kridanta}*	to pustak vaachtaa vaachtaa zopee gelaa. (He fell asleep while reading a book.) He book while reading to sleep went	Stative

FSM based kridanta processing

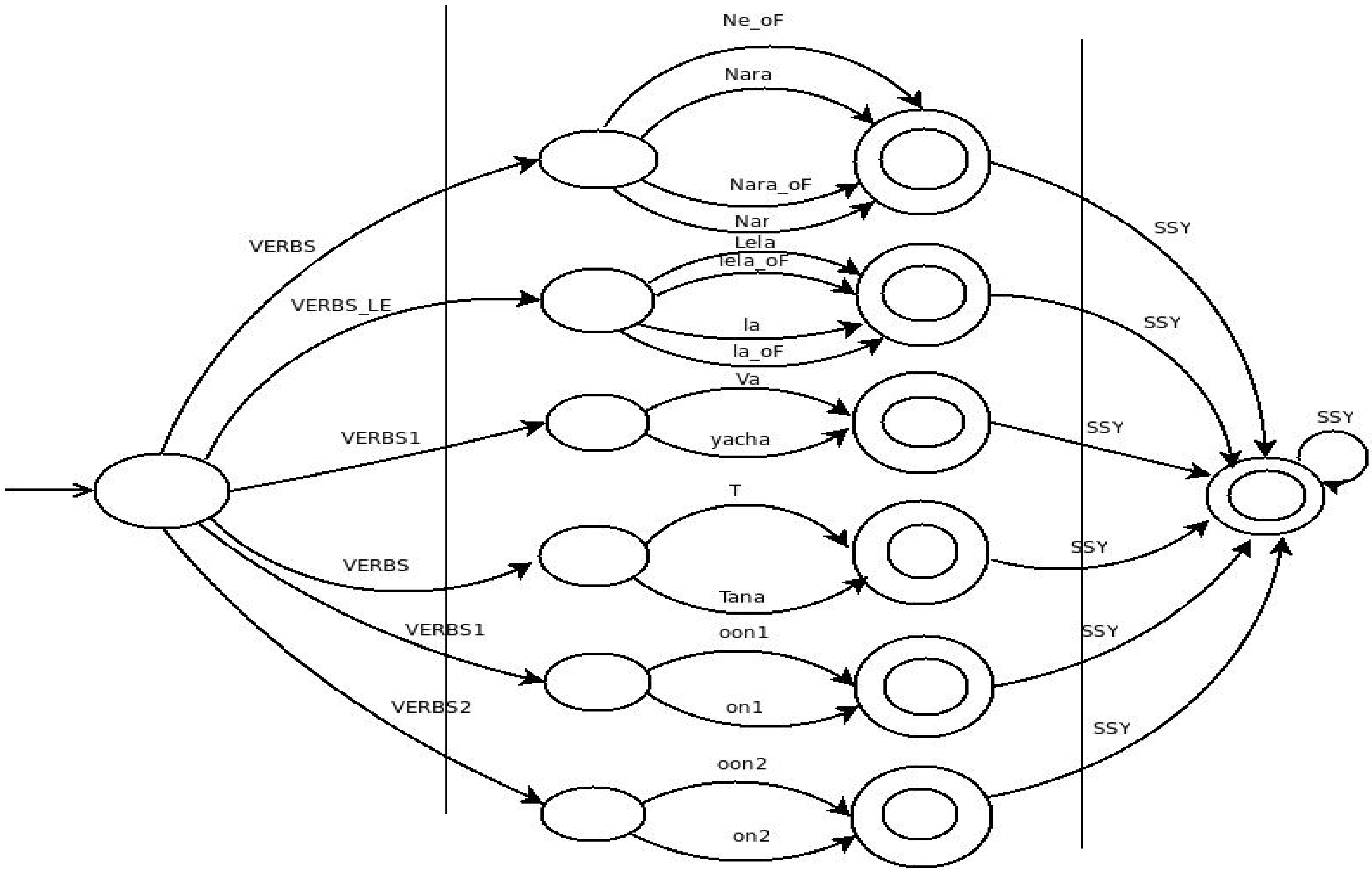
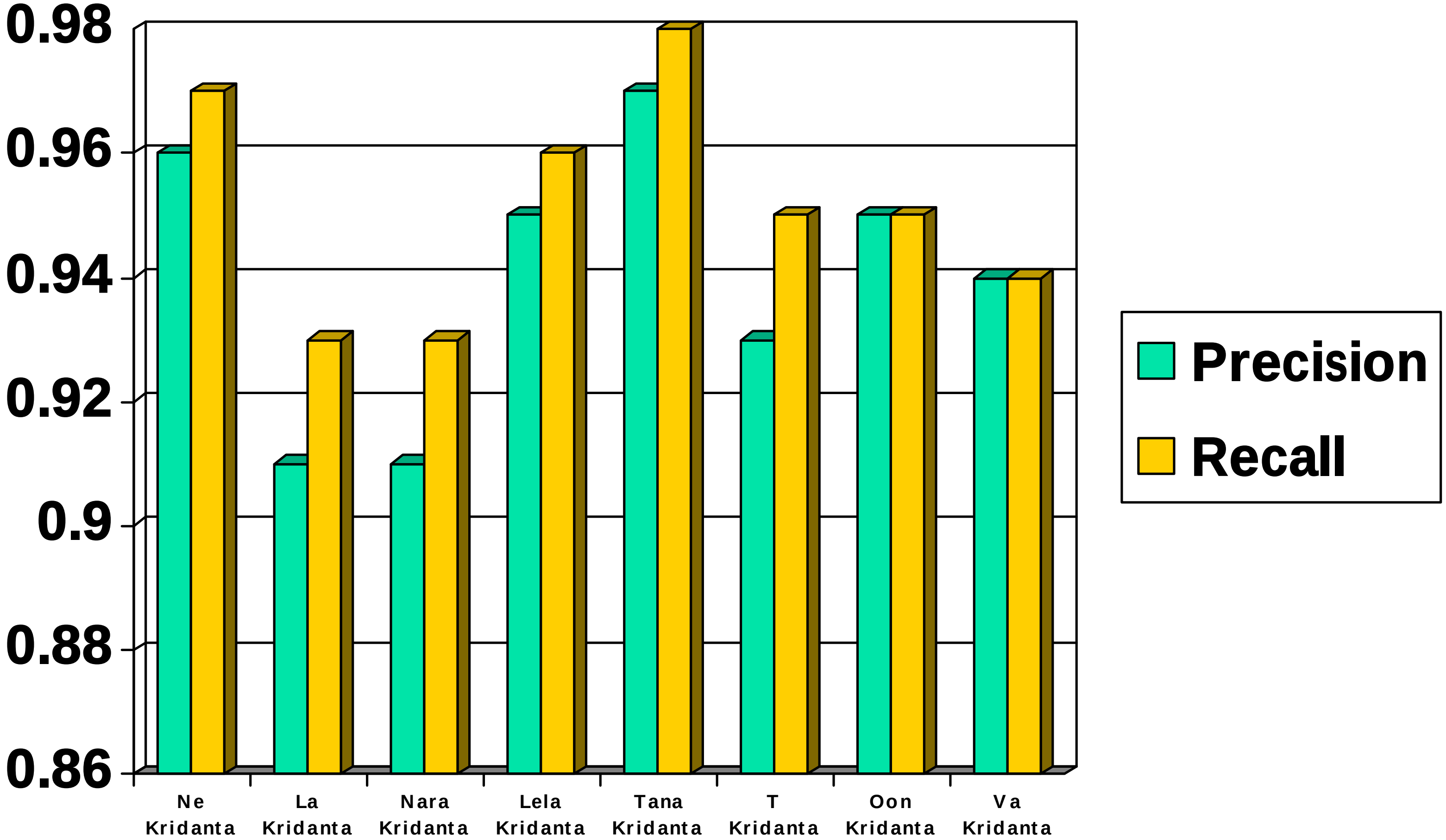


Fig. Morphotactics FSM for *Kridanta* Processing

Accuracy of Kridanta Processing: Direct Evaluation



3 classes of languages: morphology wise

- **Isolating**
  - Chinese, Vietnamese…
  - Words usually do not take affixes; tone and syntactic positions regulate their meaning

- **Agglutinative**
  - Odia, Hindi…
  - Words are constituted of multiple affixes

- **Inflectional**
  - Sanskrit, French, Italian…
  - Words conceptually contain functional features; they are not isolable.

Key notions

- #Morpheme per words
  - *Will go (1:1)*
  - *jaauMgaa (2:1)*
- Degree of fusions between adjacent morpheme
  - *None: no + one*
  - *राजर्षि (raajaRShi): राजा + ऋषि (raja + RShi)*

Morpheme classes

- Formal Classes:
  - Free vs. Bound/ Affixial
- Bound/Affix:
  - Prefix: *en-courage*, Suffix: *en-courage-ment*
  - Infix: Examples from Tagalog
    - *aral um-aral* 'teach'
    - *sulat s-um-ulat 'write' *um-sulat*
    - *Gradwet gr-um-adwet 'graduate' *um-gradwet*
- Functional Classes: Derivational: *Sing-er* Inflectional: *Sing-er-s*

Non-concatenative morphology

- Semitic languages: Arabic, Amharic, Hebrew, Tigriniya, Maltese, Syriac
- Word formation from ***radicals*** and ***patterns***
- *k-t-b: katab (to write), kAtib (writer/author/scribe), maktuwb (written/letter), maktab (office), maktabah (library)*

Derivation vs. Inflection

- Derivation typically (but not always) changes the word class
  - *write* (V) → *writer* (N)
  - But, *guitar* (N) → *guitarist* (N)
- Inflection typically (but not always) preserves the class
  - *write* (V) → *writes* (V)
  - But, *written* (J) *matter*

Derivational and inflectional morphemes

- Derivational morphemes:
  - -al, -able, de-, en-, -ence, -er, -full, -ish, -ity, -ize, -ness, -ment, -tion, -y…
- Inflectional morphemes:
  - -s, -ed, -en, -ing…

An NLP and IR Perspective

A Layered view of NLP that has come to be accepted

Discourse

Pragmatics

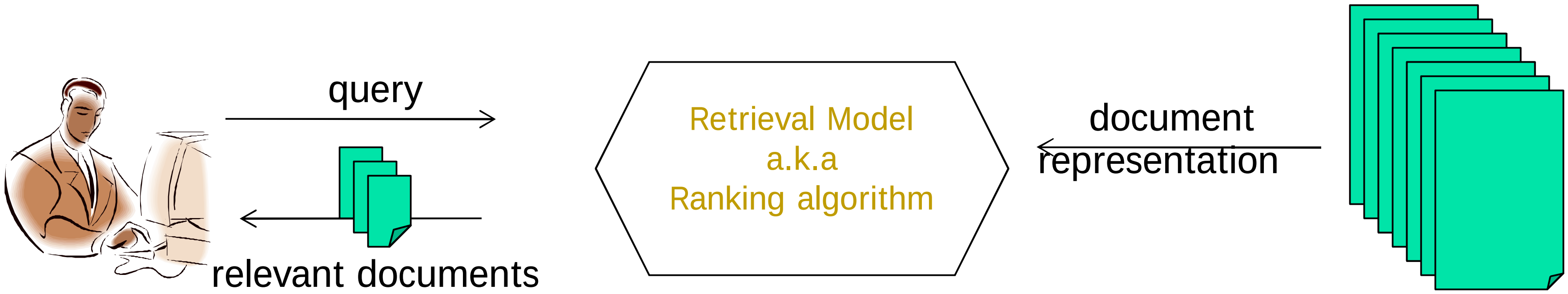
Semantic Processing

Parsing

Shallow Parsing (POS, Chunk, Verb Group)

Morphology

Classical Information Retrieval (Simplified)



late 1960's

40+ years of work in designing better models
- Vector space models
- Binary independence models
- Network models
- Logistic regression models
- Bayesian inference models
- Hyperlink retrieval models

2010

Nuts and bolts question: Morphology or Stemming? (1/2)

- NLP: Morphological Analysis; IR: stemming
- Normalize morphologically related words (e.g., swimmer, swam, swimming); else matching prevented in full text retrieval
- Stemming: an approximation to morpheme identification

Nuts and bolts question: Morphology or Stemming? (2/2)

- Definitely helps
  - Seminal study in “D. Harman. How effective is stemming? JASIS,42(1):7–15, 1991”
- Three broad classes of morphological processes result in surface forms that impair effective retrieval
  - Inflection, derivation and word formation.

Rule Based Stemming vs. Statistical Stemming (1/2)

- Rule-based stemming: based on linguistically inspired transformations
  - Snowball: stemming compiler (http://snowball.tartarus.org/)
  - Given a language specific rule set the compiler produces source code that transforms surface forms into stems

Rule Based Stemming vs. Statistical Stemming (2/2)

- Statistical stemmers: language neutral
  - Morphessor (http://www.cis.hut.fi/projects/morpho/)
  - Requires only a list of words
  - Based on Minimum Description Length Principle (Goldsmith 2001)

McNamee SIGIR 2009: Addressing Morphology Variations in IR: test collections for 18 languages

	Language	Queries	Documents	Evaluation
AR	Arabic	75	383,872	TREC '01-'02
BG	Bulgarian	149	85,427	CLEF '05-'07
BN	Bengali	50	123,040	FIRE '08
CS	Czech	50	81,735	CLEF '07
DE	German	192	294,805	CLEF '00-'03
EN	English	367	87,653	CLEF '00-'07
ES	Spanish	156	454,041	CLEF '01-'03
FA	Farsi	50	166,774	CLEF '08
FI	Finnish	120	55,344	CLEF '02-'04
FR	French	333	177,450	CLEF '00-'06
HI	Hindi	45	95,213	FIRE '08
HU	Hungarian	148	49,530	CLEF '05-'07
IT	Italian	181	157,558	CLEF '00-'03
MR	Marathi	49	99,359	FIRE '08
NL	Dutch	156	190,605	CLEF '01-'03
PT	Portuguese	146	210,734	CLEF '04-'06
RU	Russian	62	16,715	CLEF '03-'04
SV	Swedish	102	142,819	CLEF '02-'03

Performance relative to *words* baseline

	words	*snow*		*trun5*		*4-grams*		*5-grams*		Top method	
AR	0.2054			0.2148	+4.6%	0.2731$^{\blacktriangle}$	+33.0%	0.2356$^{\triangle}$	+14.7%	0.2731$^{\blacktriangle}$	+33.0%
BG	0.2164			0.2959$^{\blacktriangle}$	+36.7%	0.3105$^{\blacktriangle}$	+43.5%	0.2820$^{\blacktriangle}$	+30.3%	0.3105$^{\blacktriangle}$	+43.5%
BN	0.2630			0.3058$^{\triangle}$	+16.3%	0.3247$^{\triangle}$	+23.5%	0.3173$^{\triangle}$	+20.6%	0.3247$^{\triangle}$	+23.5%
CS	0.2270			0.3005$^{\blacktriangle}$	+32.4%	0.3294$^{\blacktriangle}$	+45.1%	0.3223$^{\blacktriangle}$	+42.0%	0.3329$^{\blacktriangle}$	+46.7%
DE	0.3303	0.3695$^{\blacktriangle}$	+11.9%	0.3656$^{\blacktriangle}$	+10.7%	0.4098$^{\blacktriangle}$	+24.1%	0.4201$^{\blacktriangle}$	+27.2%	0.4201$^{\blacktriangle}$	+27.2%
EN	0.4060	0.4373$^{\blacktriangle}$	+7.7%	0.4216$^{\triangle}$	+3.8%	0.3990	-1.7%	0.4152	+2.3%	0.4373$^{\blacktriangle}$	+7.7%
ES	0.4396	0.4846$^{\blacktriangle}$	+10.2%	0.4666$^{\triangle}$	+6.1%	0.4597	+4.6%	0.4609$^{\triangle}$	+4.8%	0.4846$^{\blacktriangle}$	+10.2%
FA	0.3617			0.3645	+0.8%	0.3986$^{\triangle}$	+10.2%	0.3821	+5.6%	0.3986$^{\triangle}$	+10.2%
FI	0.3406	0.4296$^{\blacktriangle}$	+26.1%	0.4652$^{\blacktriangle}$	+36.6%	0.4989$^{\blacktriangle}$	+46.5%	0.5078$^{\blacktriangle}$	+49.1%	0.5078$^{\blacktriangle}$	+49.1%
FR	0.3638	0.4019$^{\blacktriangle}$	+10.5%	0.3953$^{\blacktriangle}$	+8.7%	0.3844$^{\triangle}$	+5.7%	0.3930$^{\blacktriangle}$	+8.0%	0.4019$^{\blacktriangle}$	+10.5%
HI	0.2429			0.2914$^{\blacktriangle}$	+20.0%	0.3305$^{\blacktriangle}$	+36.1%	0.3271$^{\blacktriangle}$	+34.7%	0.3305$^{\blacktriangle}$	+36.1%
HU	0.1976			0.3082$^{\blacktriangle}$	+56.0%	0.3746$^{\blacktriangle}$	+89.6%	0.3624$^{\blacktriangle}$	+83.4%	0.3746$^{\blacktriangle}$	+89.6%
IT	0.3749	0.4178$^{\blacktriangle}$	+11.4%	0.3963	+5.7%	0.3738	-0.3%	0.3997$^{\triangle}$	+6.6%	0.4178$^{\blacktriangle}$	+11.4%
MR	0.2572			0.3477$^{\blacktriangle}$	+35.2%	0.4114$^{\blacktriangle}$	+60.0%	0.3739$^{\blacktriangle}$	+45.4%	0.4164$^{\blacktriangle}$	+61.8%
NL	0.3813	0.4003$^{\triangle}$	+5.0%	0.3946	+3.5%	0.4219$^{\blacktriangle}$	+10.6%	0.4243$^{\blacktriangle}$	+11.3%	0.4243$^{\blacktriangle}$	+11.3%
PT	0.3162			0.3423$^{\triangle}$	+8.3%	0.3358	+6.2%	0.3524$^{\blacktriangle}$	+11.4%	0.3524$^{\blacktriangle}$	+11.4%
RU	0.2671			0.3739$^{\blacktriangle}$	+40.0%	0.3406$^{\blacktriangle}$	+27.5%	0.3330$^{\triangle}$	+24.7%	0.3739$^{\blacktriangle}$	+40.0%
SV	0.3387	0.3756$^{\blacktriangle}$	+10.9%	0.3770$^{\triangle}$	+11.3%	0.4236$^{\blacktriangle}$	+25.1%	0.4271$^{\blacktriangle}$	+26.1%	0.4271$^{\blacktriangle}$	+26.1%

Observation from *McNamee, SIGIR 2009*

- Rule-based stemming using Snowball rule sets performed well in English and the Romance family
- In those languages it tended to perform better than n-grams
- In highly complex languages, it proved essential to cater for morphology to obtain the best results

Rule Based Stemming: Porter Stemmer

Motivated by IR

- Terms with a common stem will usually have similar meanings, for example:
  - *CONNECT CONNECTED CONNECTING CONNECTION CONNECTIONS*

- Conflation into a single term improves IR performance
- Removal of the various suffixes -ED, -ING, -ION, IONS to leave the single term CONNECT
- Reduce the size and complexity of the data in the system

MA vs. Stemming

- “In any suffix stripping program for IR work, two points must be borne in mind. Firstly, the suffixes are being removed simply to improve IR performance, and not as a linguistic exercise. This means that it would not be at all obvious under what circumstances a suffix should be removed, even if we could exactly determine the suffixes of a word by automatic means.”
- (quote from Porter’s original paper, 1979)
- Genesis of unsupervised morph analysis

Basic approach of suffix stripping

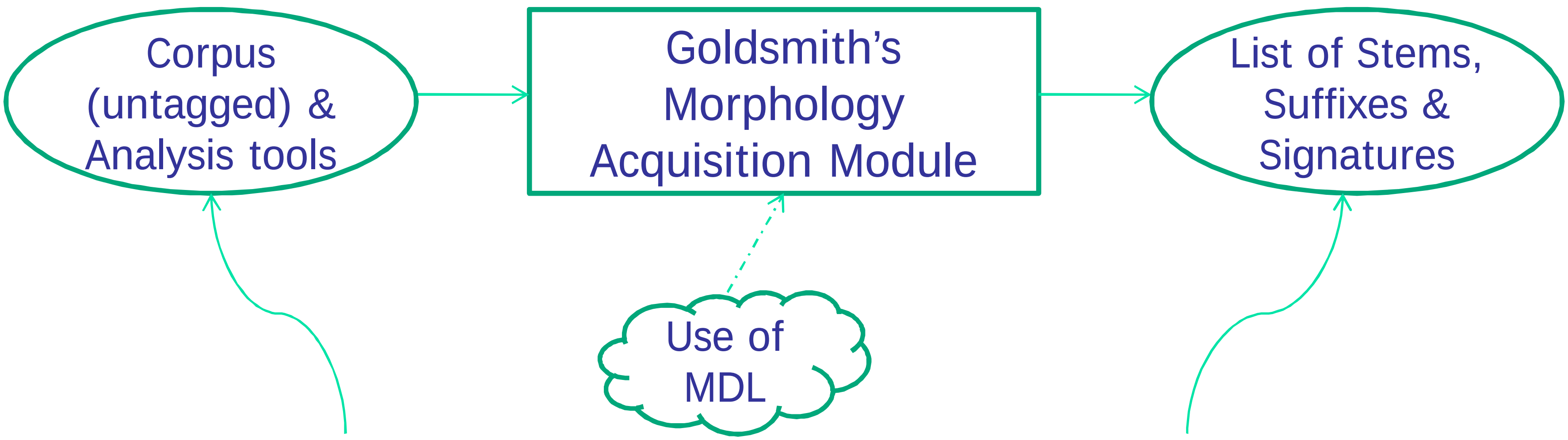
- Suffix list plus Rules under which they operate
- E.g.
  - (m>1) EED -> EE ('VC' combination repeated m times)
    - feed -> feed (m=1)
    - agreed -> agree (m=2; *'agr'* and *'eed'*)
  - (*v*) ED -> (contains a vowel)
    - plastered -> plaster
    - bled -> bled (contains no vowel)
  - (*v*) ING ->
    - motoring -> motor
    - sing -> sing (contains no vowel)

Minimum Description Length based Unsupervised Morphology

-Goldsmith 2001

Implemented as ***Morfessor***

About the approach…



Criteria: *satisfying the motive of “Unsupervised Learning”*

Criteria: *matching the output given by a human morphologist*

Some terms...

- ***Signature:*** a list of all the suffixes with which a stem appears in the given corpus.
  - A stem is unique to a signature, but a suffix is not.
  - e.g.: ***{attack, boil, borrow} {NULL.ed.er.ing.s}***

- ***MDL:*** Minimum Description Length, aims at picking up that model or representation for the data, which gives the most compact description of the data, including the description of the model itself.

The approach...

Step:1 Assign a probability distribution to the sample space from which the data is assumed to be drawn

Step:2 Assign a compressed length to the data, which is said to be the "optimal compressed length of the data"

Step:3 Assign a compressed length to the model of the data

Step:4 Select the optimal analysis, the one for which length of compressed data + length of model is the smallest

MDL analysis

- Suppose the corpus has the words:
  - cat, cats, dog, dogs, hat, hats, laugh, laughed, laughing, laughs, walk, walked, walking, walks, Jim
- A start: Lets count letters
  - It gives a total of 72 letters!!! (≈72*8 = 576 bits!!!)

Separate stems and suffixes

Stems:	Suffixes:	Unanalyzed:
cat	s	Jim
dog	ed	**Total: 3**
hat	ing	
laugh	**Total: 6**	
walk		
Jim		
Total: 21		

- Total of 30 letters!!! (≈30*8 bits!!!)
- A saving of approx. 336 bits
- But what about stem suffix association?

Model using signatures (for English)

A. Stem-list

1. *cat*
2. *dog*
3. *hat*
4. *laugh*
5. *walk*
6. *Jim*

B. Suffix-list

1. *NULL*
2. *s*
3. *ed*
4. *ing*

C. Signature-list

Signature 1:

$$\begin{Bmatrix} ptr(cat) \\ ptr(dog) \\ ptr(hat) \end{Bmatrix} \begin{Bmatrix} ptr(NULL) \\ ptr(s) \end{Bmatrix}$$

Signature 2:

$$\begin{Bmatrix} ptr(laugh) \\ ptr(walk) \end{Bmatrix} \begin{Bmatrix} ptr(NULL) \\ ptr(ed) \\ ptr(ing) \\ ptr(s) \end{Bmatrix}$$

Signature 3:

$$\{ptr(Jim)\}$$

Need to store only pointers?!

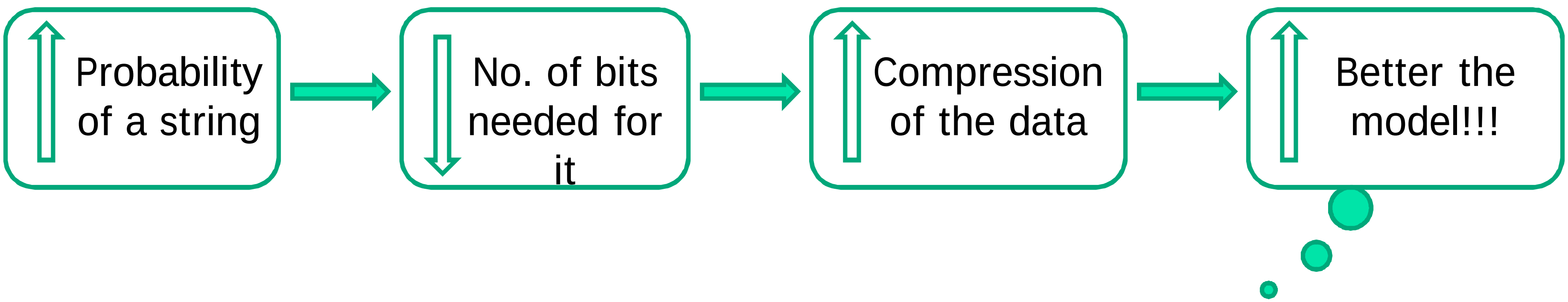
Some representations...

- t = stem — T = set of stems
- f = suffix — F = set of suffixes
- σ = signature — Σ = set of signatures

- $\langle T \rangle$, $\langle F \rangle$, etc. represent no. of members of the set
- $[t]$, $[f]$, etc. represent no. of occurrences of stem, suffix, etc. respectively.
- W = set of all words in the corpus
- $[W]$ = length of the corpus
- $\langle W \rangle$ = size of the vocabulary

Information Theoretic Principle

- The morphology that assigns the highest probability to the corpus is considered to be the best morphology



Human mediated stemming

Facilitating Multi-Lingual Sense Annotation:
Human Mediated Lemmatizer

Pushpak Bhattacharyya[1] ; Ankit Bahuguna[2];
Lavita Talukdar[3]; Bornali Phukan[4]

Background and Related Work

- **Lovins** (Lovins,1968): use of a manually developed list of 294 suffixes, each linked to 29 conditions, plus 35 transformation rules. For an input word, the suffix with an appropriate condition is checked and removed.
- **Porter stemmer** (Porter,1980): The most widely used algorithm for English language.
- **Plisson** (Plisson et,2008). proposed the *most accepted rule based approach* for lemmatization.

Background and Related Work (contd..)

- **Kimmo** (Karttunen,1983) is a two level morphological analyzer.
- **OMA** (Ozturkmenoglu,2012) is a Turkish morphological Analyzer.
- **Tarek El-Shishtawy**(El-Shishtawy,2012) proposed the first non statistical **Arabic** Lemmatizer.
- **Ramanathan and Rao**(Rao,2003) used manually sorted suffix list and performed longest match ***strippin****g* for building a Hindi stemmer.

Background and Related Work (contd..)

- **GRALE**(Loponen,2013) is a graph based lemmatizer for Bengali language.

- A Hindi Lemmatizer is proposed, where suffixes are stripped according to various rules and necessary addition of character(s) is done to get a proper root form (Paul, 2013).

Trie based Lemmatization with backtracking

- The scope of our work is **suffix based morphology.**

First or Direct Variant:

- First setup the data structure "**Trie**" using the words in the wordnet of a specific language.
- Next, we ***match byte by byte***, input word form and wordnet words.
- The **output** is all wordnet words retrieved after the maximum substring match.

Our Approach to lemmatization (Cont..)

Second or backtrack variant:

- The backtrack variant prints the results “n” level previous to the maximum matched prefix obtained in the “direct” variant of our lemmatizer

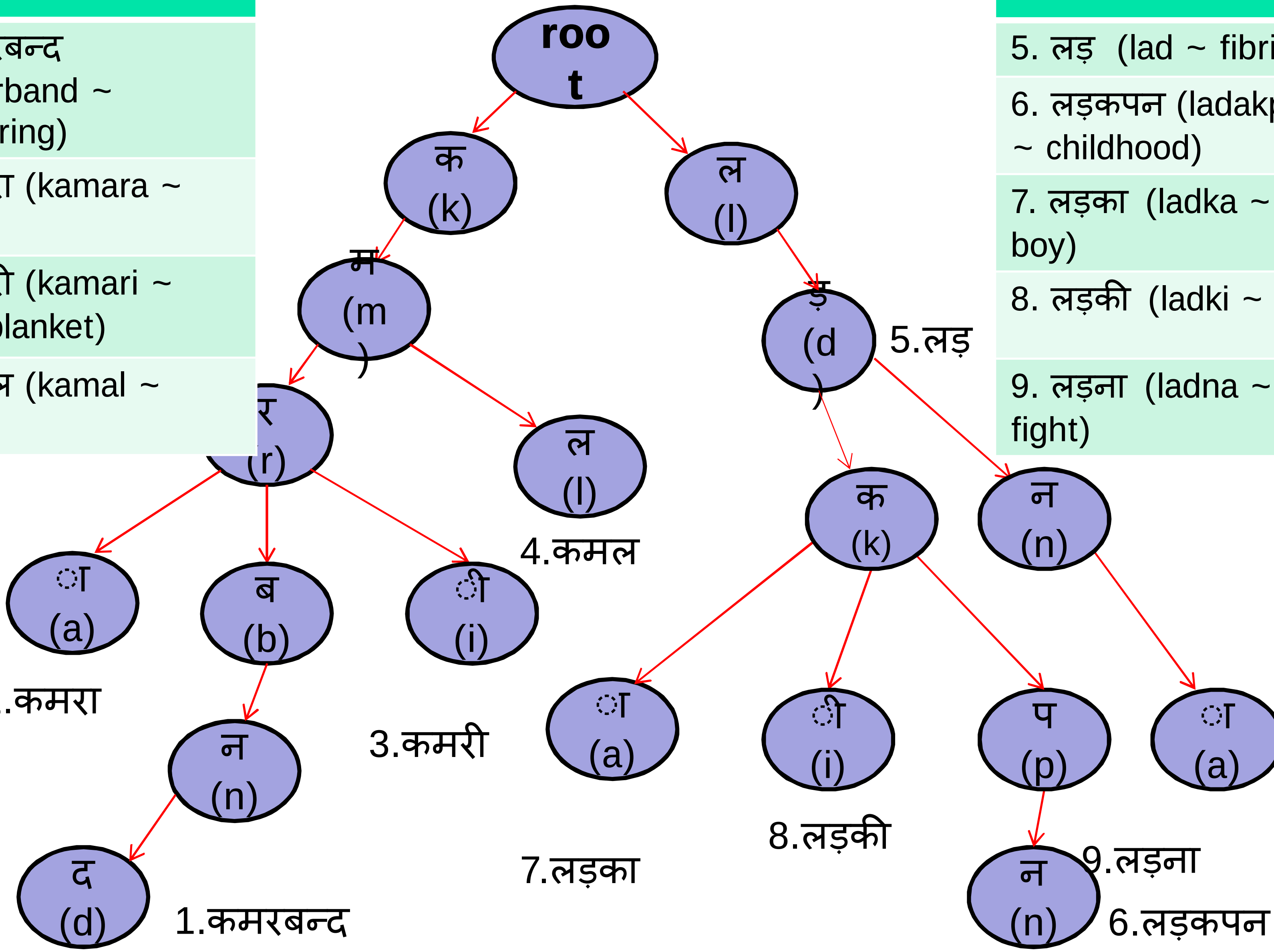
- The value of “n” is user controlled.

List of Words

1. कमरबन्द (kamarband ~ drawstring)
2. कमरा (kamara ~ room)
3. कमरी (kamari ~ small blanket)
4. कमल (kamal ~ Lotus)

List of Words

5. लड़ (lad ~ fibril)
6. लड़कपन (ladakpan ~ childhood)
7. लड़का (ladka ~ boy)
8. लड़की (ladki ~ girl)
9. लड़ना (ladna ~ fight)



Example: Direct Approach

- Inflected word “लड़कियाँ” (ladkiyan, *i.e.,* girls).Our lemmatizer gives the following results:

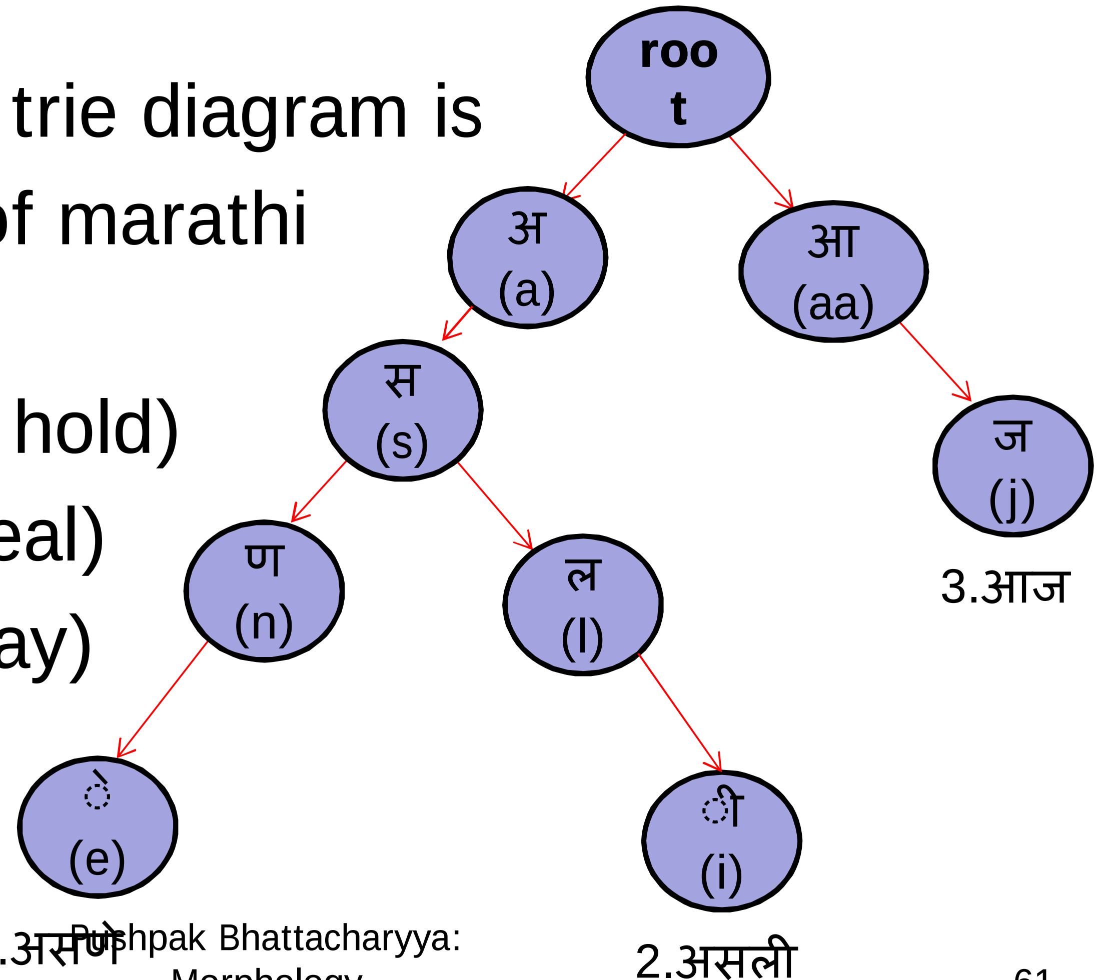
- (ल लड़ लड़का लड़की लड़कपन लड़कोरी लड़कौरी).

- From this result set, a trained lexicographer can pick up the root word as “लड़की” (ladki, i.e., girl).

Example: Backtracking

Backtracking:

- In figure a sample trie diagram is shown consisting of marathi words.
  1. असणे (asane ~ hold)
  2. असली (asali ~ real)
  3. आज (aaj ~ today)



Backtracking

- We take the example of “असलेले”(aslele) which is an inflected form of the Marathi word “असणे” (asane)
- In the first iterative procedure the word “असली”(asali) is given as output
  - not the correct result
- Through backtracking (असणे असंभव असंयत असंयम असंख्य असंगती असंमती असंयमी असतेपण असंतोषी असंबद्ध असंयमित)

Ranking lemmatizer Results

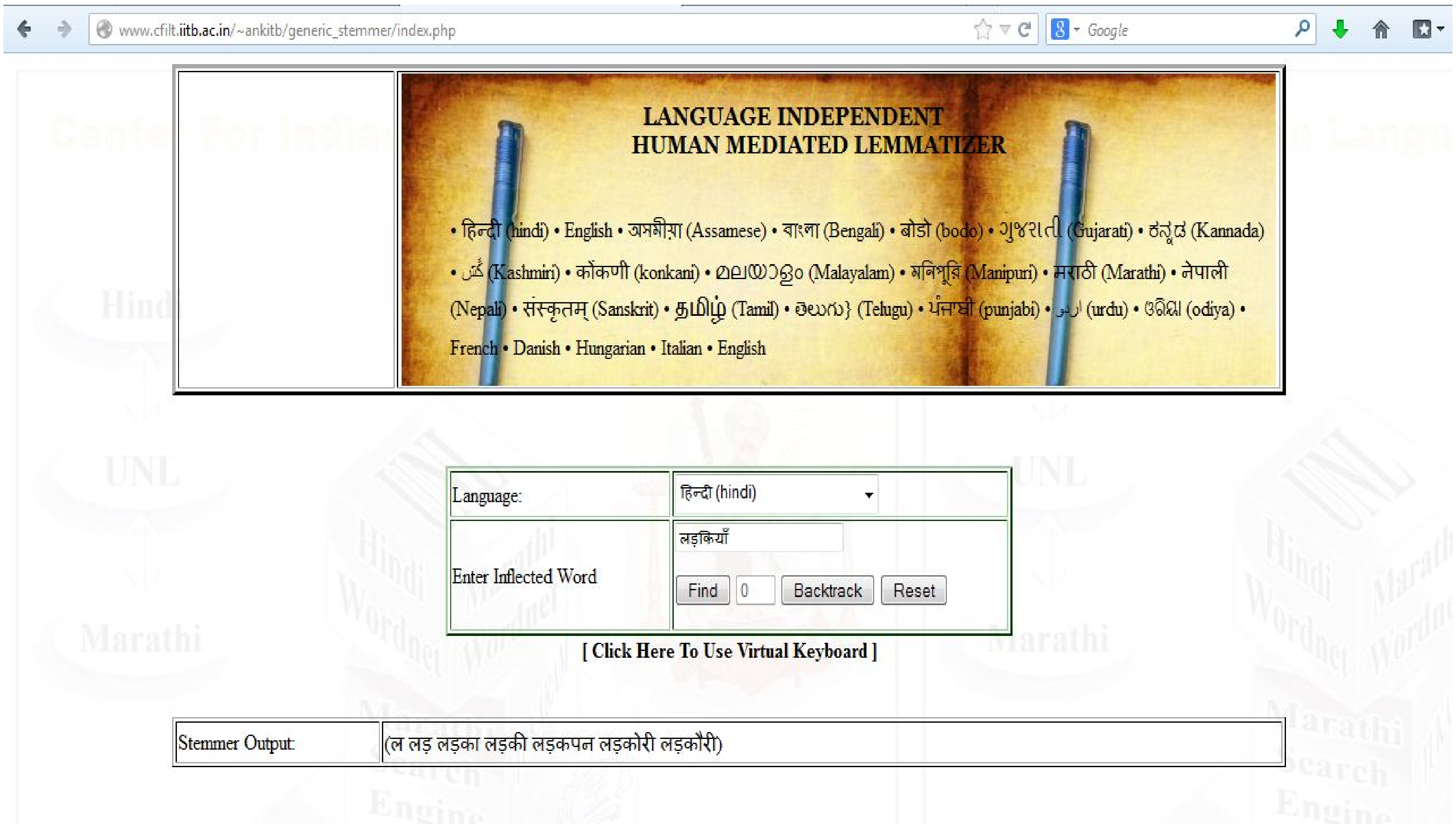
1. Only those results are displayed whose length is less than or equal to inflected word.

2. The filtered results are sorted on the basis of length.

Implementation

- **on-line interface** and a downloadable **Java** based **executable jar**.
- Allows input from **18 different Indian languages** and **5 European languages**.
- “**Backtrack**” feature allows backtracking up to 8 levels.
- **facility to upload** a text document

Online Interface



Experiments and Results

- **Assumption:** consider 'correct' if the desired word appears in the first 10 outputs
- For Hindi, Marathi, Bengali, Assamese, Punjabi and Konkani: gold standard data used
- For Dravidian languages and European languages we had to perform manual evaluation.

Results

Language	Corpus Type	Total words	Precision Value
Hindi	Health	8626	89.268
Hindi	Tourism	16076	87.953
Bengali	Health	11627	93.249
Bengali	Health	11305	93.199
Assamese	General	3740	96.791
Punjabi	Tourism	6130	98.347
Marathi	Health	11510	87.655
Marathi	Tourism	13176	85.620
Konkani	Tourism	12388	75.721
Malayalam*	General	135	100.00
Kannada*	General	39	84.165
Italian*	General	42	88.095
French*	General	50	94.00

(*) Denotes the languages evaluated manually

Error Analysis

Errors are due to following reasons:

1. **Agglutination** in Marathi and Dravidian languages: Marathi and Dravidian languages like Kannada and Malayalam show the process of agglutination.
2. **Suppletion**:

   For example the word “go ” has an irregular past tense form “went”.

Comparative Evaluation

- We have compared performance of our system with most commonly used lemmatizers, viz. Morpha, Snowball and Morfessor.

Corpus Name	Human mediated Lemmatizer	Morpha	Snowball	Morfessor
English-General	89.20	90.17	53.125	79.16
Hindi-General	90.83	NA	NA	26.14
Marathi-General	96.51	NA	NA	37.26

Summary

- **light weight** and **quick** to create. .
- The human annotator can chose the result

Future Work:

- Improvement of the ranking algorithm so the we can get the correct lemma within top 2 results.
- Integration of Human mediated lemmatizer to all languages sense marking tasks.

Resources

- http://www.cfilt.iitb.ac.in/indowordnet/
- http://www.cfilt.iitb.ac.in/wordnet/webhwn/
- http://www.cfilt.iitb.ac.in/Publications.html
- http://snowball.tartarus.org/
- http://www.cfilt.iitb.ac.in/wsd/annotated_corpus/
- http://www.en.wikipedia.org/wiki/Agglutination
- https://www.en.wikipedia.org/wiki/Suppletion
- http://www.cfilt.iitb.ac.in/~ankitb/ma/

Back to MDL

The actual MDL analysis(1/2)

- Length of the model is

  length(T) + length(F) + length(Σ)

- length(T) = $\log_2(\langle T\rangle) + \sum_{t \epsilon T} \text{length(t)} * \log_2 26$

$= \log_2(6) + \log_2 26 * (3+3+3+5+4+3)$

= 108 bits .... (i)

A. Stem-list

1. *cat*
2. *dog*
3. *hat*
4. *laugh*
5. *walk*
6. *Jim*

The actual MDL analysis(1/2)

- Length of the model is
  length(T) + length(F) + length(Σ)

- length(F) = $\log_2(\langle F \rangle) + \sum_{f \epsilon F} \text{length(f)} * \log_2 26$

$$= \log_2(4) + \log_2 26 * (1 + 2 + 3)$$

= 32 bits .... (ii)

B. Suffix-list

1. *NULL*
2. *s*
3. *ed*
4. *ing*

The actual MDL analysis(1/2)

- Length of the model is
  length(T) + length(F) + length(Σ)

- length(Σ) = $$\log_2(\langle\Sigma\rangle) + \sum_{\sigma \epsilon \Sigma} \left(\log_2(\langle T|\sigma\rangle) + \log_2(\langle F|\sigma\rangle) + \sum_{t \epsilon T(\sigma)} \log_2 \frac{[\mathrm{W}]}{[\mathrm{t}]} + \sum_{f \epsilon F(\sigma)} \log_2 \frac{[\sigma]}{[\mathrm{words}(\mathrm{f}) \cap \mathrm{words}(\sigma)]} \right)$$

The actual MDL analysis(1/2)

- length(Σ1) =

$$\log_2 3 + \log_2 2 + \log_2\left(\frac{15}{2} * \frac{15}{2} * \frac{15}{2}\right) + \log_2\left(\frac{6}{3} * \frac{6}{3}\right)$$

  = 2 + 1 + 9 + 2

  = 14 bits
- length(Σ2) = 1 + 2 + 4 + 8 = 15
- length(Σ3) = 1 + 4 = 5
- length(Σ) = $\log_2 3$ + 14 + 15 + 5

  = 36 bits..........(iii)

C. Signature-list

Signature 1:

$$\begin{Bmatrix} ptr(cat) \\ ptr(dog) \\ ptr(hat) \end{Bmatrix} \begin{Bmatrix} ptr(NULL) \\ ptr(s) \end{Bmatrix}$$

Signature 2:

$$\begin{Bmatrix} ptr(laugh) \\ ptr(walk) \end{Bmatrix} \begin{Bmatrix} ptr(NULL) \\ ptr(ed) \\ ptr(ing) \\ ptr(s) \end{Bmatrix}$$

Signature 3:

$$\{ptr(Jim)\}$$

The actual MDL analysis(1/2)

- Total length of the model is obtained by the summation of (i), (ii) and (iii), i.e.,

  108 + 32 + 36 = 176 bits

The actual MDL analysis(2/2)

- Length of the corpus:

$$-\sum_{w=t+f} [w] * (\log prob(\sigma(w)) + \log prob(t)$$

$$+ \log prob(f|\sigma(w)))$$

$$= -\sum_{w=t+f} [w] * \log\left(\frac{[\sigma]}{[w]} * \frac{[t]}{[\sigma]} * \frac{[f \text{ in } \sigma]}{[\sigma]}\right)$$

$$= -\sum_{w=t+f} [w] * \log\left(\frac{[t]}{[w]} * \frac{[f \text{ in } \sigma]}{[\sigma]}\right)$$

The actual MDL analysis(2/2)

$$= -\sum_{w=t+f} [w] * \log\left(\frac{[t]}{[w]} * \frac{[f \text{ in } \sigma]}{[\sigma]}\right)$$

$$= 2 * \log_2\left(\frac{15}{2} * \frac{6}{3}\right) + 2 * \log_2\left(\frac{15}{2} * \frac{6}{3}\right)$$

$$+ 2 * \log_2\left(\frac{15}{2} * \frac{6}{3}\right) + 4 * \log_2\left(\frac{15}{4} * \frac{8}{2}\right)$$

$$+ 4 * \log_2\left(\frac{15}{4} * \frac{8}{2}\right) + 1 * \log_2\left(\frac{15}{1}\right)$$

$$= 8 + 8 + 8 + 16 + 16 + 4 = 60 \text{ bits}$$

Corpus:
cat
cats
dog
dogs
hat
hats
laugh
laughed
laughing
laughs
walk
walked
walking
walks
Jim

The total size of the analysis...

- The total size is the summation of the size of the model and the size of the corpus, which is,

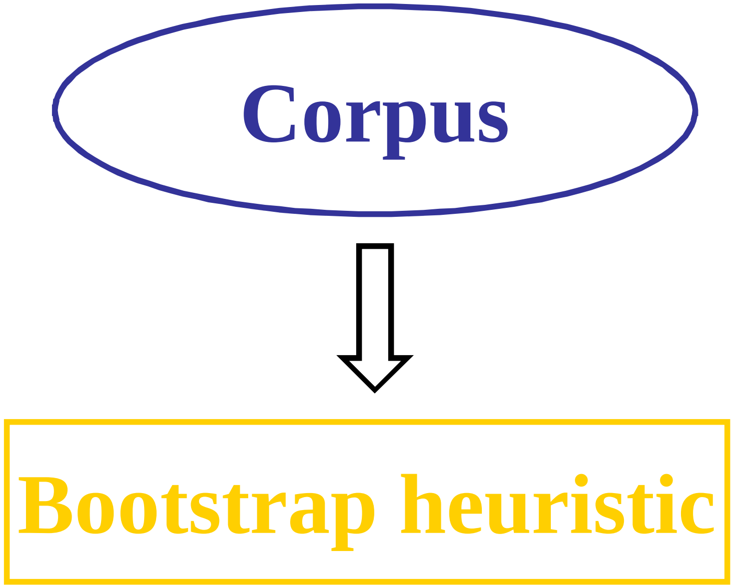
  176 bits (model) + 60 bits (corpus)

  = 236 bits!!!

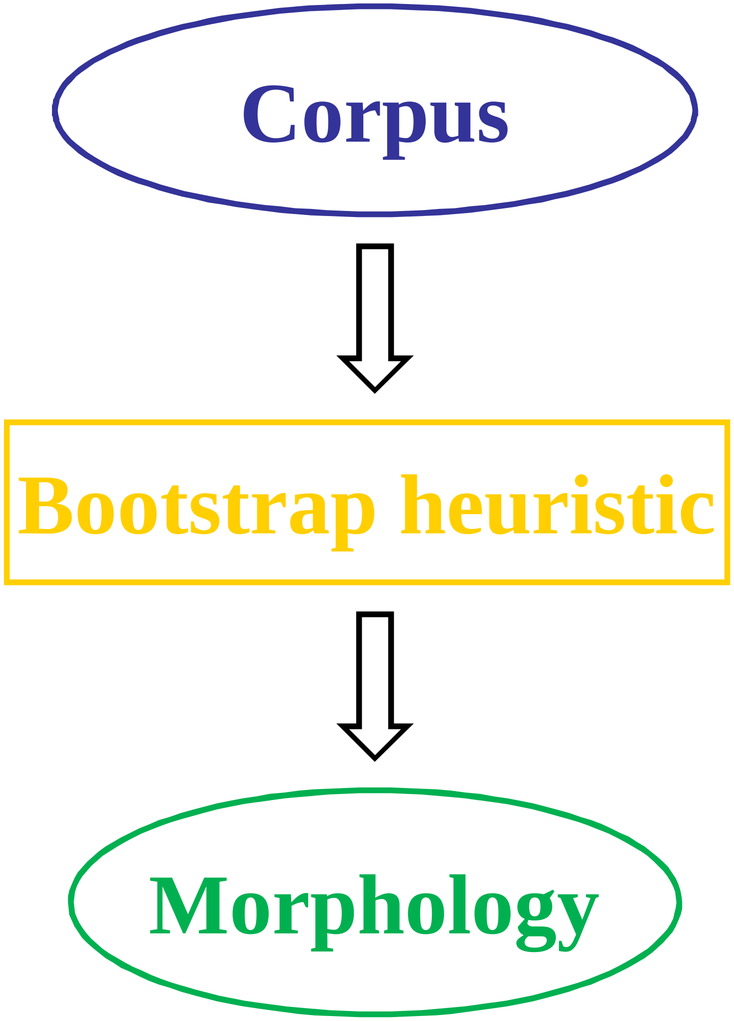
  - Which means a saving of 340 bits!!!

Corpus

Pick a large corpus from a language -- 5,000 to 1,000,000 words.



Feed it into the “bootstrapping” heuristic...



Out of which comes a preliminary morphology, which need not be superb.

Corpus

⇓

Bootstrap heuristic

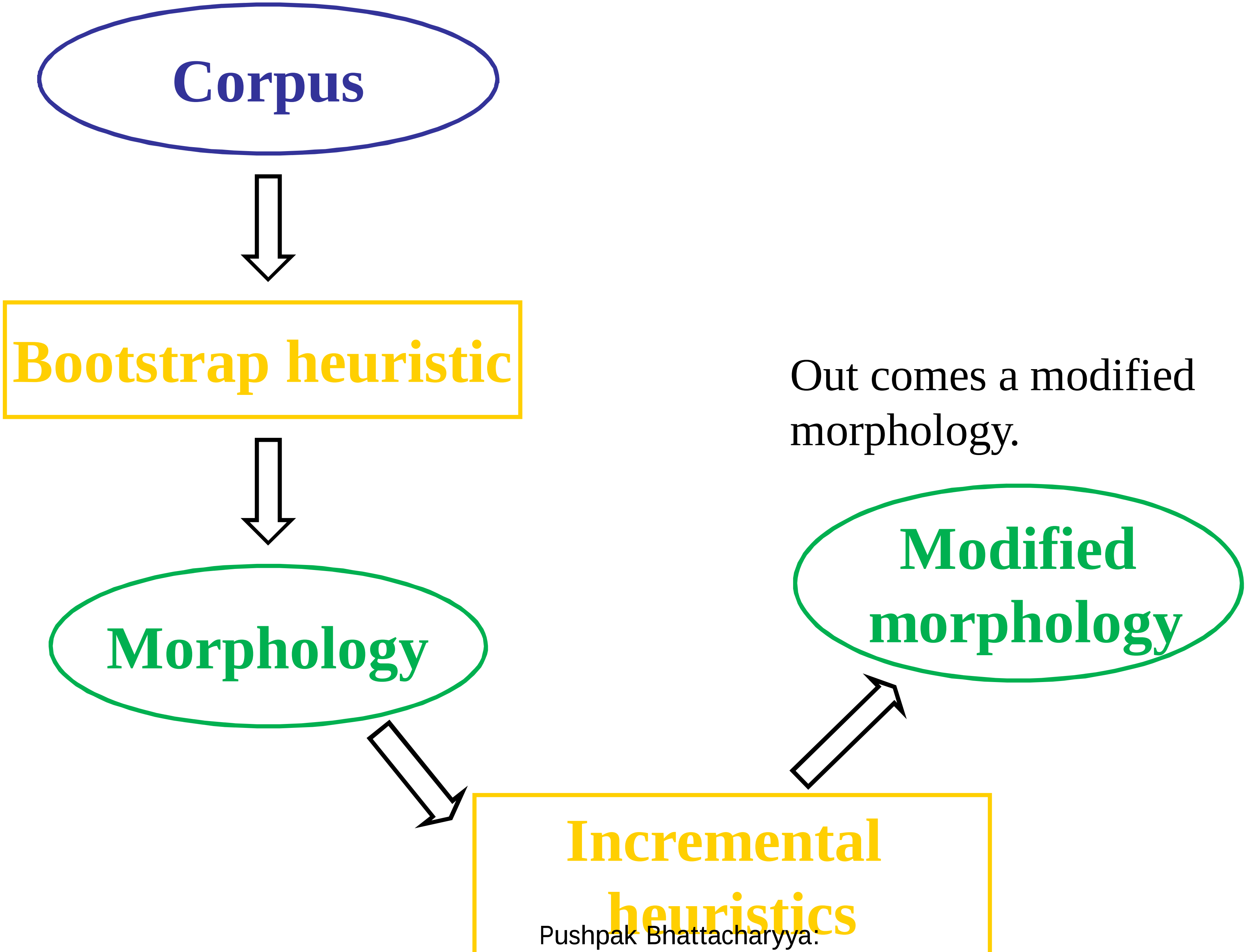
⇓

Morphology

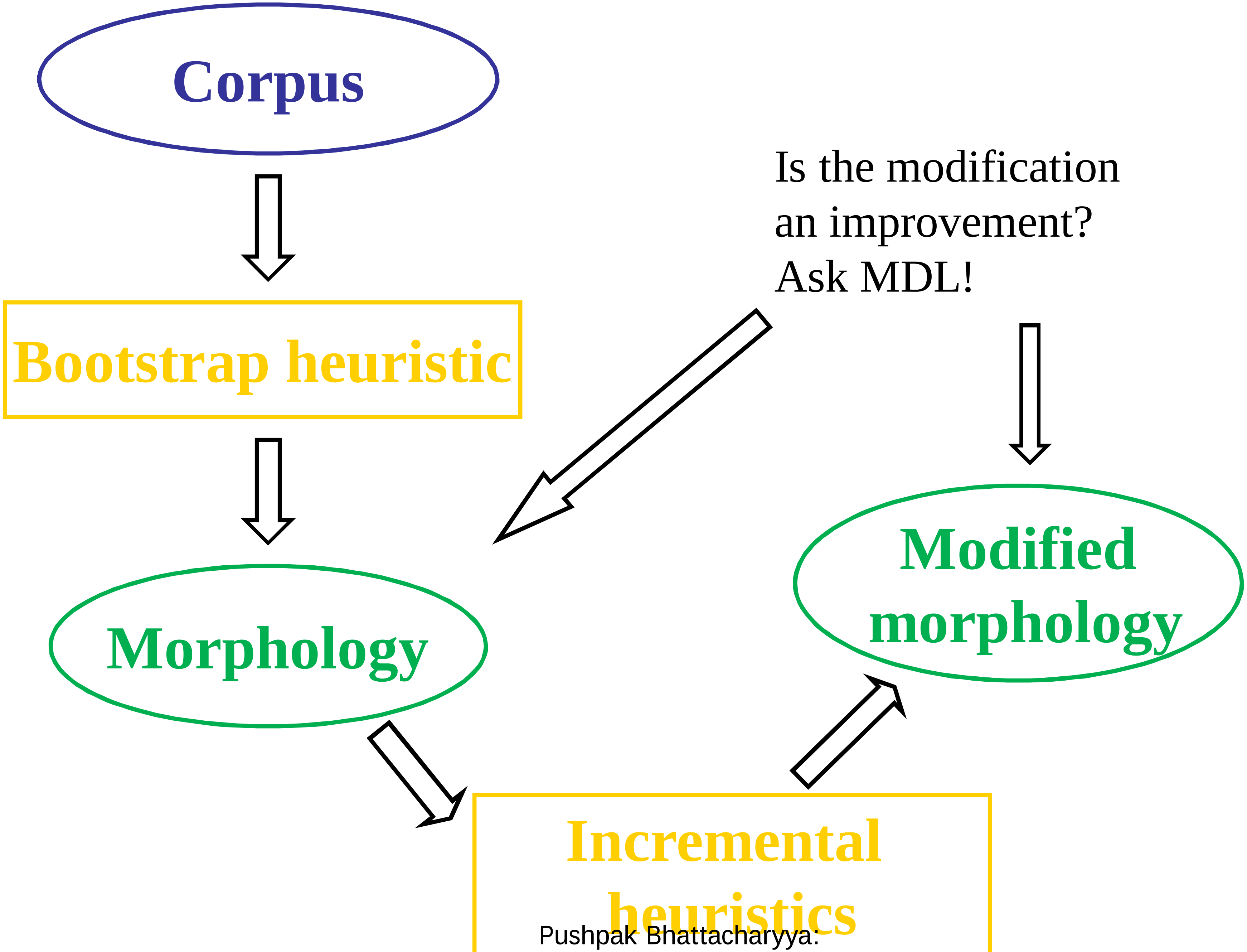
⇘

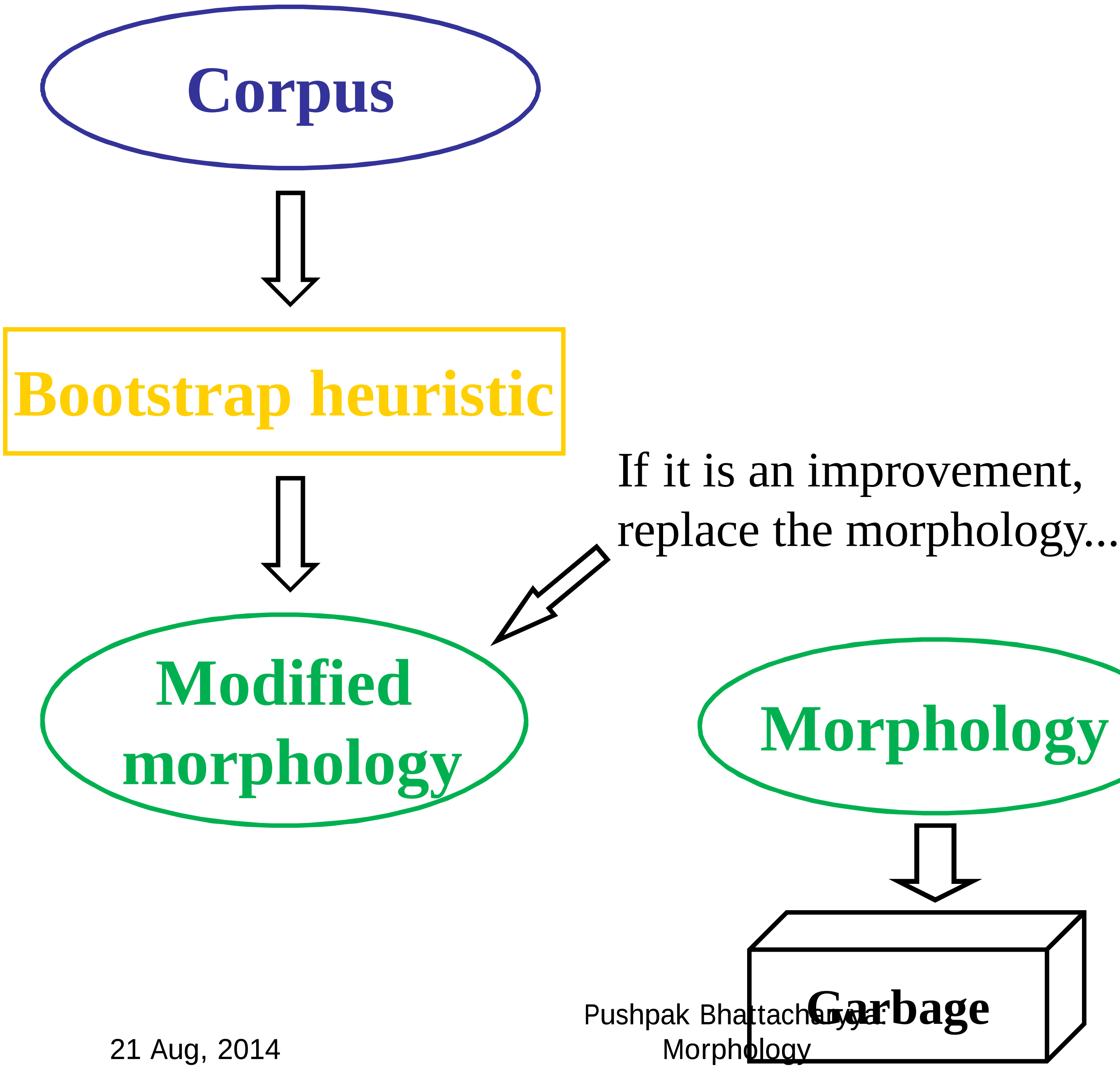
Incremental heuristics

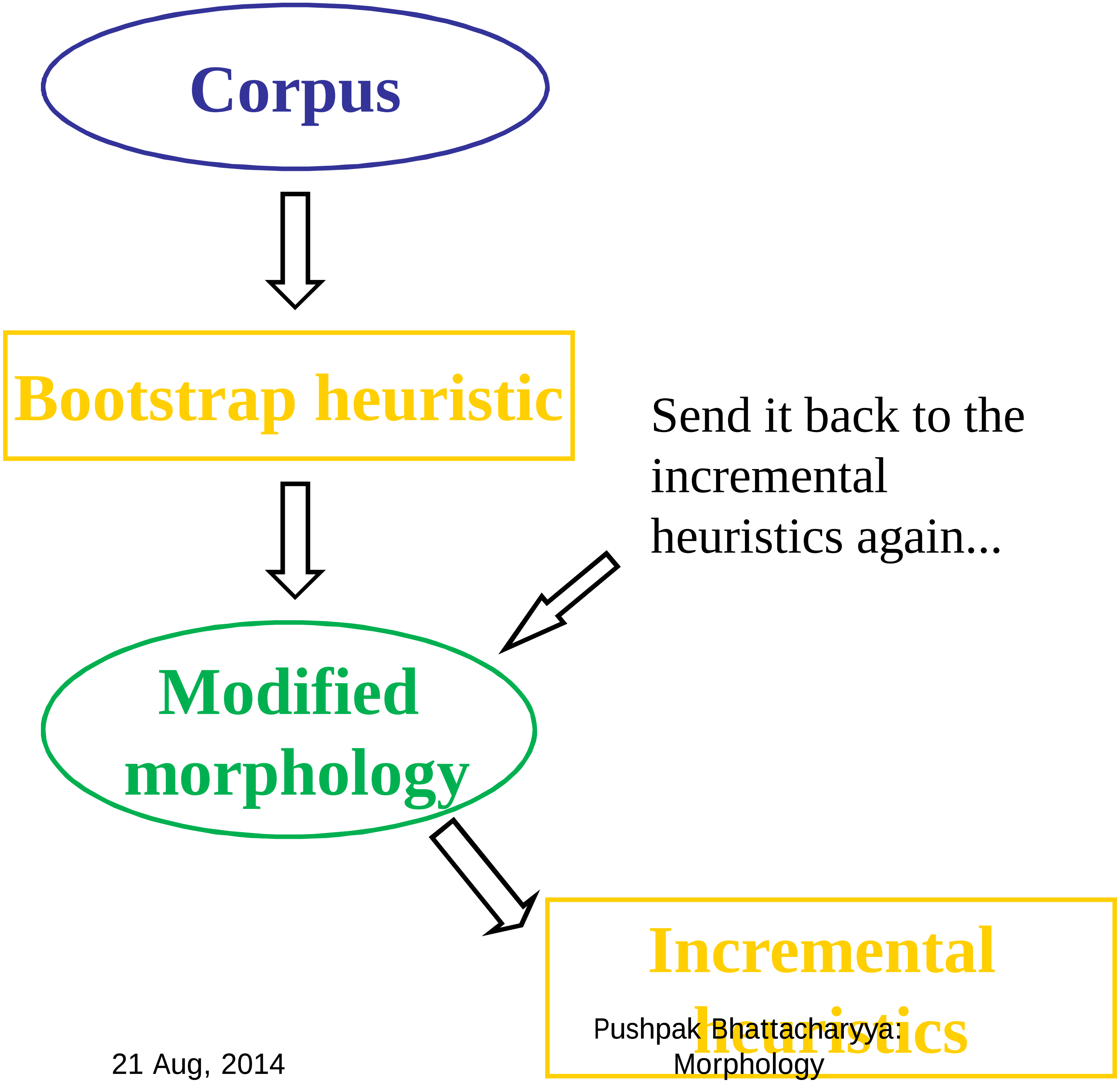
Feed it to the incremental heuristics...



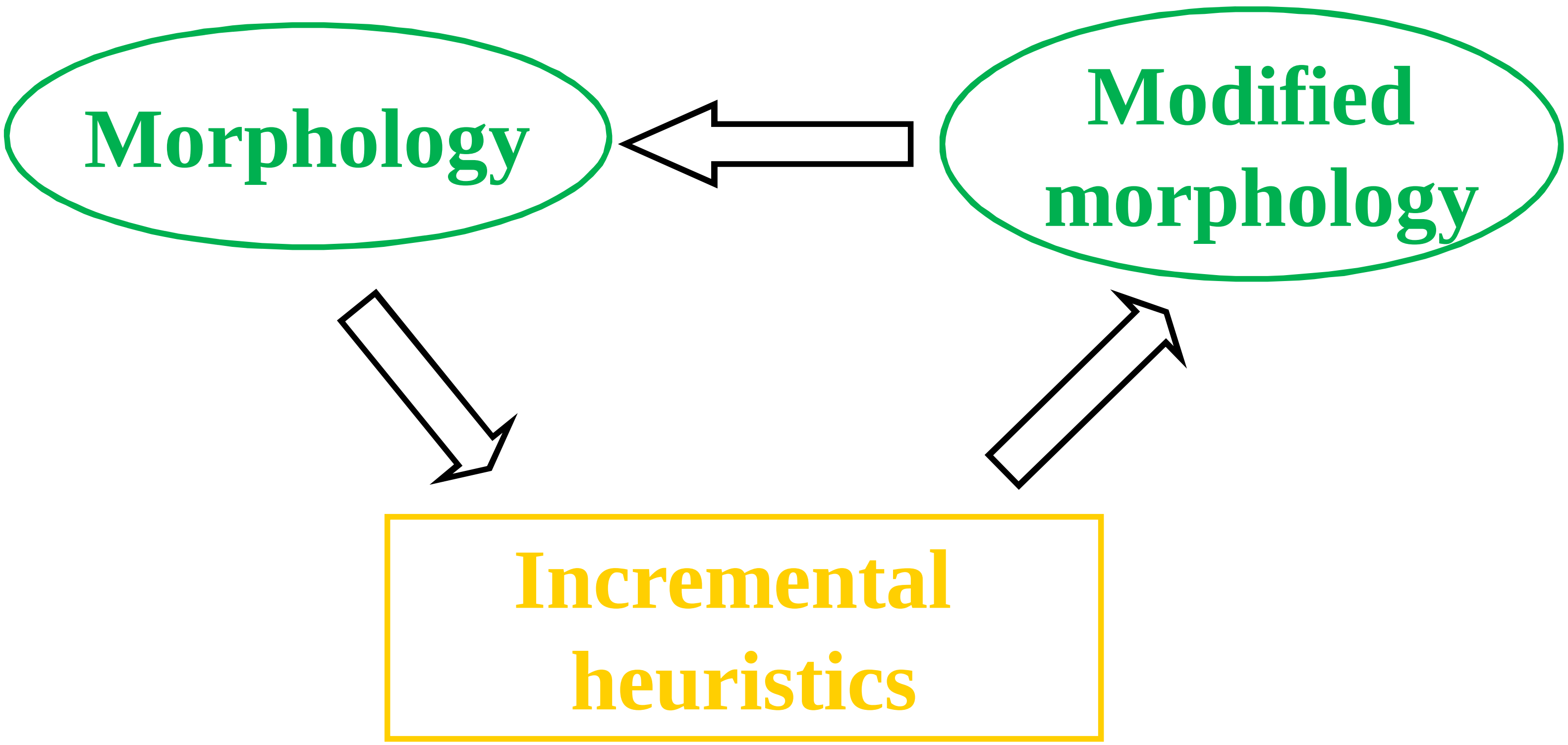
Corpus
Bootstrap heuristic
Morphology
Incremental heuristics
Modified morphology
Out comes a modified morphology.


Corpus
Bootstrap heuristic
Morphology
Incremental heuristics
Modified morphology
Is the modification an improvement? Ask MDL!


Corpus
Bootstrap heuristic
If it is an improvement,
replace the morphology...
Modified
morphology
Morphology
Garbage


Corpus
Bootstrap heuristic
Send it back to the incremental heuristics again...
Modified morphology
Incremental heuristics

Continue until there are no improvements to try.



Assignment- “morphology”

Assignment on “morphology” (1/7)

- Strictly speaking this is not an assignment on morphology, because in morph analysis you have to break apart lemma and suffixes. Still you will get a sense of finite state machine based MA.

Assignment on “morphology” (2/7)

- **Problem statement**
- Auxiliary verbs of English have the following forms:
  - a: Forms of *be (is, am, are, was, were, been)*
  - b: Forms of *have (have, has, had)*
  - c: Forms of *do (do, does, did)*
  - d: Modal auxiliaries *can, could, will, would, shall, should, may, might, must*

Assignment on “morphology” (3/7)

- Phrases like
  - *will have gone,*
  - *could be going,*
  - *might have been found*
- etc. are called verb groups (VG) which have a sequence of auxiliaries followed by a main verb at the end.

Assignment on “morphology” (4/7)

- Give a grammar for VG *(S, V, T, P)*.
  - The grammar should be such that trees with proper depth are found for the strings, *i.e.*, not shallow, flat trees.
  - Assume particles like *not* and *also* are present.
  - Be careful to accept ALL and ONLY the valid strings.

Assignment on “morphology” (5/7)

- Experiment on
  - whether top down or
  - bottom up or
  - combined top down bottom

- approach will be the best for parsing of VG.

Assignment on “morphology” (6/7)

- Convert your grammar to Chomsky Normal Form (CNF) and

- run CYK algorithm on the string:
  - *could also not have been going*

Assignment on "morphology" (7/7)

- The above problem, though given for English, is universal across languages.
- The place of auxiliaries can be taken by suffixes (as in Marathi and Dravidian languages and other agglutinative languages like Turkish, Arabic and Hungarian).
- The order in which such entities combine to form a group or a word form is a matter of parsing.

References

- Cormen, Thomas H. and Stein, Clifford and Rivest, Ronald L. and Leiserson, Charles E. 2001. Introduction to Algorithms, 2nd Edition, ISBN:0070131511, McGraw-Hill Higher Education.
- Creutz Mathis, and Krista Lagus. 2005. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0., Technical Report A81, Publications in Computer and Information Science, Helsinki University of Technology.

References

- Dabre Raj,Amberkar Archana and Bhattacharyya Pushpak 2012. Morphology Analyser for Affix Stacking Languages: a case study in Marathi, COLING 2012, Mumbai, India, 10-14 Dec, 2012.
- El-Shishtawy Tarek and El-Ghannam Fatma 2012. An Accurate Arabic Root-Based Lemmatizer for Information Retrieval Purposes, IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 3, January 2012 ISSN (Online): 1694-0814.
- Goldsmith John A. 2001. Unsupervised Learning of the morphology of a Natural Language, Computational Linguistics, 27(2): 153-198.

References

- Lauri Karttunen 1983. KIMMO: A General Morphological Processor , Texas Linguistic Forum, 22 (1983), 163-186.
- Lovins, J.B. 1968. Development of a stemming algorithm, Mechanical Translations and Computational Linguistics Vol.11 Nos 1 and 2, pp. 22-31.
- Majumder Prasenjit , Mitra Mandar, Parui Swapan K., Kole Gobinda, Mitra Pabitra, and Datta Kalyankumar. 2007. YASS: Yet another suffix stripper, Association for Computing Machinery Transactions on Information Systems, 25(4):18-38.
- Majumder, Prasenjit and Mitra, Mandar and Datta, Kalyankumar 2007. Statistical vs Rule-Based Stemming for Monolingual French Retrieval, Evaluation of Multilingual and Multi-modal Information Retrieval, Lecture Notes in Computer Science vol. 4370, ISBN 978-3-540-74998-1, Springer, Berlin, Heidelberg.

References

- Ozturkmenoglu Okan and Alpkocak Adil 2012. Comparison of different lemmatization approaches for information retrieval on Turkish text collection , Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on.
- Porter M.F. 2006. Stemming algorithms for various European languages, Available at [URL] http://snowball.tartarus.org/texts/stemmersoverview.html As seen on May 16, 2013.
- Ramanathan Ananthakrishnan, and Durgesh D. Rao, 2003. A Lightweight Stemmer for Hindi. , Workshop on Computational Linguistics for South-Asian Languages, EACL
- Snigdha Paul, Nisheeth Joshi and Iti Mathur 2013.
- Development of a Hindi Lemmatizer, CoRR, DBLP:journals/corr/abs/1305.6211 2013

URLS

http://www.cse.iitb.ac.in/~pb

http://www.cfilt.iitb.ac.in