CSE528 Natural Language Processing Venue: ADB-405 SLOTS: A2+TA2 Topic: Text Processing

Prof. Tulasi Prasad Sariki, SCSE, VIT Chennai Campus www.learnersdesk.weebly.com

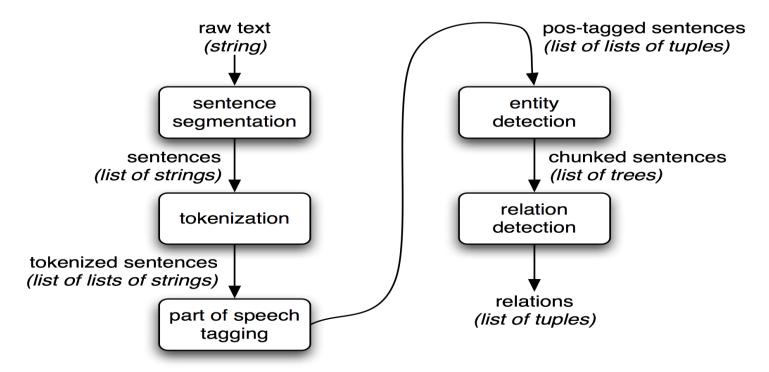


Contents

- Text Processing
- Text Preprocessing
- Challenges in Text Preprocessing
- Types of Writing Systems

Text Processing

In the linguistic analysis of a digital natural language text, it is necessary to clearly define the characters, words, and sentences in any document.



TEXT PROCESSING

Text Preprocessing

The task of converting a raw text file, essentially a sequence of digital bits, into a well-defined sequence of linguistically meaningful units:

- at the lowest level characters representing the individual graphemes in a language's written system,
- Words consisting of one or more characters,
- sentences consisting of one or more words

Text preprocessing is an essential part of any NLP system, since the characters, words, and sentences identified at this stage are the fundamental units passed to all further processing stages.

Text / Word segmentation

Text segmentation is the process of converting a well-defined text corpus into its component words and sentences.

This is very important task to work on morphology and syntax levels of NLP.

Word segmentation breaks up the sequence of characters in a text by locating the word boundaries, the points where one word ends and another begins.

For computational linguistics purposes, the words thus identified are frequently referred to as tokens, and word segmentation is also known as tokenization.

Sentence Segmentation / Text Normalization

Sentence segmentation is the process of identifying sentence boundaries between words in different sentences.

Since most written languages have punctuation marks that occur at sentence boundaries, sentence segmentation is frequently referred to as sentence boundary detection, sentence boundary disambiguation

Text normalization is a related step that involves merging different written forms of a token into a canonical normalized form; for example, a document may contain the equivalent tokens "Mr.", "Mr", "mister", and "Mister" that would all be normalized to a single form.

Challenges of Text Preprocessing

The type of writing system (SCRIPT) used for a language is the most important factor for determining the best approach to text preprocessing.

It needs:

- at least one set of defined base elements or symbols, individually termed characters and collectively called a script;
- at least one set of rules and conventions (orthography) understood and shared by a community, which arbitrarily assigns meaning to the base elements (graphemes), their ordering and relations to one another;
- at least one language (generally spoken) whose constructions are represented and able to be recalled by the interpretation of these elements and rules

Classification of Systems

Classification by Daniels

Туре	Each symbol represents	Example
Logographic	morpheme	Chinese characters
Syllabic	syllable or mora	Japanese kana
Alphabetic	phoneme (consonant or vowel)	Latin alphabet
Abugida	phoneme (consonant+vowel)	Indian Devanāgarī
Abjad	phoneme (consonant)	Arabic alphabet
Featural	phonetic feature	Korean hangul

Logographic writing systems

In a logographic writing system, in theory, each symbol (word or morpheme) represents one idea example: Chinese



Logophonetic Writing Systems

Definition: there are two major types of signs, ones denoting morphemes and ones denoting sounds. (ex) Egyptian, Japanese and sumerian

3	9	0	S	С
ka [ka]	kha [kʰa]	ga [ga]	gha [ga]	na [ŋa]
Ø	\mathfrak{D}	Q	ဈ	ည
ca [sa]	cha [s ^s a]	ja [za]	jha [za]	ña (ña)
E	G	2	ບ	ന
ța [ta]	tha [t ^h a]	da [da]	dha [da]	ņa [na]
0	∞	З	0	န
ta [ta]	tha [tʰa]	da [da]	dha [da]	na [na]
O	Q	ଓ	ဘ	6
pa [pa]	pha [pʰa]	ba [ba]	bha [ba]	ma (ma)
ω	ବ	0	0	သ
ya [ya]	ra [ya]	la [la]	wa [wa]	<mark>sa</mark> [Өа]
ဟ	P.	39		
ha [ha]	la [la]	a [a]		

Abugida

South Asian scripts such as Brahmi and its descendants fit into both syllabary and alphabet.

It is syllabic because the basic sign contains a consonant and a vowel.

Greek had CV, CVC, CCVC, CVCC syllable structures, so they invent a way to cut down syllables to consonant and vowels

ព្រះរាជាណាចក្រកម្ពុជា

Alphabetic

A system of consonant and vowel symbols that, either individually or in combinations, represent the speech sounds of a written language (ex) English

ABCDEFGHIJKL MNOPDRSTUVW XYZÁÁÉÍŐŐ Üabcdefg hijklmnopqrstuvwxyzàåéîõøü & 1234567890(S£.,!?)

TEXT PROCESSING

Abjad or Consonantal Alphabet

alphabetic writing systems in which only the consonants in words are written, and the vowels are left out (ex) Hebrew, Arabic



TEXT PROCESSING

Character representation

How Characters in languages can be represented?

At its lowest level, a computer-based text or document is merely a sequence of digital bits in a file.

The first essential task is to interpret these bits as characters of a writing system of a natural language.

Unicode

Fundamentally, computers just deal with numbers. They store letters and other characters by assigning a number for each one.

Before Unicode was invented, there were hundreds of different encoding systems for assigning these numbers.

Even for a single language like English no single encoding was adequate for all the letters, punctuation, and technical symbols in common use.

Unicode

These encoding systems also conflict with one another. That is, two encodings can use the same number for two *different* characters, or use different numbers for the *same* character.

Unicode covers all the characters for all the writing systems of the world, modern and ancient. It also includes technical symbols, punctuations, and many other characters used in writing text. The Unicode Standard is intended to support the needs of all types of users, whether in business or academia, using mainstream or minority scripts.

Types of Encoding

Two Types Encoding

Character Encoding

• ASCII, ISCII, Unicode

Font Encoding

• Eenadu, vaartha, Kumudam, Daily Thanthi

ASCII Features

American Standard Code for Information Interchange

7-bit code

8th bit is unused (or used for a parity bit)

2⁷ = 128 codes

Two general types of codes:

- 95 are "Graphic" codes (displayable on a console)
- 33 are "Control" codes (control features of the console or communications channel)

$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	Dec HxOct Char	Dec Hx Oct Html Chr	Dec Hx Oct Html Chr Dec Hx Oct Html Chr
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	0 0 000 <mark>NUL</mark> (null)	32 20 040 Space	64 40 100 «#64; 🕑 96 60 140 «#96; `
33003ETX(end of text)3523043e435;6743103e477;C9963143e4910;d44004EOT(enquiry)3725045e436;6844104e676;D10064144e4100;d55005EN0((enquiry)3725045e433;c7046105e670;F10266146e4102;f66006ACK(ackmowledge)3927047e439;717177767e7147;10366146e4102;f770BEL(bell)3927047e4312;7147137e616e470;F10266146e4103;f99011 <tab< td="">(borizontal tab)4028050e441;734911e677;K10664152e4106;f11B013VT(vertical tab)4328053e442;7744113e677;K10666153e4106;f12C014FF(NFform feed, new page)4328053e442;7744113e677;K10666153e4106;11056645144;116ff<td< td=""><td>l l OOl <mark>SOH</mark> (start of heading)</td><td>33 21 041 ! !</td><td>65 41 101 A A 97 61 141 a a</td></td<></tab<>	l l OOl <mark>SOH</mark> (start of heading)	33 21 041 ! !	65 41 101 A A 97 61 141 a a
44004 EOT(end of transmission)3624044 α #36; 66844104 α #68; D10064144 α #100; d55005 ENQ(enquiry)3725045 α #37; %6945105 α #69; E10165145 α #101; e6600 ACK(acknowledge)3826046 α #38; a7046106 α #70; F10266146 α #102; f77007 BEL(bell)3927047 α #39; '7147107 α #71; G10367147 α #105; f99011 TAB(horizontal tab)4129051 α #41;)7349111 α #73; I10664152 α #105; f11B013 VT(vertical tab)422A052 α #42; f7444112 α #77; M10666153 α #105; f12C014 FF(NF form feed, new page)432B053 α #42; f7546114 α #77; J10660155 α #105; f13D015 CR(carriage return)452D055 α #46; f7740115 α #77; M1096D155 α #105; f14E016020DLe(dati ta kescape)4630060 α #47; f794F116 α #77; M1096D155 α #111; o	2 2 002 STX (start of text)	34 22 042 " "	66 42 102 «#66; B 98 62 142 «#98; b
55005EMQ (enquiry)3725045#37; %694510566/9; E10165145 $a\#101; e$ 66006ACK (acknowledge)3826046 $a#39; t$ 7046106 $a#70; F$ 10266146 $a#102; f$ 7707BEL (bell)3927047 $a#39; t$ 7147107 $a#71; F$ 10367147 $a#103; G$ 86010BS(backspace)4028050 $a#40; t$ 7248110 $a#72; H$ 10668150 $a#104; h$ 99011TAB (horizontal tab)4129051 $a#44; t$ 7349114 $a#73; K$ 10666152 $a#106; J$ 11B013VT(vertical tab)4220052 $a#44; t$ 7444112 $a#74; J$ 10666152 $a#106; J$ 12C014Ff(MP form feed, new page)4322055 $a#45; -$ 7740115 $a#74; J$ 10666150 $a#106; J$ 13D015CR(carriage return)4520055 $a#45; -$ 7740115 $a#10; P$ $a#10; P$ 14E012S0(shift out)472F057 $a#47; J$ 1066714 $a#76; L$ 106 $a#12; P$ 17102DC1(3 3 003 ETX (end of text)	35 23 043 # #	67 43 103 «#67; C 99 63 143 «#99; C
660.06ACK(acknowledge)38260.46 $4/38$; 470461.06 $e^{3}/7$; F1.02661.46 $e^{4}/102$; f770.07BEL(bel1)39270.47 $e^{3}/39$; '7147107 $e^{7}/1$; C1.03671.47 $e^{4}/103$; g880.10BS(backspace)40280.50 $e^{4}/40$; (72481.06691.51 $e^{4}/103$; g990.11TAB(borizontal tab)41290.51 $e^{4}/41$;)7349111 $e^{7}/72$; H106641.52 $e^{4}/105$; 110A0.12LF(NL line feed, new line)42200.52 $e^{4}/44$; +764C1.14 $e^{7}/7$; K1076B1.53 $e^{4}/107$; K12C0.14FF(NP form feed, new page)422D0.55 $e^{4}/44$; -764C1.14 $e^{7}/7$; M1096D1.54 $e^{4}/107$; R14E0.1650(shift out)452D0.55 $e^{4}/45$; -764C1.14 $e^{7}/7$; M1096D1.54 $e^{4}/107$; N15F0.17ST(shift out)472F0.57 $e^{4}/7$; /794F1.15 $e^{4}/102$; N1.106D $e^{4}/102$; N1610.20DLC(device control 1)5032 <td>4 4 004 EOT (end of transmission)</td> <td>36 24 044 \$ \$</td> <td>68 44 104 «#68; D 100 64 144 «#100; d</td>	4 4 004 EOT (end of transmission)	36 24 044 \$ \$	68 44 104 «#68; D 100 64 144 «#100; d
77007BEL (bell)3927047 $\#39$; '7147107 $\#71$; C10367147 $\#103$; G99011TAB (horizontal tab)4026050 $\#40$; (7248110 $\#73$; I10569151 $\#103$; G10A012LF (NL line feed, new line)422A052 $\#412$; '7444112 $\#74$; J1066A152 $\#105$; i11B013VT (vertical tab)422A052 $\#42$; '7444112 $\#74$; J1066A152 $\#105$; i12C014Ff (NF form feed, new page)422D055 $\#442$; '754B113 $\#77$; M1096D155 $\#107$; K13D015CR (carriage return)452D055 $\#452$; -774D115 $\#77$; M1096D155 $\#107$; M14F016S0 $\#1ft$ in)472F057 $\#472$; /794F117471166152 $\#107$; M1610020DLF(data link escape)4830060 $\#482$; 08050120 $\#832$; 71166 $\#112$; 70166 $\#112$; 701610020DLF(data link escape)4830060 $\#492$; 18151121 $\#832$; 282122<	5 5 005 <mark>ENQ</mark> (enquiry)		
88010BS(backspace)4028050 $4\#40$; (7248110 $4\#72$; H10468150 $4\#104$; h99011TAB(horizontal tab)4129051 $4\#412$; *7349111 $6\#73$; I10664152 $4\#105$; i11B013VT(vertical tab)432B053 $4\#43$; +754B113 $6\#75$; K1076B153 $6\#107$; K12C014FF(WF form feed, new page)442C054 $6\#445$; -764C114 $4\#76$; L1086C154 $6\#107$; K13D015CR(carriage return)452D055 $6\#45$; -764E116 $6\#73$; II1106E156 $6\#107$; M14E01630(shift in)472F057 $4\#47$; /794F1116F157 $6\#112$; p1711021DC1(device control 1)4931061 $6\#457$; 28050120 $6\#367$; 911171161 $6\#112$; P1612022DC2(device control 2)5032062 $4\#52$; 4845111472162 $4\#112$; P1711021DC1(device control 4)5234064 $4\#52$; 48454124 $6\#37$; 511472 </td <td></td> <td></td> <td></td>			
99011TAB (horizontal tab)4129051 $#441; i$ 7349111 $#73; I$ 10569151 $#4105; i$ 10A012LF(NL line feed, new line)422A052 $#42; i$ 7444112 $#74; J$ 1066A152 $#4105; i$ 11B013VT(vertical tab)432B053 $#442; i$ 7444112 $#74; J$ 1066A152 $#4105; i$ 12C014FF(NP form feed, new page)442C054 $#442; i$ 754B113 $#75; K$ 1066A152 $#4105; i$ 13D015CR(carriage return)452D055 $#442; i$ 774D115 $#77; M$ 1096D155 $#109; n$ 14E016S0(shift out)472F057 $#447; i$ 794F117 $#979; 0$ 1116F157 $#111; o$ 15F017SI(shift sin)472F057 $#47; i$ 794F117 $#979; 0$ 1116F157 $#111; o$ 1610020DCE(data link escape)4830060 $#48; 0$ 8050120 $#60; F$ 111614112; q1812022DC2(device control 1)5331063 $#51; 3$ 8353123 $#63; S$			
10A 012LF(NL line feed, hew line)422A 052 $\%442$; *744A 112 $\%74; J$ 1066A 152 $\%106; j$ 11B 013VT(vertical tah)432B 053 $\%43; +$ 754B 113 $\%75; K$ 1076B 153 $\%107; k$ 12C 014FF(NP form feed, new page)442C 054 $\%44; .$ 764C 114 $\%75; K$ 1076B 153 $\%107; k$ 13D 015CR(carriage return)452D 055 $\%445; -$ 774D 115 $\%77; m$ 1096D 155 $\%109; m$ 14E 016S0(shift out)462E 056 $\%445; -$ 784E 116 $\%77; m$ 1016E 156 $\%110; n$ 15F 017SI(shift in)472F 057 $\%47; /$ 794F 117 $\%79; 0$ 1116F 157 $\%111; n$ 1610020DLE(data link escape)4330060 $\#48; 0$ 8050120 $\#80; F$ 1127066 $\#112; p$ 1711021DC1(device control 2)5032062 $\#50; 2$ 8252122 $\#83; 5$ 11573163 $\#112; 2$ 1913023DC2MAK(negative acknowledge)5335358551125 $\#83; 5$ 11573163 $\#114; 2$ 2115026SYN(synchronous idle)543666 $\#$	· • ·	•	
11B013VT(vertical tab)432B053 $\#43$; +754B113 $\#75$; K1076B153 $\#\#107$; K12C014FF(NPform feed, new page)442C054 $\#44$; ,764C114 $\#75$; K1076B153 $\#\#107$; K13D015CR(carriage return)452D055 $\#445$; ,774C114 $\#77$; M1096D155 $\#109$; M14E016SO(shift out)462E056 $\#467$; ,784E116 $\#77$; M1096D155 $\#109$; M15F017SI(shift in)472F057 $\#477$; /794F117 $\#797$; O1116F157 $\#111$; O1610020DLE(data link escape)4830060 $\#482$; O8050120 $\#803$; S1116F157 $\#111$; O1711021DC1(device control 1)4931061 $\#492$; 18151121 $\#813$; C11371163 $\#112$; P1913023DC3(device control 4)5133063 $\#53$; S8353123 $\#833$; S11573163 $\#112$; P2114024DC4(device control 4)5234064 $\#52$; 48454<			
12C014FF(NP form feed, new page)442C054 $a#44:$ 764C114 $a#76:$ 11086C154 $a#108:$ 113D015CR(carriage return)452D055 $a#45:$ 774D115 $a#77:$ M1096D155 $a#109:$ m14E016SO(shift out)462E056 $a#45:$ 784E116 $a#77:$ M1096D155 $a#102:$ n15F017SI(shift in)472F057 $a#47:$ 794F117 $a#79:$ 01116F157 $a#111:$ 01610020DLE(data link escape)4830060 $a#49:$ 18151121 $a#80:$ P11270160 $a#112:$ P1711021DC1(device control 1)4931061 $a#49:$ 181511371161 $a#113:$ q1812022DC2(device control 3)5133063 $a#51:$ 3335123 $a#83:$ 5153163 $a#114:$ 72162 $a#114:$ 72162 $a#114:$ 12115025NAK(negative acknowledge)5335065455:3555555555165666666 <td>· · · · ·</td> <td></td> <td></td>	· · · · ·		
13D015CR(carriage return)452D055 $\$445$; -774D115 $\$77$; M1096D155 $\$#109$; M14E016SO(shift out)462E056 $\$446$; .784E116 $\$77$; N1096D155 $\$#109$; M15F017SI(shift in)472F057 $\$477$; /794F117 $\$79$; O1116F157 $\$#112$; O1711021DC1(device control 1)4931061 $\$49$; 18151120 $\$805$; P11270160 $\$#112$; P1812022DC2(device control 3)5032062 $\$855$; 28252122 $\$83$; 5311573163 $\$#113$; P1913023DC3(device control 4)5133063 $\$851$; 38353123 $\$83$; 5311573163 $\$#115$; 92014024DC4(device control 4)5335065 $\$85$; 55125 $\$85$; 55125 $\$85$; 5711674164 $\$8116$; 711775165 $\$8117$; U2115025NAK(nagetive acknowledge)5335065 $\$85$; 55125 $\$85$; 55125 $\$85$; 5711776166 $\$8117$; U2216026SYN(synchronous id	• •		
14E016S0(shift out)462E056 $#46$;784E116 $#78$; N1106E156 $##110$; N15F017SI(shift in)472F057 $#47$;794F117 $#79$; O1116F157 $##111$; O1610020DLE(data link escape)4830060 $#443$; O8050120 $#803$; P1116F157 $##111$; O1710DC1(device control 1)4931061 $#449$; 18151121 $#863$; S11371161 $##113$; Q1812022DC2(device control 3)5133063 $#51$; 38353123 $#863$; S11573163 $##115$; S2014024DC4(device control 4)5234064 $#52$; 484545456311674164 $##116$; t2115025NAK(negative acknowledge)5335065 $#53$; 5855511775165 $##117$; u2216026SYN(synchronous idle)5436066 $#55$; 78757127 $#86$; W11977167 $##120$; X2418030CAN(cancel)5537067 $#55$; 7875712778166 $##117$; u			
15F017SI(shift in)472F057 $\ast #47$;794F117 $\ast #79$;01116F157 $\ast #111$;01610020DLE(data link escape)4830060 $\ast #48$;08050120 $\ast #80$;P11270160 $\ast #112$;p1711021DC1(device control 1)4931061 $\ast #49$;18151121 $\ast #81$;011371161 $\ast #113$;q1812022DC2(device control 2)5032062 $\ast #50$;28252122 $\ast #82$;R11472163 $\ast #115$;s2014024DC4(device control 4)5234064 $\ast #52$;48454124 $\ast #84$;T11674164 $\star #116$;t2115025NAK(negative acknowledge)5335065 $\ast #53$;58555125 $\ast #85$;U11775165 $\ast #117$;u2216026SIN(cancel)5537067 $\ast #55$;78757127 $\ast #67$;U11977167 $\star #119$;v2416ofmedium)5739071 $\star #55$;78757127 $\ast #67$;U11977<			
1610020DLE(data link escape)4830060 $\&$ #48:08050120 $\&$ #80; P11270160 $\&$ #112; P1711021DC1(device control 1)4931061 $\&$ #49:18151121 $\&$ #81; Q11371161 $\&$ #113; Q1812022DC2(device control 2)5032062 $\&$ #50; 28252122 $\&$ #82; R11472162 $\&$ #114; F1913023DC3(device control 3)5133063 $\&$ #51; 38353123 $\&$ #83; S11573163 $\&$ #115; S2014024DC4(device control 4)5234064 $\&$ #52; 48454124 $\&$ #84; F11674164 $\&$ #116; t2115025NAK(negative acknowledge)5335065 $\&$ #53; 58555126 $\&$ #86; U11775166 $\&$ #117; u2216026SYN(synchronous idle)5537067 $\&$ #55; 78757127 $\&$ #87; U11977166 $\&$ #118; V2317027ETB(end of medium)5537067 $\&$ #55; 78757127 4 #87; U11977167 $\&$ #120; X2418030CAN(cancel)5638070 $&$ #56; 8 <td></td> <td></td> <td></td>			
17 11 021 DC1 (device control 1) 49 31 061 \$#49; 1 81 51 121 \$#81; 0 113 71 161 \$#113; q 18 12 022 DC2 (device control 2) 50 32 062 \$#50; 2 82 52 122 \$#82; R 114 72 162 \$#114; r 19 13 023 DC3 (device control 3) 51 33 063 \$#51; 3 83 53 123 \$#83; S 115 73 163 \$#115; s 20 14 024 DC4 (device control 4) 52 34 064 \$#52; 4 84 54 124 \$#84; T 116 74 164 \$#116; t 21 15 025 NAK (negative acknowledge) 53 35 065 \$#53; 5 85 55 125 \$#86; U 118 76 166 \$#116; t 21 15 025 NAK (negative acknowledge) 55 37 067 \$#55; 7 87 <td>· · · · ·</td> <td></td> <td></td>	· · · · ·		
18 12 022 DC2 (device control 2) 50 32 062 \$			
19 13 023 DC3 (device control 3) 51 33 063 \$			
20 14 024 DC4 (device control 4) 52 34 064 4 4 84 54 124 T T 116 74 164 t t 21 15 025 NAK (negative acknowledge) 53 35 065 5 5 85 55 125 U U 117 75 165 u U 22 16 026 SYN (synchronous idle) 54 36 066 6 6 86 56 126 V V 118 76 166 v V 23 17 027 ETB (end of trans. block) 55 37 067 7 7 87 57 127 V X 120 78 170 p Y 24 18 030 CAN (cancel) 57 39 071 9 9 89 59 131 Y Y 121 79 171 y Y 26 1A 032 SUB (substitute) 58 3A 072 8 E 91 58 <td></td> <td></td> <td></td>			
21 15 025 NAK (negative acknowledge) 53 35 065 \$\$\$3;5 85 55 125 \$			
22 16 026 SYN (synchronous idle)54 36 066 $\&$ #54;686 56 126 $\&$ #86; V118 76 166 $\&$ #118; V23 17 027 ETB (end of trans. block)55 37 067 $\&$ #55;787 57 127 $\&$ #87; W119 77 167 $\&$ #119; W24 18 030 CAN (cancel)56 38 070 $\&$ #56;888 58 130 $\&$ #88; X120 78 170 $\&$ #120; X25 19 031 EM (end of medium)57 39 071 $\&$ #57;989 59 131 $\&$ #89; Y121 79 171 $\&$ #121; Y26 1A 032 SUB (substitute)58 3A 072 $\&$ #58;:90 5A 132 $\&$ #90; Z122 7A 172 $\&$ #122; Z27 1B 033 ESC (escape)59 3B 073 $\&$ #59;;91 5B 133 $\&$ #91; [123 7B 173 $\&$ #123; {28 1C 034 FS (file separator)60 3C 074 $\&$ #60; <			
2317027ETB(end of trans. block)5537067 $\&$ #55; 78757127 $\&$ #87; W11977167 $\&$ #119; W2418030CAN(cancel)5638070 $\&$ #56; 88858130 $\&$ #86; X12078170 $\&$ #120; X2519031EM(end of medium)5739071 $\&$ #57; 98959131 $\&$ #89; Y12179171 $\&$ #121; Y261A032SUB(substitute)583A072 $\&$ #58; :905A132 $\&$ #90; Z1227A172 $\&$ #122; Z271B033ESC(escape)593B073 $\&$ #59; ;915B133 $\&$ #91; [1237B173 $\&$ #123; {281C034FS(file separator)603C074 $\&$ #60; <			
24 18 030 CAN (cancel) 56 38 070 & #56; 8 88 58 130 & #88; X 120 78 170 & #120; X 25 19 031 EM (end of medium) 57 39 071 & #57; 9 89 59 131 & #89; Y 121 79 171 & #121; Y 26 1A 032 SUB (substitute) 58 3A 072 & #58;: 90 5A 132 & #90; Z 122 7A 172 & #122; Z 27 1B 033 ESC (escape) 59 3B 073 & #59;; 91 5B 133 & #91; [123 7B 173 & #123; (28 1C 034 FS (file separator) 60 3C 074 & #60; <			
25 19 031 EM (end of medium) 57 39 071 \$			
26 1A 032 SUB (substitute) 58 3A 072 \$4\fmush85;: 90 5A 132 \$4\fmush90;: 122 7A 172 \$4\fmush122;: z 27 1B 033 ESC (escape) 59 3B 073 \$4\fmush59;: 91 5B 133 \$4\fmush91:: 123 7B 173 \$4\fmush123:: { 28 1C 034 FS (file separator) 60 3C 074 \$4\fmush60::<			
27 1B 033 ESC (escape) 59 3B 073 \$			
28 1C 034 FS (file separator) 60 3C 074 \$#60; <			
29 1D 035 G\$ (group separator) 61 3D 075 = = 93 5D 135]] 125 7D 175 } } 30 1E 036 R\$ (record separator) 62 3E 076 > > 94 5E 136 ^ ^ 126 7E 176 ~ ~	· • •		
30 1E 036 RS (record separator) 62 3E 076 & #62; > 94 5E 136 & #94; ^ 126 7E 176 & #126; ~	· ± ·		
31 1F 037 US (unit separator) 63 3F 077 ? 2 95 5F 137 _ 127 7F 177 DEL			
	31 1F 037 <mark>US</mark> (unit separator)	63 3F 077 ? <mark>?</mark>	95 5F 137 _ _ 127 7F 177 DEL

Source: www.LookupTables.com

ISCII(Indian Standard Code for Information Interchange)

	Hex	0	1	2	3	4	5	6	7	8	9	Α	в	C	D	E	F
Hex	Dec	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
0	0	NUL	DLE	SP	0	@	Р	4	р				ओ	¢	₹.	ò	EXT
1	1	SOH	DC1	1	1	A	Q	a	q			ð	ओ	म	ल	6	a
2	2	STX	DC2		2	В	R	b	r.			ó	अगे	त	ळ	ð	3
3	3	ETX	DC3	#	3	С	s	c	8			OT	赛	4	क	ŏ	ş
4	4	EOT	DC4	\$	4	D	т	d	τ.			37	य	द	व	ो	3
5	5	ENQ	NAK	%	5	E	U	e	u			आ	π	-87	8	ो	8
6	6	ACK	SYN	&	6	F	v	f	v.			÷.	घ	ৰ	ष	ो	4
7	7	BEL	ETB		7	G	w	g	w			÷	8	ज	स	đ	ą.
8	8	BS	CAN	0	8	н	х	h	x			3	Π.	Ψ.	8	2	0
9	9	HT	EM)	9	1	Y	i	У			क	5	क	INV	0	6
A	10	LF	SUB	+	:	J	Z	j	z			ऋ	স	য	or	1	2
в	11	VT	ESC	+	:	K	[k	1			ě	17	ч.	ю		
C	12	FF	FS		<	L	١.	1	1			ए	স	म	fo		
D	13	CR	GS		- 1	М]	m	3			ĝ	5	य	9		
Е	14	SO	RS		>	N	A	n	~			ų,	3	7	Q.		
F	15	SI	US	1	2	0	_	0	DEL			ओ	\$	₹	q	ATR	

It is a coding scheme for representing various writing systems of India. It encodes the main Indic scripts and a Roman transliteration.

The supported scripts are: Assamese, Bengali (Bengla), Devanagari, Gujarati, Gurmukhi, Kannada, Malayalam, Oriya, Tamil, and Telugu.

One motivation for the use of a single encoding is the idea that it will allow easy transliteration from one writing system to another.

Unicode

Unicode is a computing industry standard for the consistent encoding, representation and handling of text expressed in most of the world's writing systems.

The latest version (*Unicode 7.0*) of Unicode contains a collection of more than 110,000 characters covering 100 scripts and various symbols.

Unicode can be implemented by different character encodings. The most commonly used encodings are UTF-8, UTF-16.

http://www.unicodetables.com/

http://www.unicode.org/

Structural Differences with ISCII

Unicode is stateless:

- No shifting to get different scripts
- Each character has a unique number

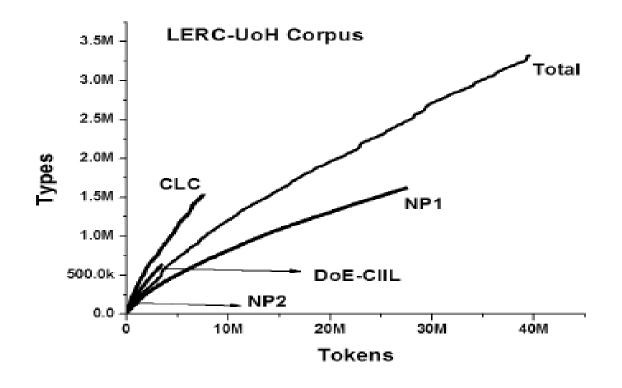
Unicode is uniform:

- No extension bytes necessary
- All characters coded in the same space

Yudit is a <u>free</u> Unicode plain-text editor for Unix-like systems.

http://www.yudit.org/

Type vs Token



Example:

Consider the sentence below A rose is a rose is a rose

There are three word types in the sentence: "rose", "is" & "a".

There are eight word tokens

Telugu

I	0C0	0C1	0C2	0C3	0C4	0C5	0C6	0C7	l							
o		80 0C10	ර °C20	ර °C30	ි 8049		<u>ౠ</u>		A	ರ್ಷ	చ	ప		ී	γ	Ч
1	्र		ک ۵C21	0031	ാ _{0C41}		တ္ <u>ာ</u> ₀₀₆₁			0C0A	0C1A	0C2A		0C4A	0C6A	0C7A
2	ಂ	ی _{0C12}	ک ۵C22	<u></u> 0C32	ూ		್ಲ್ರ		В	2008 0C0B	ည် 0C1B	م 0C2B		് _{0C4B}		Щ
з	ഃ	<u>د</u> ٥с13	6022 0C23	ද 0C33	ാ ₀₀₄₃		្លូ			8	8	బ		ా	۹	
4		0010 0014	من ۵C24		ౄ				C	0000	0010	0C2C		0C4C	0C6C	0C7C
5	9 0005	<u>درام</u> 0015	0024 Č	ی۔ ₀₀₃₅		് _{0C55}			D		ఝ	భ	ح	్	s	ح
6	ສ	ಖ	ద	శ	ි	្រួ	0				0C1D	0C2D	0C3D	0C4D	0C6D	0C7D
7	3000 8	റ്	ос26 Ф	ష	ി		0000		E	ఎ	ವೆ	మ	ి		σ	Ł
8	0C07	0017 ఘ	_{0C27}	ం	م م	าร	0C67	8		0C0E	0C1E	0C2E	0C3E		0C6E	0C7E
9		_{0C18}	0C28	హ	0C48	₀₀₅₈		0C78	F		ట	య	े		٦	୧
<u> </u>	0C09	0C19		0C39		0C59	0C69	0C79		0C0F	0C1F	0C2F	0C3F	///////	0C6F	0C7F

TEXT PROCESSING

Devanagari

	090	091	092	093	094	095	096	097						Ś			
ο	ි	य	ਠ	र	ी	ઙૻ૾ૢ	ૠ	o	Α	ઝ	च	प	ं	া	ग	8	य
	0900	0910	0920	0930	0940	0950	0960	0970		090A	091A	092A	093A	094A	095A	096A	097A
1	े 0901	ऑ	ड ⁰⁹²¹	<u>र</u> ₀₉₃₁	9 %	스 0951	<u>ल्ट</u>	• 0971					+	À	-	1.	
2	்	ऒ	ढ	ल	্য	0	્ર	ॲ	В	ऋ	S 091B	ጥ 0928	() 093B	094B) 095B	Q96B	J 097B
3	0902	आ	ण	0932	₀₉₄₂ ර	0952	0962 Ce 0963	अ		रू	জ	ৰ	\bigcirc	ौ	ड	£	<u>ড</u>
4	ञ	ॵ	त	<u>0933</u>	₀₉₄₃ 	0953 ර	0963	⁰⁹⁷³	С	090C	0910	92C	093C	094C	9. 095C	9 96C	097C
5	अ	0914	₀₉₂₄	व	094 기	⁰⁹⁵⁴ 기	0964	ॵ	D	$\overleftarrow{\mathbf{U}}$	झ	भ	5	\circ	ढ	७	2
6	^{०९०५}	0915 ख	⁰⁹²⁵	₀₉₃₅	₀₉₄₅	0955	0965	⁰⁹⁷⁵	D	090D	091D	092D	093D	094D	095D	096D	097D
Ŭ	0906	0916	0926	0936	0946	0956	0966	0976	-	ਜ	ञ	म	ा	े	फ़	1	<u>ड</u>
7	इ 0907	V 0917	ध 0927	Ч 0937	0947	ु ७९५७	Q 0967	ॷ	E	090E	● (091E	092E	093E	094E	095E	096E	997E
8	फि	घ ⁰⁹¹⁸	न	स	^ 0948	क़	२ 0968			π	ਟ	य	£	শ	य	0	ਹ
9	3	ड ्	न	ह	ॉ		N 0969	<u>ज</u> ़्र	F	90F	C 091F	92F	⊺ ⊃ 093F	094F	9 095F	096F	097F

Font

A *font* file is a binary file that contains glyphs, or "pictures", of symbols representing the building blocks of a displayable character set.

Depending on the language, multiple glyphs can comprise a single character.

Code Table

In basic terms, a *code table* is a two column list that maps a numerical value to a glyph. The most widely used code table is Unicode

Font

Encoding

Encoding values are "stored" from a code table. There are many different encoding types to choose from depending on the application.

UTF-8 UTF-16 (UCS(universal character set)-2) UTF-32 (UCS(universal character set)-4)

Allows us to generate displays of text strings in many different languages by using fonts which contain the glyphs corresponding to their alphabet

The computer system takes each code and displays the glyph associated with it which is displayed on a monitor or printed out.

Font

The glyphs may be viewed as the building blocks for the letter to be displayed where, by placing the glyphs one after another, the required display is generated.

Fonts also incorporate a feature whereby some of the glyphs may be defined to have zero width even though they extend over a horizontal range

Thus when the system places a zero width glyph next to another, the two are superimposed and thus permit more complex shapes to be generated, such as accented letters.

Sentence

A sentence is a group of words that are put together to mean something.

A sentence is the basic unit of language which expresses a complete thought.

It does this by following the grammatical rules of syntax.

Sentence Boundary Disambiguation

- People use . ? and !
- Sometimes ;
- End-of-sentence marks are overloaded.

Sentence Boundary Disambiguation

English employs whitespace between most words and punctuation marks at sentence boundaries, but neither feature is sufficient to segment the text completely and unambiguously.

Tibetan and Vietnamese both explicitly mark syllable boundaries, either through layout or by punctuation, but neither marks word boundaries.

Written Chinese and Japanese have adopted punctuation marks for sentence boundaries, but neither denotes word boundaries.

Period - most ambiguous. Decimals, e-mail addresses, abbreviations, initials in names, honorific titles.

Sentence Boundary Disambiguation

For example:

<u>U.S.</u> Dist. Judge Charles L. Powell denie motions made by defense attorneys Monday in Portland's insurance fraud trial. Of the handful of painters that Austria has produced in the 20th century, only one, Oskar Kokoschka, is widely known in <u>U.S.</u> This state of unawareness may not last much longer.

Sentence boundary detection by humans is tedious, slow, error-prone, and extremely difficult to codify.

Algorithmic syntactic sentence boundary detection is a necessity.

POS tagging and syntax can be done on sentences

Related Work

As of 1997:

"identifying sentences has not received as much attention as it deserves." [Reynar and Ratnaparkhi1997]

"Although sentence boundary disambiguation is essential . . ., it is rarely addressed in the literature and there are few public-domain programs for performing the segmentation task." [Palmer and Hearst1997]

Two approaches

- Rule based approach
- Machine-learning-based approach

Related Work

Rule based

- Regular expressions
 - [Cutting1991]
 - Mark Wasson converted grammar into a finite automata with 1419 states and 18002 transitions.
- Lexical endings of words
 - [Müller1980] uses a large word list.

Machine-learning-based approach

- [Riley1989] uses regression trees.
- [Palmer and Hearst1997] uses decision trees or neural network.

Maximum Entropy Approach

Potential sentence boundaries are identified by scanning the text for sequences of characters separated by whitespace (tokens) containing one of the symbols !, . or ?.

The system that focused on maximizing performance used the following hints, or contextual "templates":

The Prefix, The Suffix

The presence of particular characters in the Prefix or Suffix

Whether the Candidate is an honorific (e.g. *Ms., Dr., Prof.*)

Maximum Entropy Approach

Whether the Candidate is a corporate designator (e.g. Corp., *M.L.A., M.L.C.)*

Features of the word left of the Candidate

Features of the word right of the Candidate

The templates specify only the form of the information. The exact information used by the maximum entropy model for the potential sentence boundary marked by Corp. in Example sentence would be:

- ANLP Corp. chairman Dr. Smith resigned.
- PreviousWordIsCapitalized, Prefix=Corp, Suffix=NULL, PrefixFeature=CorporateDesignator.

Maximum Entropy Approach

For each potential sentence boundary token (., ?, and !), we estimate a joint probability distribution p of the token and its surrounding context, both of which are denoted by c, occurring as an actual sentence boundary.

The distribution is given by:

$$\mathbf{p}(\mathbf{b},\mathbf{c}) = \pi \prod_{j=1}^{k} \alpha_{j}^{f_{j}(b,c)}$$

Where $b \in \{no, yes\}$, where the α_j 's are the unknown parameters of the model, and where each α_j corresponds to a $f_{j'}$ or a *feature*.

Thus the probability of seeing an actual sentence boundary in the context c is given by p(yes, c).

Corpus is a large collection of text covering different domains, styles, territorial and social variants of usage etc.

A *corpus* is a collection of pieces of language that are selected and ordered according to explicit linguistic criteria in order to be used as a sample of the language.

A corpus provides grammarians, lexicographers, and others a better description of a language.

Chomsky's Critique of Corpus-Based Methods

1. Corpora model performance, while linguistics is aimed at the explanation of competence

If you define linguistics that way, linguistic theories will never be able to deal with actual, messy data

2. Natural language is in principle infinite, whereas corpora are finite, so many examples will be missed

Excellent point, which needs to be understood by anyone working with a corpus.

But does that mean corpora are useless?

Introspection is unreliable (prone to performance factors), and pretty useless with small and unrepresentative data.

Insights from a corpus might lead to generalization/induction beyond the corpus- if the corpus is a good sample of the "text population"

3. Ungrammatical examples won't be available in a corpus Depends on the corpus, e.g., spontaneous speech, language learners, etc.

Corpora analysis provide lexical information, morpho-syntactic information, syntactic as well as semantic information.

Variety of Corpus

- Raw corpus
- POS tagged
- Parsed
- Multilingual aligned
- Spoken language
- Semantic tagged

Raw Corpus

The texts are segmented into sentences and paragraphs

- Reuters corpus (180 Million Word)
- CIIL corpus (3 Million words for 10 major Indian languages)

POS Tagged Corpus

• BNC tagged corpus(100 Million CLAWS tagset)

Parsed

Each sentence is annotated with a phrase-structure parse marking the boundaries of sentence, clause, phrase and coordinated word constituents.

- Lancaster Parsed Corpus (British English)
- Susanne parsed corpus

Semantic Corpus

Sense tagged corpus

 The FrameNet lexical database contains around 1,200 semantic *frames*, 13,000 *lexical units* (a pairing of a word with a meaning; polysemous words are represented by several *lexical units*) and over 190,000 example sentences

Multilingual aligned

Identification of the corresponding sentences in multiple languages and align them

- **CRATER:-** Multilingual Aligned Annotated Corpus (English, French, Spanish)
- JRC-Acquis Multilingual Parallel Corpus: collection of parallel texts in the following 22 languages: Bulgarian, Czech, Danish, German, Greek, English, Spanish, Estonian, Finnish etc.
- Parallel corpora are useful for all types of cross-lingual research

Uses of Corpora

Lexicography / terminology

Linguistics / computational linguistics

Dictionaries & grammars (Collins Cobuild) English Dictionary for Advanced Learners; Longman Grammar of Spoken and Written English

Critical Discourse Analysis

- Study texts in social context
- Analyze texts to show underlying ideological meanings and assumptions

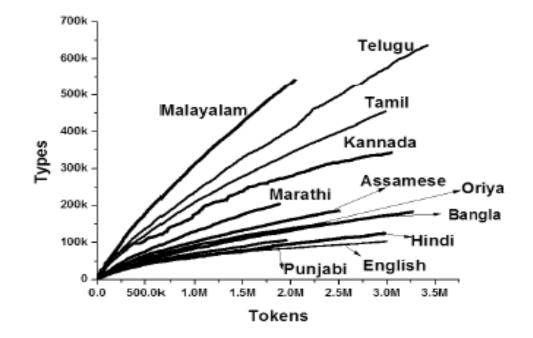
- Analyze texts to show how other meanings and ways of talking could have been used....and therefore the ideological implications of the ways that things were stated

Uses of Corpora

- Literary studies
- Translation practice and theory
- Language teaching / learning
 - ESL Teaching (English as Second Language)
 - LSP Teaching (Language for Specific Purposes)

Type-Token Analysis

Each distinct word form is a type and each occurrence of a type counts as a token.



TEXT PROCESSING

The Telugu corpus developed at the Language Engineering Research Centre (LERC), Department of Computer and Information Sciences, University of Hyderabad, India, hereafter referred to as LERC-UoH corpus, adds up to nearly 39 Million words, perhaps one of the largest corpora for any Indian language today.



TEXT PROCESSING