Chapter NLP:II

II. Text Models

- Text Structure
- Text Preprocessing I
- Text Preprocessing II
- Text Representation
- □ Text Similarity
- □ Sequence Modeling
- □ Language Modeling

The goal of text preprocessing is its conversion into a canonical form.

PRELIMINARY PROOFS.

Unpublished Work ©2008 by Pearson Education, Inc. To be published by Pearson Prentice Hall, Pearson Education, Inc., Upper Saddle River, New Jorsey. All rights reserved. Permission to use this unpublished Work is granted to individuals registering through Melinda_Maggerty@prenhall.com for the instructional purposes not exceeding one academic term or semester.

Chapter 1 Introduction

Dave Bowman: Open the pod bay doors, HAL. HAL: I'm sorry Dave, I'm afraid I can't do that. Stanley Kubrick and Arthur C. Clarke, screenplay of 2001: A Space Odyssey

The idea of giving computers the ability to process human language is as old as the idea of computers themselves. This book is about the implementation and implications of that exciting idea. We introduce a vibrant interdisciplinary field with many names corresponding to its many facets, names like speech and language processing, human language technology, natural language processing, computational linguistics, and speech recognition and synthesis. The goal of this new field is to get computers to perform useful tasks involving human language, tasks like enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech.

Conversational agent

Dialogue system

One example of a useful such task is a **conversational agent**. The HAL 9000 computer in Stanley Kubrick's film 2001: A Space Odyssey is one of the most recognizable characters in twentieth-century cinema. HAL is an artificial agent capable of such advanced language-processing behavior as speaking and understanding English, and at a crucial moment in the plot, even reading lips. It is now clear that HAL's creator Arhur C. Clarke was a little optimistic in predicting when an artificial agent such as HAL would be available. But just how far off was he? What would it take to create at least the language-related parts of HAL? We call programs like HAL that converse with humans via natural language **conversational agents** or **dialogue systems**. In this text we study the various components that make up modern conversational agents, including language input (automatic speech recognition and natural language

Let's turn to another useful language-related task, that of making available to non-English-speaking readers the vast amount of scientific information on the Web in English. Or translating for English speakers the hundreds of millions of Web pages written in other languages like Chinese. The goal of **machine translation** is to automatically translate a document from one language to another. We will introduce the algorithms and mathematical tools needed to understand how modern machine translation works. Machine translation is far from a solved problem; we will cover the algorithms currently used in the field, as well as important component tasks.

Question answering

Machine translation

Many other language processing tasks are also related to the Web. Another such task is **Web-based question answering**. This is a generalization of simple web search, where instead of just typing keywords a user might ask complete questions, ranging from easy to hard, like the following:

- What does "divergent" mean?
- · What year was Abraham Lincoln born?
- · How many states were in the United States that year?

× = screenshot-jurafsky08-speech	-and-language-	processing-pdftote>	ct-output.txt		
Open 🔻 🗔				:	Sav
1 P R E L I M I N A R Y P R 0 O F S. 2 c 3 Unpublished Work ◆2008 4 by Pearson Education, Inc. To be publi 5 Pearson Education, Inc., Upper Saddle 6 this unpublished Work is granted to in	shed by Pearson River, New Jers dividuals regis	Prentice Hall, ay. All rights reserv tering through Melind	red. Permission to a_Haggerty@prenha	use 11.com	
/ for the instructional purposes not exc § Chapter 1 Dintroduction Dintroduction 1 Stanley Kubrick and Arthur C. Clarke, 14 screenplay of 2001: A Space Odyssey 15 FT 17 18 D 17 18 D 20 Conversational	AL. t do that.	emic term or semester			
<pre>24 ayent 25 ayent 26 ayent 26 of an of giving computers the abili 25 of computers themmeelves. This book is 26 that exciting idea. We introduce a vib many facets, names like speech and lan 27 language technology, natural language 28 speech recognition and synthesis. The 20 to perform useful tasks involving human- 20 communication, improving human-human c 21 one example of a useful such task is a 21 film 2001: A Space 00yssey is one of the 21 characters in twentieth century cinema 27 cocreasing behavior as speaking and un 24 c. Clarke was a little optimistic in p 25 the language-related parts of HAL? We language conversational agents or dal 27 study the various components that makes 20 latis turn to another useful language 20 latis turn to another useful language 21 translate a document from one language 21 translate a document from one anguage 21 and themslation speech to under 21 mathematical tools needed to under 21 dang well as important components 21 dang well as important components 21 translate a document from one anguage 22 and mathematical tools needed to under 35 mould be avellable for the speech recogn 22 havella tarolate a document from one anguage 24 and mathematical tools needed to under 35 mould be avellable for the speech recogn 24 mound the language agences in far from a solv field, as well as important components 35 task is Web-based question answering. 35 where instaad of just typing kaywords.</pre>	ty to process h about the imple and interdisci guage processing, com goal of this me no language, tas ommunication, o interdiscipation derstanding Eng interdiscipation of the second of the second interdiscipation of the second interdiscipation of machine oal of machine is a gener asss. a user might as a user might as	<pre>mman language is as a mentation and implica joinary field with mar a, human putational linguistic w field is to get comp so that is to get comp agent. The HAL SOUD able lish, and at a lish, and at a has a trificial agent capable lish, and at a matrificial agent capable an artificial agent sa mould it take to cre rail anguage understs an artificial agent sa matrificial agent sa ma</pre>	Id as the idea tions of y names correspor s, and uters processing of te computer in Stanl of such advanced creator Arthur ich as HAL tate at least with humans via icluding idea to nonEnglish- ide to nonEnglish- idea the stanl for comatically igorithms works. thms currently us such eb search, ranging	nding to ey Kubr languag natural ge outp speakin English	ut g
19 Dialogue system 30 31 Machine 32 translation 53 answering 54 Waat vas "divergent" mean? 59 What year was Abraham Lincoln born?	totos that				
55 - now many states were in the United S 30 61 m 10	tates that year	r			
	plate Tart -	Televidele a -		_	IN

The goal of text preprocessing is its conversion into a canonical form.

PRELIMINARY PROOFS.

Unpublished Work ©2008 by Pearson Education, Inc. To be published by Pearson Prentice Hall, Pearson Education, Inc., Upper Saddle River, New Jorsey. All rights reserved. Permission to use this unpublished Work is granted to individuals registering through Melinda_Maggerty@prenhall.com for the instructional purposes not exceeding one academic term or semester.

Chapter 1 Introduction

Dave Bowman: Open the pod bay doors, HAL. HAL: I'm sorry Dave, I'm afraid I can't do that. Stanley Kubrick and Arthur C. Clarke, screenplay of 2001: A Space Odyssey

The idea of giving computers the ability to process human language is as old as the idea of computers themselves. This book is about the implementation and implications of that exciting idea. We introduce a vibrant interdisciplinary field with many names corresponding to its many facets, names like speech and language processing, human language technology, natural language processing, computational linguistics, and speech recognition and synthesis. The goal of this new field is to get computers to perform useful tasks involving human language, tasks like enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech.

Conversational agent

Dialogue system

One example of a useful such task is a **conversational agent**. The HAL 9000 computer in Stanley Kubrick's film 2001: A Space Odyssey is one of the most recognizable characters in twentieth-century cinema. HAL is an artificial agent capable of such advanced language-processing behavior as speaking and understanding English, and at a crucial moment in the plot, even reading lips. It is now clear that HAL's creator Arthur C. Clarke was a little optimistic in predicting when an artificial agent such as HAL would be available. But just how far off was he? What would it take to create at least the language-related parts of HAL? We call programs like HAL that converse with humans via natural language **conversational agents** or dialogue systems. In this text we study the various components that make up modern conversational agents, including language input (automatic speech recognition and natural language

Let's turn to another useful language-related task, that of making available to non-English-speaking readers the vast amount of scientific information on the Web in English. Or translating for English speakers the hundreds of millions of Web pages written in other languages like Chinese. The goal of machine translation is to automatically translate a document from one language to another. We will introduce the algorithms and mathematical tools needed to understand how modern machine translation works. Machine translation is far from a solved problem; we will cover the algorithms currently used in the field, as well as important component tasks.

Question answering

Machine translation

Many other language processing tasks are also related to the Web. Another such task is **Web-based question answering**. This is a generalization of simple web search, where instead of just typing keywords a user might ask complete questions, ranging from easy to hard, like the following:

- What does "divergent" mean?
- · What year was Abraham Lincoln born?
- · How many states were in the United States that year?

	anchat jurafeku08 saaash	and language	processing clappo	d byb	
	ensnot-jurarsky08-speech	-and-tanguage	-processing-cleane	a.cxc	
Open 🔻	J+l				Save
1 Chapter 1 2 Introductio	n				
3 4 Dave Bowman	: Open the pod bay doors. H	AL.			
5 HAL: I'm so 6 Stanley Kub 7	rry Dave, I'm afráid I can'i rick and Arthur C. Clarke, s	t do that. screenplay of 2	001: A Space Odysse	У	
8 The idea of themselves. a vibrant i and languag linguistics perform use improving h	giving computers the abili This book is about the imp nterdisciplinary field with e processing, human languag , and speech recognition an ful tasks involving human l uman-human communication, o	ty to process h lementation and many names cor e technology, n d synthesis. Th anguage, tasks r simply doing	uman language is as implications of the responding to its m atural language pro- e goal of this new like enabling human- useful processing o	old as the idea of a at exciting idea. We any facets, names lil cessing, computationa field is to get compu- machine communicatio f text or speech.	computers introduce ke speech al uters to on,
9 0 One example film 2001: is an artif HAL's creat HAL would b related par conversatio modern conv language un	of a useful such task is a 4 Space Odyssey is one of t icial agent capable of such ng English, and at a crucia or Arthur C. Clarke was a 1 e available. But just how f. so f HA? We call programs nal agents or dialogue syst resational agents, includin derstanding) and language o	conversational ne most recogni advanced langu I moment in the ittle optimisti ar off was he? like HAL that ems. In this te g language inpu utput (natural	agent. The HAL 900 zable characters in age-processing beha plot, even reading c in predicting whe What would it take converse with human xt we study the var t (automatic speech language generation	O computer in Stanley twentieth-century ci vior as speaking and lips. It is now clees n an artificial agent to create at least th s via natural languag ious components that recognition and natu and speech synthesis	y Kubrick': inema. HAL ar that t such as ne languag ge make up ural s).
 Let's turn readers the speakers th machine tra introduce t works. Mach the field, Many other 	to another useful language- vast amount of scientific. e hundreds of millions of W nslation is to automatically he algorithms and mathemati ine translation is far from as well as important compon language <u>processing tasks</u> a	related task, t information on ab pages writte y translate a d cal tools neede a solved probl ent tasks. re also related	hat of making avail the Web in English. n in other language: ocument from one lan d to understand how em; we will cover th to the Web. Anothe	able to nonEnglish—sp Or translating for B s like Chinese. The o nguage to another. We modern machine trans he algorithms current r such task is Web—ba	peaking English goal of e will slation tly used in ased
keywords a 5 — What does 6 — What year 7 — How many	wering. This is a general, user might ask complete que: "divergent" mean? was Abraham Lincoln born? states were in the United S ¹	tates that year	from easy to hard,	like the following:	ing
		Plain Text 🔻	Tab Width: 2 🔻	Ln 14, Col 1	 INS

Overview

Rationale:

Ease implementation of subsequent processing steps

A unified input format simplifies implementing processing steps. Example: In web search engines, all documents are converted to HTML. All indexing steps can expect HTML as input.

Avoid processing errors and model bias

Many rule-based and learning-based processing steps in an NLP pipeline may fail or be misled because of random text artifacts. High-level processing steps presume clean text. Examples: Text classifications may learn to exploit PDF-to-text conversion artifacts rather than a text's contents; parsers require grammatical text.

Constraints:

Task-dependence

The canonical form depends on the task at hand and its requirements.

Provenance

The possibility to determine from where in a raw text corpus, a preprocessed text originated.

Reversibility

The capability to render a preprocessed text in human-readable form.

Overview

Common preprocessing steps:

- Conversion to plain text
- Encoding detection and unification
- □ Line break unification (n UNIX, rn Windows)
- Extraction of main content and meta information
- Normalization and/or paraphrasing
- Annotation

Plain text formats:

- □ unformatted, raw
- □ formatted and/or markup
- □ inline or external annotations

Overview

Normalization and/or paraphrasing:

- □ Faithful to the original text
- Departure from the original text
 - Unification across texts
 - Canonicalization

Whitespace, spelling, grammar; translation of text messages to common text norms

- Expansion and/or abstraction

Abbreviations, anaphora, translation to spoken language, canonicalization of tokens

Annotation:

- □ Syntactic units: phonemes, morphemes, tokens (esp. words), sentences
- Discourse units: paragraphs, sections, chapters
- **Typographic units:** lines, pages (layout, meta-information), documents
- □ Meta-information: title, authors, date, properties, ...

Tokenization

Tokenization turns a sequence of characters into a sequence of tokens.

Example:

```
Friends, Romans, Countrymen, lend me your ears !
Friends , Romans , Countrymen , lend me your ears !
```

Terminology: (simplified)

- □ A token is a character sequence forming a useful semantic unit.
- □ A type is to a token what a class is to an object.

Token-granularity:

- □ Word-level: may or may not inlcude whitespace between words
- Phrase-level: identification of multi-term named entities and common phrases
- Sentence-level: one token corresponds to one clause, or one sentence

Remarks:

- A related philosophical concept is the type-token distinction (see unit about corpus linguistics in this course). Here, a token is a specific instance of a word (i.e., its specific written form), and a type refers to its underlying concept as a whole. This is comparable to the distinction between class and object in object-oriented computer programming. For example, the sentence "A rose is a rose is a rose." comprises nine token instances but only four types, namely "a", "rose", "is", and ".".
- Tokenization is strongly language-dependent. English is already among the easiest languages to be tokenized, and there are still many problems to be solved. In Chinese, for example, words are not separated by a specific character, rendering the process of determining word boundaries much more difficult.

Tokenization: Special Cases

Contractions

Apostrophes can be a part of a word, a part of a possessive, or just a mistake: it's, o'donnell, can't, don't, 80's, men's, master's degree, shriner's

Hyphenated compounds

Hyphens may be part of a word, a separator, and some words refer to the same concept with or without hyphen: winston-salem, e-bay, wal-mart, active-x, far-reaching, loud-mouthed, 20-year-old.

Compounds

English: wheelchair, German: Computerlinguistik for computational linguistics.

Other special characters

Special characters may form part of words, **especially in technology-related text**: M*A*S*H, I.B.M., Ph.D., C++, C#, , http://www.example.com.

Numbers

Numbers form tokens of their own, and may contain punctuation as well: 6.5, 1e+010.

Derived Phrase tokens: named entities, phone numbers, dates

San Francisco, (800) 234-2333, Mar 11,1983.

Tokenization Approaches

Heuristics

- Whitespace: A token is every character sequence separated by whitespace characters.
- TREC: A token is every alphanumeric sequence of characters of length > 3, separated by a space or punctuation mark.

Rule-based

Applications of rules to a text so that tokens are separated by whitespace from each other. This allows subsequent processing steps to apply the whitespace heuristic

Machine learning-based

Based on sufficiently large training data of correctly tokenized text, a model can be trained to decide at every character position whether to split tokens.

Rule-based Tokenization [Jurasky and Martin, 2007] [Grefenstette, 1999]

- Algorithm: Tokenization with Regular Expressions.
- Input: *d*. Document in the form of a string.
 - A. Dictionary of abbreviations.
- Output: The document with space in-between its tokens.

Tokenize(d, A)

- 1. alnum = [A-Za-z0-9]; nalnum = [^A-Za-z0-9]; alwayssep = [?!()";/\|']
- 2. <u>clitic</u> = ('|:|-|'S|'D|'M|'LL|'RE|'VE|N'T|'s|'d|'m|'ll|'re|'ve|n't)
- 3. // Put whitespace around unambiguous separators.
- 4. // Put whitespace around commas that aren't inside numbers.
- 5. // Segment single quotes not preceded by letter (not apostrophes).
- 6. // Segment unambiguous word-final clitics and punctuation.
- 7. Split d by whitespace (/\s+/) to obtain a list of tokens T.
- 8. // Segment periods from each $t \in T$ that isn't an abbreviation in A or like one (letter period sequence or letter followed by consonants).
- 9. // Optionally expand clitics to normalize them.
- 10. Return a whitespace-separated string of T.

Rule-based Tokenization [Jurasky and Martin, 2007] [Grefenstette, 1999]

- Algorithm: Tokenization with Regular Expressions.
- Input: *d*. Document in the form of a string.
 - A. Dictionary of abbreviations.
- Output: The document with space in-between its tokens.

Tokenize(d, A)

- 1. alnum = [A-Za-z0-9]; nalnum = [^A-Za-z0-9]; alwayssep = [?!()";/\|']
- 2. <u>clitic</u> = ('|:|-|'S|'D|'M|'LL|'RE|'VE|N'T|'s|'d|'m|'ll|'re|'ve|n't)
- 3. Apply **s/\$alwayssep/_\$&_/g** to d.
- 4. Apply s/([^0-9]),/\$1_,_/g and s/,([^0-9])/_,_\$1/g to d.
- 5. Apply s/^'/\$&_/g and s/(\$nalnum)'/\$1_'/g to d.
- 6. Apply s/\$clitic\$/_\$&/g and s/\$clitic(\$nalnum)/_\$1_\$2/g to d.
- 7. Split d by whitespace (/\s+/) to obtain a list of tokens T.
- 8. Apply s/\.\$/_\./ to t∈T if t matches /\$alnum\./ and is not in A and doesn't match /^([A-Za-z]\.)+|[A-Z][bcdfghj-np-tvxz]+\.)\$/.
- 9. Optionally expand clitics: **s/'ve/have/** and **s/'m/am/** and so on.
- 10. Return a whitespace-separated string of \boldsymbol{T} .

Stopping (Token Removal)

Stopping refers to the removal of tokens from a token sequence that are not useful in order to reduce data and improve performance of subsequent tasks:

- Frequent tokens (collection-specific)
 Example: Wikipedia when processing Wikipedia.
- Function word tokens (language-dependent)
 the, of, and, etc; strong overlap with frequent tokens.
 Problem: to be or not to be would be completely lost.
- Punctuation-only tokens Counter-example: ; -)
- Number-only tokens

Short tokens

Short words may be important. Examples: xp, ma, pm, ben e king, el paso, master p, gm, j lo, world war II.

Stop word lists are typically customized to the text domain.

The top 100 most common words account for up to 50% of all words. [Wikipedia]

Example:

The idea of giving computers the ability to process human language is as old as the idea of computers themselves. This book is about the implementation and implications of that exciting idea. We introduce a vibrant interdisciplinary field with many names corresponding to its many facets, names like speech and language processing, human language technology, natural language processing, computational linguistics, and speech recognition and synthesis. The goal of this new field is to get computers to perform useful tasks involving human language, tasks like enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech.

Example:

The idea of giving computers the ability to process human language is as old as the idea of computers themselves. This book is about the implementation and implications of that exciting idea. We introduce a vibrant interdisciplinary field with many names corresponding to its many facets, names like speech and language processing, human language technology, natural language processing, computational linguistics, and speech recognition and synthesis. The goal of this new field is to get computers to perform useful tasks involving human language, tasks like enabling human-machine communication, improving human-human communication, or simply doing useful processing of text or speech.

Token Normalization

Application of heuristic rules to each token in an attempt to unify them.

□ Lower-casing

Problem: Capitalization may carry distinctions between word semantics. Examples: Bush vs. bush, Apple vs. apple.

Removal of special characters

Example: U.S.A. \rightarrow USA

Removal of diacritical marks

Example: café \rightarrow cafe

Spelling correction

Example: My gramma got die of beaties \rightarrow My grandma got diabetes

Reduction of morphology

Lemmatization or stemming heuristics