B. Sucha

CONTENTS

Scanned by CamScanner

Scanned by CamScanner

INTRODUCTION

CHAPTER OVERVIEW

This chapter gives an idea of natural language processing (NLP) and information retrieval (IR). Various levels of analysis involved in NLP along with the knowledge used by these levels of analysis are discussed. Some of the difficulties in analysing text and specific factors that make automatic processing of languages difficult are also touched upon. The chapter underlines the role of grammar in language processing and introduces transformational grammar. Indian languages differ a lot from English. These differences are clearly pointed out. Further, a number of NLP applications are introduced along with some of the early NLP systems. Towards the end, information retrieval is discussed.

1.1 WHAT IS NATURAL LANGUAGE PROCESSING (NLP)

Language is the primary means of communication used by humans. It is the tool we use to express the greater part of our ideas and emotions. It shapes thought, has a structure, and carries meaning. Learning new concepts and expressing ideas through them is so natural that we hardly realize how we process natural language. But there must be some kind of representation in our mind, of the content of language. When we want to express a thought, this content helps represent language in real time. As children, we never learn a computational model of language, yet this is the first step in the automatic processing of languages. *Natural language processing* (NLP) is concerned with the development of computational models of aspects of human language processing. There are two main reasons for such development:

1. To develop automated tools for language processing
2. To gain a better understanding of human communication

Building computational models with human language-processing abilities requires a knowledge of how humans acquire, store, and process language. It also requires a knowledge of the world and of language.

Scanned by CamScanner

Historically, there have been two major approaches to NLP—the *rationalist* approach and the *empiricist* approach. Early NLP research took a rationalist approach, which assumes the existence of some language faculty in the human brain. Supporters of this approach argue that it is not possible for children to learn a complex thing like natural language from limited sensory inputs. Empiricists do not believe in existence of a language faculty. Instead, they believe in the existence of some general organization principles such as pattern recognition, generalization, and association. Learning of detailed structures can, therefore, take place through the application of these principles on sensory inputs available to the child.

ORIGINS OF NLP

Natural language processing sometimes mistakenly termed natural language understanding—originated from machine translation research. While natural language understanding involves only the interpretation of language, natural language processing includes both understanding (interpretation) and generation (production). The NLP also includes speech processing. However, in this book, we are concerned with text processing only, covering work in the area of computational linguistics, and the tasks in which NLP has found useful application.

Computational linguistics is similar to theoretical- and psycho-linguistics, but uses different tools. Theoretical linguists mainly provide structural description of natural language and its semantics. They are not concerned with the actual processing of sentences or generation of sentences from structural description. They are in a quest for principles that remain common across languages and identify rules that capture linguistic generalization. For example, most languages have constructs like noun and verb phrases. Theoretical linguists identify rules that describe and restrict the structure of languages (grammar). Psycholinguists explain how humans produce and comprehend natural language. Unlike theoretical linguists, they are interested in the representation of linguistic structures as well as in the process by which these structures are produced. They rely primarily on empirical investigations to back up their theories.

Computational linguistics is concerned with the study of language using computational models of linguistic phenomena. It deals with the application of linguistic theories and computational techniques for NLP. In computational linguistics, representing a language is a major problem; most knowledge representations tackle only a small part of knowledge. Representing the whole body of knowledge is almost impossible. The words knowledge and language should not be confused. This is discussed in detail in Section 1.3.

Computational models may be broadly classified under knowledge-driven and data-driven categories. Knowledge-driven systems rely on explicitly coded linguistic knowledge, often expressed as a set of handcrafted grammar rules. Acquiring and encoding such knowledge is difficult and is the main bottleneck in the development of such systems. They are, therefore, often constrained by the lack of sufficient coverage of domain knowledge. Data-driven approaches presume the existence of a large amount of data and usually employ some machine learning technique to learn syntactic patterns. The amount of human effort is less and the performance of these systems is dependent on the quantity of the data. These systems are usually adaptive to noisy data.

As mentioned earlier, this book is mainly concerned with computational linguistics approaches. We try to achieve a balance between semantic (knowledge-driven) and data-driven approaches on one hand, and between theory and practice on the other. It is at this point that the book differs significantly from other textbooks in this area. The tools and techniques have been covered to the extent that is needed to build sufficient understanding of the domain and to provide a base for application.

The NLP is no longer confined to classroom teaching and a few traditional applications. With the unprecedented amount of information now available on the web, NLP has become one of the leading techniques for processing and retrieving information. In order to cope with these developments, this book brings together information retrieval with NLP. The term *information retrieval* is used here in a broad manner to include a number of information processing applications such as information extraction, text summarization, question answering, and so forth. The distinction between these applications is made in terms of the level of detail or amount of information retrieved. We consider retrieval of information as part of processing. The word 'information' itself has a much broader sense. It includes multiple modes of information, including speech, images, and text. However, it is not possible to cover all these modes due to space constraints. Hence, this book focuses on textual information only.

1.3 LANGUAGE AND KNOWLEDGE

Language is the medium of expression in which knowledge is deciphered. We are not competent enough to define language and knowledge and its

Scanned by CamScanner

implications. We are here considering the text form of the language and the content of it as knowledge.

Language, being a medium of expression, is the outer form of the content it expresses. The same content can be expressed in different languages. But can language be separated from its content? If so, how can the content itself be represented? Generally, the meaning of one language is written in the same language (but with a different set of words). It may also be written in some other, formal, language. Hence, to process a language means to process the content of it. As computers are not able to understand natural language, methods are developed to map its content in a formal language. Sometimes, formal language content may have to be expressed in a natural language as well. Thus, in this book, language is taken up as a knowledge representation tool that has historically represented the whole body of knowledge and that has been modified, maybe through generation of new words, to include new ideas and situations. The language and speech community, on the other hand, considers a language as a set of sounds that, through combinations, conveys meaning to a listener. However, we are concerned with representing and processing text only. Language (text) processing has different levels, each involving different types of knowledge. We now discuss various levels of processing and the types of knowledge it involves.

The simplest level of analysis is *lexical analysis*, which involves analysis of words. Words are the most fundamental unit (syntactic as well as semantic) of any natural language text. Word-level processing requires morphological knowledge, i.e., knowledge about the structure and formation of words from basic units (morphemes). The rules for forming words from morphemes are language specific.

The next level of analysis is *syntactic analysis*, which considers a sequence of words as a unit, usually a sentence, and finds its structure. Syntactic analysis decomposes a sentence into its constituents (or words) and identifies how they relate to each other. It captures grammaticality or non-grammaticality of sentences by looking at constraints like word order, number, and case agreement. This level of processing requires syntactic knowledge, i.e., knowledge about how words are combined to form larger units such as phrases and sentences, and what constraints are imposed on them. Not every sequence of words results in a sentence. For example, 'I went to the market' is a valid sentence whereas 'went the I market to' is not. Similarly, 'She is going to the market' is valid, but 'She are going to the market' is not. Thus, this level of analysis requires detailed knowledge about rules of grammar.

Yet another level of analysis is *semantic analysis*. Semantics is associated with the meaning of the language. Semantic analysis is concerned with creating meaningful representation of linguistic inputs. The general idea of semantic interpretation is to take natural language sentences or utterances and map them onto some representation of meaning. Defining meaning components is difficult as grammatically valid sentences can be meaningless. One of the famous examples is, 'Colorless green ideas sleep furiously' (Chomsky 1957). The sentence is well-formed, i.e., syntactically correct, but semantically anomalous. However, this does not mean that syntax has no role to play in meaning. Bach (2002) considers:

> '... *semantics to be a projection of its syntax. That is semantic structure is interpreted syntactic structure.*'

But definitely, syntax is not the only component to contribute meaning. Our conception of meaning is quite broad. We feel that humans apply all sorts of knowledge (i.e., lexical, syntactic, semantic, discourse, pragmatic, and world knowledge) to arrive at the meaning of a sentence. The starting point in semantic analysis, however, has been lexical semantics (meaning of words). A word can have a number of possible meanings associated with it. But in a given context, only one of these meanings participates. Finding out the correct meaning of a particular use of word is necessary to find meaning of larger units. However, the meaning of a sentence cannot be composed solely on the basis of the meaning of its words. Consider the following sentences:

Kabir and Ayan are married. (1.1a)

Kabir and Suha are married. (1.1b)

Both sentences have identical structures, and the meanings of individual words are clear. But most of us end up with two different interpretations. We may interpret the second sentence to mean that Kabir and Suha are married to each other, but this interpretation does not occur for the first sentence. Syntactic structure and compositional semantics fail to explain these interpretations. We make use of pragmatic information. This means that semantic analysis requires pragmatic knowledge besides semantic and syntactic knowledge.

A still higher level of analysis is *discourse analysis*. Discourse-level processing attempts to interpret the structure and meaning of even larger units, e.g., at the paragraph and document level, in terms of words, phrases, clusters, and sentences. It requires the resolution of anaphoric references and identification of discourse structure. It also requires discourse knowledge, that is, knowledge of how the meaning of a sentence is determined by preceding sentences—e.g., how a pronoun refers to the

Scanned by CamScanner

preceding noun—and how to determine the function of a sentence in the text. In fact, pragmatic knowledge may be needed for resolving anaphoric references. For example, in the following sentences, resolving the anaphoric reference 'they' requires pragmatic knowledge:

The district administration refused to give the trade union permission for the meeting because they feared violence. (1.2a)

The district administration refused to give the trade union permission for the meeting because they oppose government. (1.2b)

The highest level of processing is *pragmatic analysis*, which deals with the purposeful use of sentences in situations. It requires knowledge of the world, i.e., knowledge that extends beyond the contents of the text. The Cyc project (Lenat 1986) at University of Austin is an attempt to utilize world knowledge in NLP. However, its usefulness in a general-domain NLP system is yet to be demonstrated. Furthermore, whether or not semantics can be associated with a symbol manipulator and whether humans use logic in the same way as the Cyc project, are both issues of debate.

4 THE CHALLENGES OF NLP

There are a number of factors that make NLP difficult. These relate to the problems of representation and interpretation. Language computing requires precise representation of content. Given that natural languages are highly ambiguous and vague, achieving such representation can be difficult. The inability to capture all the required knowledge is another source of difficulty. It is almost impossible to embody all sources of knowledge that humans use to process language. Even if this were done, it is not possible to write procedures that imitate language processing as done by humans. In this section, we detail some of the problems associated with NLP.

Perhaps the greatest source of difficulty in natural language is identifying its semantics. The principle of compositional semantics considers the meaning of a sentence to be a composition of the meaning of words appearing in it. In the earlier section, we saw a number of examples where this principle failed to work. Our viewpoint is that words alone do not make a sentence. Instead, it is the words as well as their syntactic and semantic relation that give meaning to a sentence. As pointed out by Wittgenstein (1953): 'The meaning of a word is its use in the language.' A language keeps on evolving. New words are added continually and existing words are introduced in new context. For example, most newspapers and TV channels use 9/11 to refer to the terrorist act on the World Trade Centre in the USA in 2004. When we process written text or spoken utterances, we have access to underlying mental representation. The only way a machine can learn the meaning of a specific word in a message is by considering its context, unless some explicitly coded general world or domain knowledge is available. The context of a word is defined by co-occurring words. It includes everything that occurs before or after a word. The frequency of a word being used in a particular sense also affects its meaning. The English word 'while' was initially used to mean 'a short interval of time'. But now it is more in use as a conjunction. None of the usages of 'while' discussed in this chapter correspond to this meaning.

Idioms, metaphor, and ellipses add more complexity to identify the meaning of the written text. As an example, consider the sentence:

The old man finally kicked the bucket. (1.3)

The meaning of this sentence has nothing to do with the words 'kick' and 'bucket' appearing in it.

Quantifier-scoping is another problem. The scope of quantifiers (the, each, etc.) is often not clear and poses problem in automatic processing.

The ambiguity of natural languages is another difficulty. These go unnoticed most of the times, yet are correctly interpreted. This is possible because we use explicit as well as implicit sources of knowledge. Communication via language involves two brains not just one—the brain of the speaker/writer and that of the hearer/reader. Anything that is assumed to be known to the receiver is not explicitly encoded. The receiver possesses the necessary knowledge and fills in the gaps while making an interpretation. As humans, we are aware of the context and current cultural knowledge, and also of the language and traditions, and utilize these to process the meaning. However, incorporating contextual and world knowledge poses the greatest difficulty in language computing. An example of cultural impact on language is the representation of different shades of white in the Eskimo world. It may be hard for a person living in plain to distinguish among various shades. Similarly, to an Indian, the word 'Taj' may mean a monument, a brand of tea, or a hotel, which may not be so for a non-Indian. Let us now take a look at the various sources of ambiguities in natural languages.

The first level of ambiguity arises at the word level. Without much effort, we can identify words that have multiple meanings associated with

Scanned by CamScanner

them, e.g., bank, can, bat, and still. A word may be ambiguous in its part-of-speech or it may be ambiguous in its meaning. The word 'can' is ambiguous in its part-of-speech whereas the word 'bat' is ambiguous in its meaning. We hardly consider all possible meanings of a word to get the correct one. A program on the other hand, must be explicitly coded to resolve each meaning. Hence, we need to develop various models and algorithms to resolve them. Deciding whether 'can' is a noun or a verb is solved by 'part-of-speech tagging' whereas identifying whether a particular use of 'bank' corresponds to 'financial institution' sense or 'river bank' sense is solved by 'word sense disambiguation'. 'Part-of-speech tagging' and 'word sense disambiguation' algorithms are discussed in Chapters 3 and 5 respectively.

A sentence may be ambiguous even if the words are not, for example, the sentence: '*Stolen rifle found by tree.*' None of the words in this sentence is ambiguous but the sentence is. This is an example of structural ambiguity. Verb sub-categorization may help to resolve this type of ambiguity but not always. Probabilistic parsing, which is discussed in Chapter 4, is another solution. At a still higher level are pragmatic and discourse ambiguities. Ambiguities are discussed in Chapter 5.

A number of grammars have been proposed to describe the structure of sentences. However, there are an infinite number of ways to generate them, which makes writing grammar rules, and grammar itself, extremely complex. On top of it, we often make correct semantic interpretations of non-grammatical sentences. This fact makes it almost impossible for grammar to capture the structure of all and only meaningful text.

1.5 LANGUAGE AND GRAMMAR

Automatic processing of language requires the rules and exceptions of a language to be explained to the computer. Grammar defines language. It consists of a set of rules that allows us to parse and generate sentences in a language. Thus, it provides the means to specify natural language. These rules relate information to coding devices at the language level—not at the world-knowledge level (Bharati et al. 1995). However, since world knowledge affects both the coding (i.e., words) and the coding convention (structure), this blurs the boundary between syntax and semantics. Nevertheless such a separation is made because of the ease of processing and grammar writing.

The main hurdle in language specification comes from the constantly [illegible] nature of natural languages and the presence of a large number of hard-to-specify exceptions. Several efforts have been made to provide such specifications, which has led to the development of a number of grammars. Main among them are transformational grammar (Chomsky 1957), lexical functional grammar (Kaplan and Bresnan 1982), government and binding (Chomsky 1981), generalized phrase structure grammar, transformational grammar (Chomsky 1957), dependency grammar, Paninian grammar, and tree-adjoining grammar (Joshi 1985). Some of these grammars focus on derivation (e.g., phrase structure grammar) while others focus on relationships (e.g., dependency grammar, lexical functional grammar, Paninian grammar, and link grammar). We discuss some of these in Chapter 2. The greatest contribution to grammar comes from Noam Chomsky, who proposed a hierarchy of formal grammar based on level of complexity. These grammars use phrase structure rules (or rewrite rules). The term 'generative grammar' is often used to refer to the general framework introduced by Chomsky. Generative grammar basically refers to any grammar that uses a set of rules to specify or generate all and only grammatical (well-formed) sentences in a language. Chomsky argued that phrase structure grammars are not adequate to specify natural language. He proposed a complex system of transformational grammar in his book on *Syntactic Structures* (1957), in which he suggested that each sentence in a language has two levels of representation, namely, a deep structure and a surface structure (See Figure 1.1). The mapping from deep structure to surface structure is carried out by transformations. In the following paragraphs, we introduce transformational grammar.

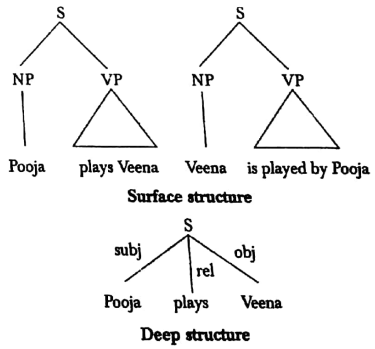

Figure 1.1 Surface and deep structures of sentence

Scanned by CamScanner

Transformational grammar was introduced by Chomsky in 1957. Chomsky argued that an utterance is the surface representation of a 'deeper structure' representing its meaning. The deep structure can be transformed in a number of ways to yield many different surface-level representations. Sentences with different surface-level representations having the same meaning, share a common deep-level representation. Chomsky's theory was able to explain why sentences like

Pooja plays veena. (1.4a)

Veena is played by Pooja. (1.4b)

have the same meaning, despite having different surface structures (roles of subject and object are inverted). Both the sentences are being generated from the same 'deep structure' in which the deep subject is Pooja and the deep object is the veena.

Transformational grammar has three components:

1. Phrase structure grammar
2. Transformational rules
3. Morphophonemic rules—These rules match each sentence representation to a string of phonemes.

Each of these components consists of a set of rules. Phrase structure grammar consists of rules that generate natural language sentences and assign a structural description to them. As an example, consider the following set of rules:

$S \rightarrow NP + VP$
$VP \rightarrow V + NP$
$NP \rightarrow Det + Noun$
$V \rightarrow Aux + Verb$
$Det \rightarrow$ the, a, an, ...
$Verb \rightarrow$ catch, write, eat, ...
$Noun \rightarrow$ police, snatcher, ...
$Aux \rightarrow$ will, is, can, ...

In these rules, S stands for sentence, NP for noun phrase, VP for verb phrase, and Det for determiner. Sentences that can be generated using these rules are termed grammatical. The structure assigned by the grammar is a constituent structure analysis of the sentence.

The second component of transformational grammar is a set of transformation rules, which transform one phrase-marker (underlying) into another phrase-marker (derived). These rules are applied on the terminal string generated by phrase structure rules. Unlike phrase structure rules, transformational rules are heterogeneous and may have more than one symbol on their left hand side. These rules are used to transform one surface representation into another, e.g., an active sentence into passive one. The rule relating active and passive sentences (as given by Chomsky) is

$$NP_1 - Aux - V - NP_2 \rightarrow NP_2 - Aux + be + en - V - by + NP_1$$

This rule says that an underlying input having the structure NP–Aux–V–NP can be transformed to NP – Aux + be + en – V – by + NP. This transformation involves addition of strings 'be' and 'en' and certain rearrangements of the constituents of a sentence. Transformational rules can be obligatory or optional. An obligatory transformation is one that ensures agreement in number of subject and verb, etc., whereas an optional transformation is one that modifies the structure of a sentence while preserving its meaning. Morphophonemic rules match each sentence representation to a string of phonemes.

Consider the active sentence:

The police will catch the snatcher. (1.5)

The application of phrase structure rules will assign the structure shown in Figure 1.2 to this sentence.

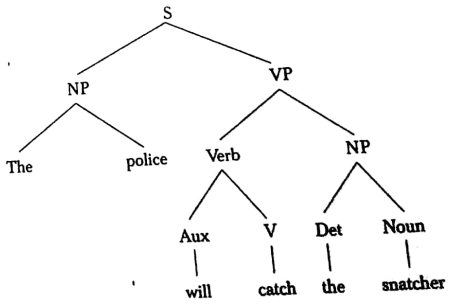

Figure 1.2 Parse structure of sentence (1.5)

The passive transformation rules will convert the sentence into:
The + culprit + will + be + en + catch + by + police (Figure 1.3).

Scanned by CamScanner

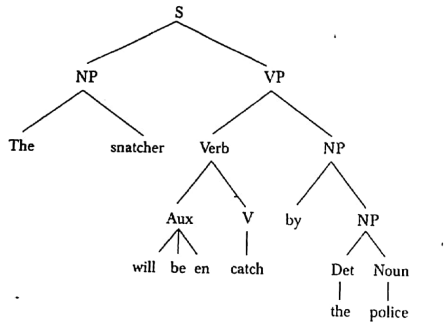

Figure 1.3 Structure of sentence (1.5) after applying passive transformations

Another transformational rule will then reorder 'en + catch' to 'catch + en' and subsequently one of the morphophonemic rules will convert 'catch + en' to 'caught'. In general, the noun phrase is not always as simple as in sentence (1.5). It may contain other embedded structures, such as adjectives, modifiers, relative clause, etc. Long distance dependencies are other language phenomena that cannot be adequately handled by phrase structure rules. Long distance dependency refers to syntactic phenomena where a verb and its subject or object can be arbitrarily apart. The problem in the specification of appropriate phrase structure rules occurs because these phenomena cannot be localized at the surface structure level (Joshi and Vijayshanker 1989). Wh-movement[1] are a specific case of these types of dependencies.

1.6 PROCESSING INDIAN LANGUAGES

There are a number of differences between Indian languages and English. This introduces differences in their processing. Some of these differences are listed here.

- Unlike English, Indic scripts have a non-linear structure.
- Unlike English, Indian languages have SOV (Subject-Object-Verb) as the default sentence structure.
- Indian languages have a free word order, i.e., words can be moved freely within a sentence without changing the meaning of the sentence.
- Spelling standardization is more subtle in Hindi than in English.
- Indian languages have a relatively rich set of morphological variants.
- Indian languages make extensive and productive use of complex predicates (CPs).
- Indian languages use post-position (*Karakas*) case markers instead of prepositions.
- Indian languages use verb complexes consisting of sequences of verbs, e.g., गा रहा है (*ga raha hai*—singing) and खेल रही है (*khel rahi hai*—playing). The auxiliary verbs in this sequence provide information about tense, aspect, modality, etc.

Except for the direction in which its script is written, Urdu is closely related to Hindi. Both share similar phonology, morphology, and syntax. Both are free-word-order languages and use post-positions. They also share a large amount of their vocabulary. Differences in the vocabulary arise mainly because a significant portion of Urdu vocabulary comes from Persian and Arabic, while Hindi borrows much of its vocabulary from Sanskrit.

Paninian grammar provides a framework for Indian language models. These can be used for computation of Indian languages. The grammar focuses on extraction of Karaka relations from a sentence. We talk about the details of modelling in Chapter 2. A parsing framework based on Paninian grammar is introduced in Chapter 4 and issues involved in Indian language translation (using Paninian grammar theory) are discussed in Chapter 8.

1.7 NLP APPLICATIONS

Machine translation is the first application area of NLP. It involves the complete linguistic analysis of a natural language sentence, and linguistic generation of an output sentence. It is one of the most comprehensive and most challenging tasks in the area (AI-complete). However, the recent dramatic progress in the field of NLP has found interesting applications in information retrieval, information extraction, text summarization, etc. This book offers an extensive coverage of these recent applications, and also of traditional ones like machine translation and natural language generation. The focus has been on bridging the gap between theory and practice rather than on offering a gamut of linguistic, psychological, and computational theories.

[1] fers to a syntactic phenomenon in which interrogative words, called wh-words, appear at the beginning of ntence. For example, when the direct object of the verb 'read' in the sentence 'She is reading a book' is ced with a wh-word, the sentence becomes 'What is she reading?' instead of 'She is reading what?'.

Scanned by CamScanner

The applications utilizing NLP include the following.

Machine Translation

This refers to automatic translation of text from one human language to another. In order to carry out this translation, it is necessary to have an understanding of words and phrases, grammars of the two languages involved, semantics of the languages, and world knowledge.

Speech Recognition

This is the process of mapping acoustic speech signals to a set of words. The difficulties arise due to wide variations in the pronunciation of words, homonym (e.g. dear and deer) and acoustic ambiguities (e.g., in the rest and interest).

Speech Synthesis

Speech synthesis refers to automatic production of speech (utterance of natural language sentences). Such systems can read out your mails on telephone, or even read out a storybook for you. In order to generate utterances, text has to be processed. So, NLP remains an important component of any speech synthesis system.

Natural Language Interfaces to Databases

Natural language interfaces allow querying a structured database using natural language sentences.

Information Retrieval

This is concerned with identifying documents relevant to a user's query. NLP techniques have found useful applications in information retrieval. Indexing (stop word elimination, stemming, phrase extraction, etc.), word sense disambiguation, query modification, and knowledge bases have also been used in IR system to enhance performance, e.g., by providing methods for query expansion. WordNet, LDOCE (*Longman Dictionary of Contemporary English*) and *Roget's Thesaurus* are some of the useful lexical resources for IR research.

Information Extraction

An information extraction system captures and outputs factual information contained within a document. Similar to an information retrieval system, it responds to a user's information need. However, unlike in an information retrieval system, the information need is not expressed as a keyword query. Instead it is specified as pre-defined database schemas or templates. Whereas an IR system identifies a subset of documents in a large repository of text database—e.g., in a library scenario, a subset of resources in a library—an information extraction system identifies a subset of information within a document that fits the pre-defined template.

Question Answering

Given a question and a set of documents, a question answering system attempts to find the precise answer, or at least the precise portion of text in which the answer appears. This is unlike an IR system, which returns whole document that seems relevant to the user's query. A question answering system is different from an information extraction system in that the content that is to be extracted is unknown. In general, a question answering system benefits from having an information extraction system to identify entities in the text. A question answering system requires more NLP than an information retrieval system or an information extraction system. It requires not only precise analysis of questions and portions of texts but also semantic as well as background knowledge to answer certain type of questions.

Text Summarization

This deals with the creation of summaries of documents and involves syntactic, semantic, and discourse level processing of text.

1.8 SOME SUCCESSFUL EARLY NLP SYSTEMS

***ELIZA* (Weizenbaum 1966)**

ELIZA is one of the earliest natural language understanding programs. It uses syntactic patterns to mimic human conversation with the user. Here is a sample conversation.

Eliza: Hello. I am ELIZA. How may I help you?
User: I am feeling a little bit sleepy.
Eliza: How long have you been feeling a little bit sleepy?
User: For almost half an hour.
Eliza: Please go on.

***SysTran* (System Translation)**

The first SysTran machine translation system was developed in 1969 for Russian–English translation. SysTran also provided the first on-line machine translation service called Babel Fish, which is used by AltaVista search engines for handling translation requests from users.

TAUM METEO

This is a natural language generation system used in Canada to generate weather reports. It accepts daily weather data and generates weather reports in English and French.

Scanned by CamScanner

***SHRDLU* (Winogard 1972)**

This is a natural language understanding system that simulates actions of a robot in a block world domain. It uses syntactic parsing and semantic reasoning to understand instructions. The user can ask the robot to manipulate the blocks, to tell the blocks configurations, and to explain its reasoning.

***LUNAR* (Woods 1977)**

This was an early question answering system that answered questions about moon rock.

1.9 INFORMATION RETRIEVAL

The availability of a large amount of text in electronic form has made it extremely difficult to get relevant information. Information retrieval systems aim at providing a solution to this.

The term 'information' should not be confused with the term 'entropy' (numerical measure of the uncertainty of an outcome) as it is used in communication theory. Information is being used here to reflect 'subject matter' or the 'content' of some text. We are not interested in 'digital communication', where bits and bytes are the information carriers. Instead our focus in on the communication taking place between human beings as expressed through natural languages. Information is always associated with some data (text, number, image, and so on): we are concerned with text only. Hence, we consider words as the carriers of information and written text as the message encoded in natural language.

As a cognitive activity, the word 'retrieval' refers to operation of accessing information from memory. We use the word 'retrieval' to refer to the operation of accessing information from some computer-based representation. Retrieval of information thus requires information to be processed and stored. Not all the information represented in computable form is retrieved. Instead, only the information relevant to the needs expressed in the form of query is located. In order to get this relevance, the stored and processed information needs to be compared against query representation. Information retrieval (IR) deals with all these facets. It is concerned with the organization, storage, retrieval, and evaluation of information relevant to the query.

Information retrieval deals with unstructured data. The retrieval is performed based on the content of the document rather than on its structure. The IR systems usually return a ranked list of documents. The IR components have been traditionally incorporated into different types

of information systems including database management systems, bibliographic text retrieval systems, question answering systems, and more recently in search engines.

Current approaches for accessing large text collections can be broadly classified into two categories. The first category consists of approaches that construct topic hierarchy, e.g., Yahoo. This helps the user locate documents of interest manually by traversing the hierarchy. However, it requires manual classification of new documents within the existing taxonomy. This makes it cost ineffective and inapplicable due to rapid growth of documents on the Web. The second category consists of approaches that rank the retrieved documents according to relevance. We discuss various IR models that support ranked retrieval in Chapter 9.

***Major Issues in Information Retrieval* (Siddiqui 2006)**

There are a number of issues involved in the design and evaluation of IR systems, which are briefly discussed in this section. The first important point is to choose a representation of the document. Most human knowledge is coded in natural language, which is difficult to use as knowledge representation language for computer systems. Most of the current retrieval models are based on keyword representation. This representation creates problems during retrieval due to polysemy, homonymy, and synonymy. Polysemy involves the phenomenon of a lexeme with multiple meaning. Homonymy is an ambiguity in which words that appear the same have unrelated meanings. Ambiguity makes it difficult for a computer to automatically determine the conceptual content of documents. Synonymy creates problem when a document is indexed with one term and the query contains a different term, and the two terms share a common meaning. Another problem associated with keyword-based retrieval is that it ignores semantic and contextual information in the retrieval process. This information is lost in the extraction of keywords from the text and cannot be recovered by the retrieval algorithms.

A related issue is that of inappropriate characterization of queries by the user. There can be many reasons for the vagueness and inaccuracy of the user's queries, say for instance, her lack of knowledge of the subject or even the inherent vagueness of the natural language. The user may fail to include relevant terms in the query or may include irrelevant terms. Inappropriate or inaccurate queries lead to poor retrieval performance. The problem of ill-specified query can be dealt with by modifying or expanding queries. An effective technique based on user-interaction is relevance feedback which modifies queries based on the feedback provided by the user on initial retrieval.

Scanned by CamScanner

In order to satisfy the user's request, an IR system matches document representation with query representation. Matching query representation with that of the document is another issue. A number of measures have been proposed to quantify the similarity between a query and the document to produce a ranked list of results. Selection of the appropriate similarity measure is a crucial issue in the design of IR systems.

Evaluating the performance of IR systems is also a major issue. There are many aspects of evaluation, the most important being the effectiveness of the system. Recall and precision are the most widely used measures of effectiveness.

As the major goal of IR is to search a document in a manner relevant to the query, understanding what constitutes relevance is also an important issue. Relevance is subjective in nature (Saracevic 1991). Only the user can tell the true relevance; it is not possible to measure this 'true relevance'. One may however, define the degree of relevance. Relevance has been considered as a binary concept, whereas it is in fact a continuous function (a document may be exactly what the user wants or it may be closely related). Current evaluation techniques do not support this continuity as it is quite difficult to put into practice. A number of relevance frameworks have been proposed (Saracevic 1996). These include the system, communication, psychological, and situational frameworks. The most inclusive is the situational framework, which is based on the cognitive view of the information seeking process and considers the importance of situation, context, multi-dimensionality, and time. A survey of relevance studies can be found in Mizzaro (1997). Most of the evaluations of IR systems have so far been done on document test collections with known relevance judgments.

The size of document collections and the varying needs of users also complicate text retrieval. Some users require answers of limited scope, while others require documents with a wider scope. These differing needs can require different and specialized retrieval methods. However, these are research issues and have not been dealt with in this book.

SUMMARY

- Language is the primary means of communication used by humans.
- Natural language processing is concerned with the development of computational models of aspects of human language processing.
- Theoretical linguists are mainly interested in providing a description of the structure and semantics of natural language, whereas computational linguists deal with the study of language from a computational point of view.
- Historically, there have been two major approaches to natural language processing, namely rationalist approach and empiricist approach.
- The highly ambiguous and vague nature of natural language makes it difficult to create a representation amenable to computing.

REFERENCES

Bach, Kent, 2002, *Meaning and Truth*, J. Keim Campbell, M. O'Rourke, and D. Shei (Eds.), Seven Bridges Press, New York, pp. 284–92.

Chomsky, Noam, 1957, *Syntactic Structures*, Mouton, The Hague.

_____1981, *Lectures on Government and Binding*, Foris Publications, Dordreeht, The Netherlands.

Joshi, Aravind K., 1985, 'Tree adjoining grammar: How much sensitivity is required to provide reasonable structural description,' *Natural Language Parsing*, D. Dowty, L. Karttunen, and A. Zwicky (Eds.), Cambridge University Press, Cambridge.

Joshi, Aravind K. and K. Vijayshanker, 1989, 'Treatment of long distance dependencies in LFG and TAG: functional uncertainty in LFG is a corollary in TAG,' *Proceedings of the 27th Annual Meeting on Association for Computational Linguistics*, Vancouver, British Columbia, pp. 220–27.

Kaplan, R.M. and Joan Bresnan, 1982, 'Lexical functional grammar: A formal system for grammatical representation,' *The Mental Representation of Grammatical Relations*, Joan Bresnan (Ed.), MIT Press, Cambridge.

Lenat, D.B., M. Prakash, and M. Shepherd, 1986, 'Cyc: using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks,' *AI Magazine*, 6(4).

Mizzaro, S., 1997, 'Relevance: the whole history,' *Journal of the American Society for Information Science*, 48(9), pp. 810–32.

Saracevic, T., 1991, 'Individual differences in organizing, searching and retrieving information,' *Proceedings of the 54th Annual Meeting of the American Society for Information Science (ASIS)*, pp. 82–86.

_____1996, 'Relevance reconsidered,' *Proceedings of CoLIS 2, Second International Conference on Conceptions of Library and Information Science: Integration in Perspective*, P. Ingwersen and N.O. Pors (Eds.), The Royal School of Librarianship, Copenhagen, pp. 201–18.

Siddiqui, Tanveer, 2006, 'Intelligent techniques for effective information retrieval: a conceptual graph-based approach,' *Ph.D. Thesis*, J.K. Institute of Applied Physics, Deptt. of Electronics and Communication, University of Allahabad.

Scanned by CamScanner

Weizenbaum, R., 1966, 'ELIZA—A computer program for the study of natural language communication between man and machine,' *Communications of the ACM*, 9(1).
Winogard, Terry, 1972, *Understanding Natural Language*, Academic Press, New York.
Woods, William, 1977, 'Lunar Rocks in Natural English: Explorations in Natural Language Question-answering,' *Linguistic Structures Processing*, A. Zampolli (Ed.), Elsevier, North Holland.

EXERCISES

1. Differentiate between the rationalist and empiricist approaches to natural language processing.
2. List the motivation behind the development of computational models of languages.
3. Briefly discuss the meaning components of a language.
4. What makes natural language processing difficult?
5. What is the role of transformational rules in transformational grammar? Explain with the help of examples.

CHAPTER 2

LANGUAGE MODELLING

CHAPTER OVERVIEW

The domain of language is quite vast. It presents an almost infinite number of sentences to the reader (or computer). To handle such a large number of sentences, we have to create a model of the domain, which can subsequently be simplified and handled computationally. A number of language models have been proposed. We introduce some of these models in this chapter. To create a general model of any language is a difficult task. There are two approaches for language modelling. One is to define a grammar that can handle the language. The other is to capture the patterns in a grammar language statistically. This chapter has a mixed approach. It gives a glimpse of both grammar-based model and statistical language model. These include lexical functional grammar, government and binding, Paninian grammar, and *n*-gram based model.

2.1 INTRODUCTION

Why and how do we model a language? This question has been discussed by linguists since 500 BC. Computational linguists also have to confront this question. It is obvious that our purpose is to understand and generate natural languages from a computational viewpoint. One approach can be to just take a language, try to understand every word and sentence of it, and then come to a conclusion. This approach has not succeeded as there are difficulties at each stage, which we will understand as we go through this book. An alternative approach is to study the grammar of various languages, compare them, and if possible, arrive at reasonable models that facilitate our understanding of the problem and designing of natural-language tools.

A model is a description of some complex entity or process. A language model is thus a description of language. Indeed, natural language is a complex entity and in order to process it through a computer-based program, we need to build a representation (model) of it. This is known

Scanned by CamScanner