

UNIT V APPLICATIONS

AI applications – Language Models – Information Retrieval- Information Extraction – Natural Language Processing – Machine Translation – Speech Recognition – Robot – Hardware – Perception – Planning – Moving.

5.1 APPLICATIONS OF ARTIFICIAL INTELLIGENCE

- a) **Artificial Intelligence in Healthcare:** Companies are applying machine learning to make better and faster diagnoses than humans. One of the best-known technologies is IBM's Watson. It understands natural language and can respond to questions asked of it. The system mines patient data and other available data sources to form a hypothesis, which it then presents with a confidence scoring schema. AI is a study realized to emulate human intelligence into computer technology that could assist both, the doctor and the patients in the following ways:
- By providing a laboratory for the examination, representation and cataloguing medical information
 - By devising novel tool to support decision making and research
 - By integrating activities in medical, software and cognitive sciences
 - By offering a content rich discipline for the future scientific medical communities.
- b) **Artificial Intelligence in business:** Robotic process automation is being applied to highly repetitive tasks normally performed by humans. Machine learning algorithms are being integrated into analytics and CRM (Customer relationship management) platforms to uncover information on how to better serve customers. Chatbots have already been incorporated into websites and e companies to provide immediate service to customers. Automation of job positions has also become a talking point among academics and IT consultancies.
- c) **AI in education:** It automates grading, giving educators more time. It can also assess students and adapt to their needs, helping them work at their own pace.

d) AI in Autonomous vehicles: Just like humans, self-driving cars need to have sensors to understand the world around them and a brain to collect, processes and choose specific actions based on information gathered. Autonomous vehicles are with advanced tool to gather information, including long range radar, cameras, and LIDAR. Each of the technologies are used in different capacities and each collects different information. This information is useless, unless it is processed and some form of information is taken based on the gathered information. This is where artificial intelligence comes into play and can be compared to human brain. AI has several applications for these vehicles and among them the more immediate ones are as follows:

- Directing the car to gas station or recharge station when it is running low on fuel.
- Adjust the trips directions based on known traffic conditions to find the quickest route.
- Incorporate speech recognition for advanced communication with passengers.
- Natural language interfaces and virtual assistance technologies.

e) AI for robotics will allow us to address the challenges in taking care of an aging population and allow much longer independence. It will drastically reduce, may be even bring down traffic accidents and deaths, as well as enable disaster response for dangerous situations for example the nuclear meltdown at the fukushima power plant.

f) Cyborg Technology: One of the main limitations of being human is simply our own bodies and brains. Researcher Shimon Whiteson thinks that in the future, we will be able to augment ourselves with computers and enhance many of our own natural abilities. Though many of these possible cyborg enhancements would be added for convenience, others may serve a more practical purpose. Yoky Matsuka of Nest believes that AI will become useful for people with amputated limbs, as the brain will be able to communicate with a robotic limb to give the patient more control. This kind of cyborg technology would significantly reduce the limitations that amputees deal with daily.

5.1.1 Artificial Intelligence Technologies

The market for artificial intelligence technologies is flourishing. Artificial Intelligence involves a variety of technologies and tools, some of the recent technologies are as follows:

- **Natural Language Generation:** it's a tool that produces text from the computer data. Currently used in customer service, report generation, and summarizing business intelligence insights.
- **Speech Recognition:** Transcribes and transforms human speech into a format useful for computer applications. Presently used in interactive voice response systems and mobile applications.
- **Virtual Agent:** A Virtual Agent is a computer generated, animated, artificial intelligence virtual character (usually with anthropomorphic appearance) that serves as an online customer service representative. It leads an intelligent conversation with users, responds to their questions and performs adequate non-verbal behavior. An example of a typical Virtual Agent is Louise, the Virtual Agent of eBay, created by a French/American developer VirtuOz.
- **Machine Learning:** Provides algorithms, APIs (Application Program interface) development and training toolkits, data, as well as computing power to design, train, and deploy models into applications, processes, and other machines. Currently used in a wide range of enterprise applications, mostly involving prediction or classification.
- **Deep Learning Platforms:** A special type of machine learning consisting of artificial neural networks with multiple abstraction layers. Currently used in pattern recognition and classification applications supported by very large data sets.
- **Biometrics:** Biometrics uses methods for unique recognition of humans based upon one or more intrinsic physical or behavioral traits. In computer science, particularly, biometrics is used as a form of identity access management and access control. It is also used to identify individuals in groups that are under surveillance. Currently used in market research.
- **Robotic Process Automation:** using scripts and other methods to automate human action to support efficient business processes. Currently used where it is inefficient for humans to execute a task.
- **Text Analytics and NLP:** Natural language processing (NLP) uses and supports text analytics by facilitating the understanding of sentence structure and meaning, sentiment, and intent through statistical and machine learning methods. Currently used in fraud

detection and security, a wide range of automated assistants, and applications for mining unstructured data.

5.2 LANGUAGE MODEL

The goal of a language model is to assign a probability to a sequence of words by means of a probability distribution. Formal grammars (e.g. regular, context free) give a hard “binary” model of the legal sentences in a language. NLP is a probabilistic model of a language that gives a probability that a string is a member of a language or not. To specify a correct probability distribution, the probability of all sentences in a language must sum to 1.

5.2.1 Uses of Language Models

- Speech Recognition
- OCR & Handwriting Recognition
- Machine Translation
- Generation
- Context sensitive spelling correction.

A language model also supports predicting the completion of a sentence. Predictive text input systems can guess what is been typed and provide choices on how to complete it.

5.2.2 N- Gram Word Models

- This model is considered over sequences of words, characters, syllables or other units.
- Estimate probability of each word given prior context.
- An N-gram model uses only N-1 words of prior context.
 - ✓ Unigram: $P(\text{phone})$
 - ✓ Bigram: $P(\text{phone} \mid \text{cell})$
 - ✓ Trigram: $P(\text{phone} \mid \text{your cell})$
- The Markov assumption is the presumption that the future behavior of a dynamical system only depends on its recent history. In particular, in a Kth-Order Markov Model, next state only depends on the k most recent states, therefore an N – gram model is a (N-1) – order Markov model.

5.2.3 N-gram Character Models

- One of the simplest language models: $P(c_I^N)$
- Language identification: given the text determine which language it is written in.
- Build a trigram character model of each candidate language: $P(c_i \mid c_{i-2:i-1}, l)$

- Train and Test Corpora
 - ✓ A language model must be trained on a large corpus of text to estimate good parameter values.
 - ✓ Model can be evaluated based on its ability to predict a high probability for a disjoint test corpus.
 - ✓ The training corpus should be representative of the actual application data.
 - ✓ To handle words in the test corpus that did not occur in the training data an explicit symbol is used.
 - ✓ Symbol to represent unknown words (<UNK>)
 - ✓ **Perplexity** – Measure of how well a model “fits” the test data.

$$Perplexity(W_1^N) = \sqrt[N]{\frac{1}{P(w_1 w_2 \dots w_N)}}$$

- ✓ Smoothing - reassigns probability mass to unseen events.

5.3 INFORMATION RETRIVAL

Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers). Generically, “collections”, Less-frequently used, “corpora” are searched and “documents” namely web pages, PDFs, PowerPoint slides, paragraphs, etc. are retrieved. Information Retrieval system consists of a software program that facilitates a user in finding the information the user needs.

The Information Retrieval System was coined by Calvin Mooers in 1952. These information retrieval systems were, truly speaking, document retrieval system, since they were designed to retrieve information. Information retrieval deals with storage, organization and access to text, as well as multimedia information resources. Information Retrieval is a process of searching some collection of documents, using the term document in its widest sense, in order to identify those documents which deal with a particular subject. Any system that is designed to facilitate this literature searching may legitimately be called an information retrieval system.

Information retrieval systems originally meant text retrieval systems, since they were dealing with textual documents, modern information retrieval systems deal with multimedia

information comprising text, audio, images and video. While many features of conventional text retrieval system are equally applicable to multimedia information retrieval, the specific nature of audio, image and video information have called for the development of many new tools and techniques for information retrieval.

Modern information retrieval deals with storage, organization and access to text, as well as multimedia information resources. The concept of information retrieval presupposes that there are some documents or records containing information that have been organized in an order suitable for easy retrieval. The documents or records we are concerned with contain bibliographic information which is quite different from other kinds of information or data. We may take a simple example. If we have a database of information pertaining to an office, or a supermarket, all we have are the different kinds of records and related facts, like names of employees, their positions, salary, and so on, or in the case of a supermarket, names of different items, prices, quantity, and so on. The main objective of a bibliographic information retrieval system, however, is to retrieve the information either the actual information or the documents containing the information that fully or partially match the user's query. The database may contain abstracts or full texts of document, like newspaper articles, handbooks, dictionaries, encyclopedias, legal documents, statistics, etc., as well as audio, images, and video information.

An information retrieval system thus has three major components- the document subsystem, the users subsystem, and the searching/retrieval subsystem. These divisions are quite broad and each one is designed to serve one or more functions, such as:

- Analysis of documents and organization of information (creation of a document database)
- Analysis of user's queries, preparation of a strategy to search the database
- Actual searching or matching of users queries with the database, and finally
- Retrieval of items that fully or partially match the search statement.

An IR is a 3 step Process:

- Asking a question (how to use the language to get what we want?)
- Building an answer from known data. (How to refer to a given text?)
- Assessing the answer. (Does it contain the information we are seeking.)

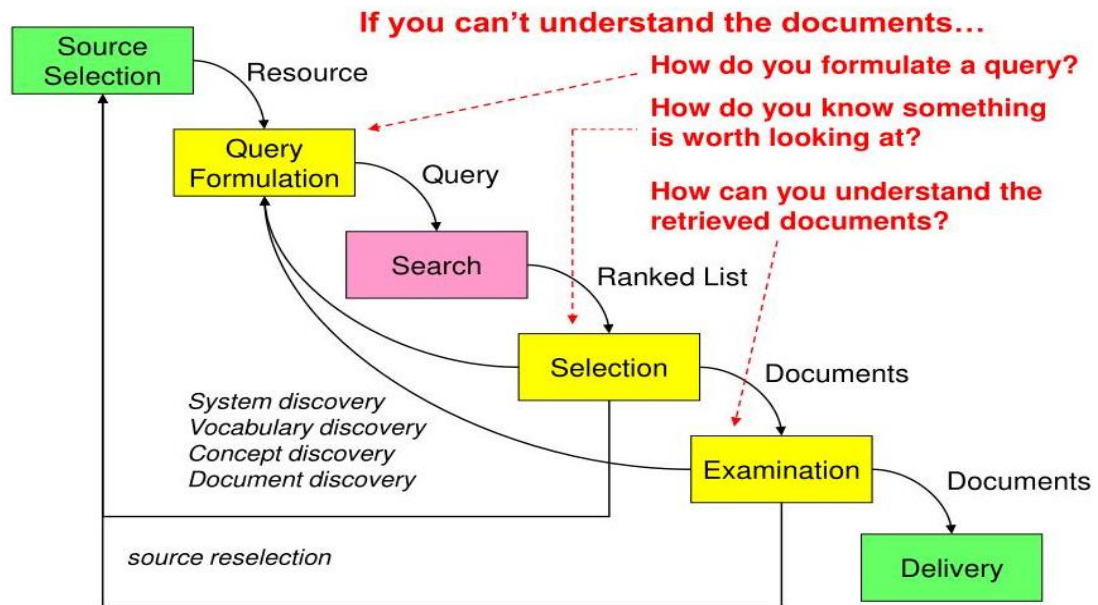
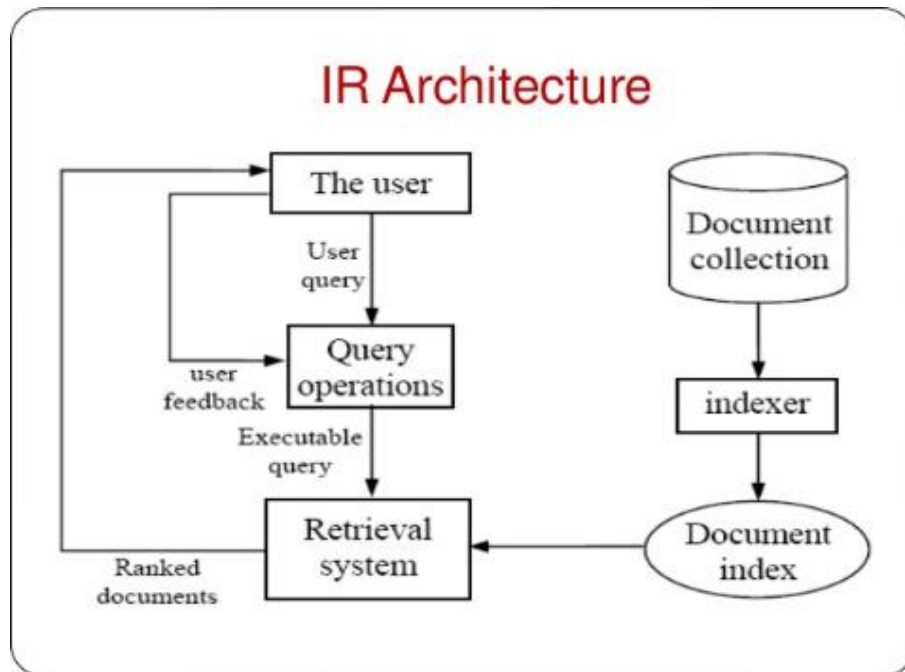


Fig: The Information Retrieval Cycle



5.3.1 IR System Components

- Text Operations forms index words (tokens).
 - ✓ Stop word removal
 - ✓ Stemming
- Indexing constructs an inverted index of word to document pointers.
- Searching retrieves documents that contain a given query token from the inverted index.
- Ranking scores all retrieved documents according to a relevance metric.
- User Interface manages interaction with the user:
 - ✓ Query input and document output.
 - ✓ Relevance feedback.
 - ✓ Visualization of results.
- Query Operations transform the query to improve retrieval:
 - ✓ Query expansion using a thesaurus.
 - ✓ Query transformation using relevance feedback.

5.3.2 Purpose of Information Retrieval System

An information retrieval system is designed to retrieve the documents or information required by the user community. It should make the right information available to the right user. Thus, an information retrieval system aims at collecting and organizing information in one or

more subject areas in order to provide it to the user as soon asked for. Belkin presents the following situation which clearly reflects the purpose of information retrieval systems:

- A writer presents a set of ideas in a document using a set of concepts
- Somewhere there will be some users who require the ideas but may not be able to identify those. In other words, there will be some persons who lack the ideas put forward by the author in his/her work.
- Information retrieval system serve to match the writers ideas expressed in the document with the user requirements or demand for those.
- Thus, an information retrieval system serves as a bridge between the world of creators or generators of information and the users of that information.

Some terminology

- An IR system looks for data matching using some criteria defined by the users in their queries.
- The language used to ask a question is called the query language.
- These queries use keywords (atomic items characterizing some data).
- The basic unit of data is a document (can be a file, an article, a paragraph, etc.).
- A document corresponds to free text (may be unstructured).
- All the documents are gathered into a collection (or corpus).

Example:

1 million documents, each counting about 1000 words

if each word is encoded using 6 bytes:

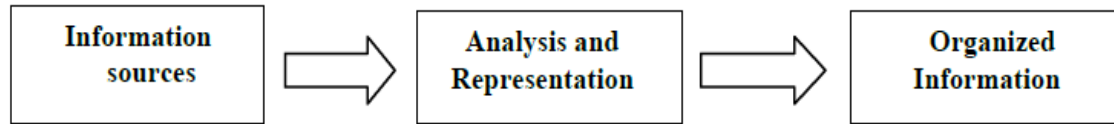
$$10^6 \times 1000 \times 6 / 1024 \approx 6\text{GB}$$

5.3.3 Components of Information Retrieval

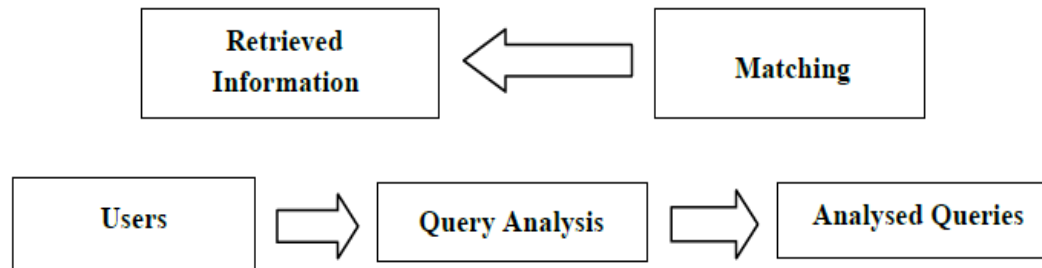
In an information retrieval system there are the documents or sources of information on one side and on the other there are the user's queries. These two sides are linked through a series of tasks. Lancaster mentions that an information retrieval system comprises six major subsystems:

- ✓ The document subsystem
- ✓ The indexing subsystem
- ✓ The vocabulary subsystem
- ✓ The searching subsystem
- ✓ The service-system interface, and

- ✓ The matching subsystem



The broad outline of an information retrieval system



Three major components of IRS

- 1) Document subsystem
 - a) Acquisition
 - b) Representation
 - c) File organization
- 2) User sub system
 - a) Problem
 - b) Representation
 - c) Query
- 3) Searching /Retrieval subsystem
 - a) Matching
 - b) Retrieved objects

5.3.4 Kinds of Information Retrieval Systems

Two broad categories of information retrieval system can be identified: in- house and online.

In- house information retrieval systems are set up by a particular library or information center to serve mainly the users within the organization. One particular type of in-house database is the library catalogue. Online public access catalogues (OPACs) provide facilities for library users to carry out online catalogue searches, and then to check the availability of the item

required. Online IR is nothing but retrieving data from web sites, web pages and servers that may include data bases, images, text, tables, and other types.

5.3.4 Functions of information retrieval system

An information retrieval system deals with various sources of information on the one hand and user's requirements on the other. It must:

- ✓ Analyze the contents of the sources of information as well as the user's queries, and then
- ✓ Match these to retrieve those items that are relevant

The major functions of an information retrieval system can be listed as follows:

- ✓ To identify the information (sources) relevant to the areas of interest of the target users community
- ✓ To analyze the contents of the sources (documents)
- ✓ To represent the contents of the analyzed sources in a way that will be suitable for matching user's queries
- ✓ To analyze user's queries and to represent them in a form that will be suitable for matching with the database
- ✓ To match the search statement with the stored database
- ✓ To retrieve the information that is relevant, and
- ✓ To make necessary adjustments in the system based on feedback form the users.

5.3.5 Features of an information retrieval system

- ✓ An effective information retrieval system must have provisions for:
- ✓ Prompt dissemination of information
- ✓ Filtering of information
- ✓ The right amount of information at the right time
- ✓ Active switching of information
- ✓ Receiving information in an economical way
- ✓ Browsing
- ✓ Getting information in an economical way
- ✓ Current literature
- ✓ Access to other information systems
- ✓ Interpersonal communications, and
- ✓ Personalized help.

5.3.6 Indexing usually consists of the several phases

- ✓ After word segmentation, stop words are removed.
- ✓ These common words like articles or prepositions contain little meaning by themselves and are ignored in the document representation.
- ✓ Second, word forms are transformed into their basic form, the stem.
- ✓ During the stemming phase, e.g. houses would be transformed into house.
- ✓ For the document representation, different word forms are usually not necessary.
- ✓ The importance of a word for a document can be different.
- ✓ Some words better describe the content of a document than others.
- ✓ This weight is determined by the frequency of a stem within the text of a document.

In multimedia retrieval, the context is essential for the selection of a form of query and document representation. Different media representations may be matched against each other or transformations may become necessary (e.g. to match terms against pictures or spoken language utterances against documents in written text).

As information retrieval needs to deal with vague knowledge, exact processing methods are not appropriate.

- Vague retrieval models like the probabilistic model are more suitable.
- Within these models, terms are provided with weights corresponding to their importance for a document.
- These weights mirror different levels of relevance.

The result of current information retrieval systems are usually sorted lists of documents where the top results are more likely to be relevant according to the system.

- In some approaches, the user can judge the documents returned to him and tell the systems which ones are relevant for user.
- The system then resorts the result set.
- Documents which contain many of the words present in the relevant documents are ranked higher.
- This relevance feedback process is known to greatly improve the performance.
- Relevance feedback is also an interesting application for machine learning.
- Based on a human decisions, the optimization step can be modeled with several approaches, e.g. with rough sets.

- In Web environments, a click is often interpreted as an implicit positive relevance judgment

5.4 INFORMATION EXTRACTION

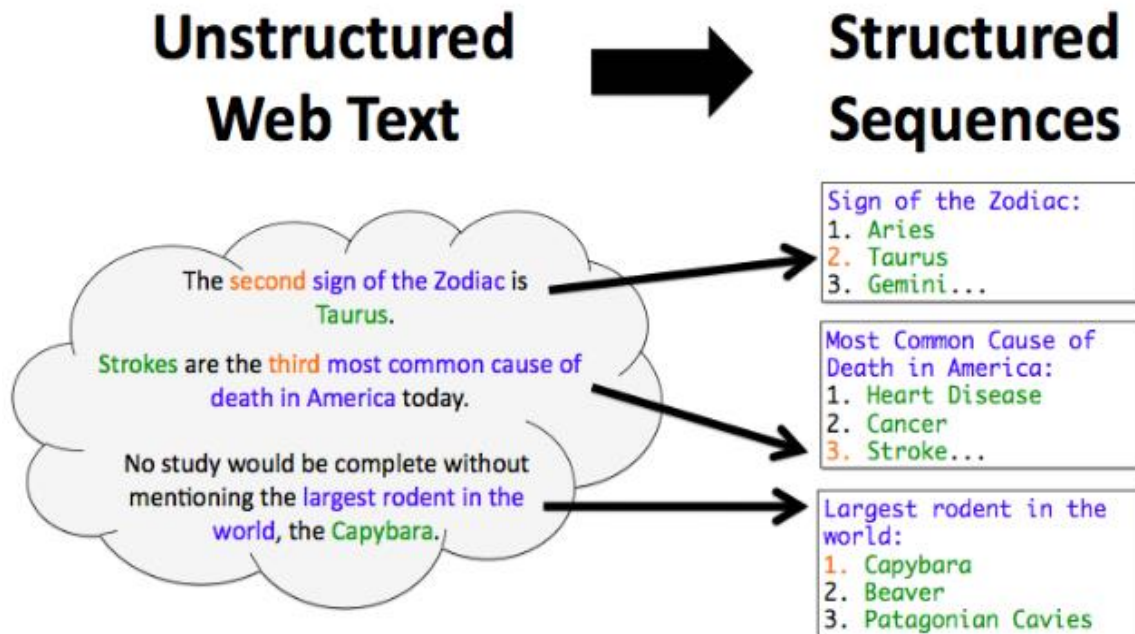
Information extraction (IE) is the automated retrieval of specific information related to a selected topic from a body or bodies of text. Information extraction is the process of extracting specific (pre-specified) information from textual sources. One of the most trivial examples is when your email extracts only the data from the message for you to add in your Calendar.

Other free-flowing textual sources from which information extraction can distill structured information are legal acts, medical records, social media interactions and streams, online news, government documents, corporate reports and more.

Information extraction tools make it possible to pull information from text documents, databases, websites or multiple sources. IE may extract info from unstructured, semi-structured or structured, machine-readable text. Usually, however, IE is used in natural language processing (NLP) to extract structured from unstructured text.

Information extraction depends on,

- Named entity recognition (NER), a sub-tool used to find targeted information to extract.
- NER recognizes entities first as one of several categories such as location (LOC), persons (PER) or organizations (ORG).
- Once the information category is recognized, an information extraction utility extracts the named entity's related information and constructs a machine-readable document from it, which algorithms can further process to extract meaning.
- IE finds meaning by way of other subtasks including co-reference resolution, relationship extraction, language and vocabulary analysis and sometimes audio extraction.
- Current efforts in multimedia document processing in IE include automatic annotation and content recognition and extraction from images and video could be seen as IE as well.
- Because of the complexity of language, high-quality IE is a challenging task for artificial intelligence (AI) systems.



Typically, for structured information to be extracted from unstructured texts, the following main subtasks are involved:

Pre-processing of the text – this is where the text is prepared for processing with the help of computational linguistics tools such as tokenization, sentence splitting, morphological analysis, etc.

Finding and classifying concepts – this is where mentions of people, things, locations, events and other pre-specified types of concepts are detected and classified.

Connecting the concepts – this is the task of identifying relationships between the extracted concepts.

Unifying – this subtask is about presenting the extracted data into a standard form.

Getting rid of the noise – this subtask involves eliminating duplicate data.

Enriching your knowledge base – this is where the extracted knowledge is ingested in your database for further use.

5.4.1 Information Extraction Architecture

The below figure shows the architecture for a simple information extraction system. At first, the raw text of the document is split into sentences using a sentence segmenter, and each sentence is further subdivided into words using a tokenizer. Next, each sentence is tagged with part-of-speech tags, which will prove very helpful in the next step, **named entity detection**. In

this step, we search for mentions of potentially interesting entities in each sentence. Finally, we use **relation detection** to search for likely relations between different entities in the text.

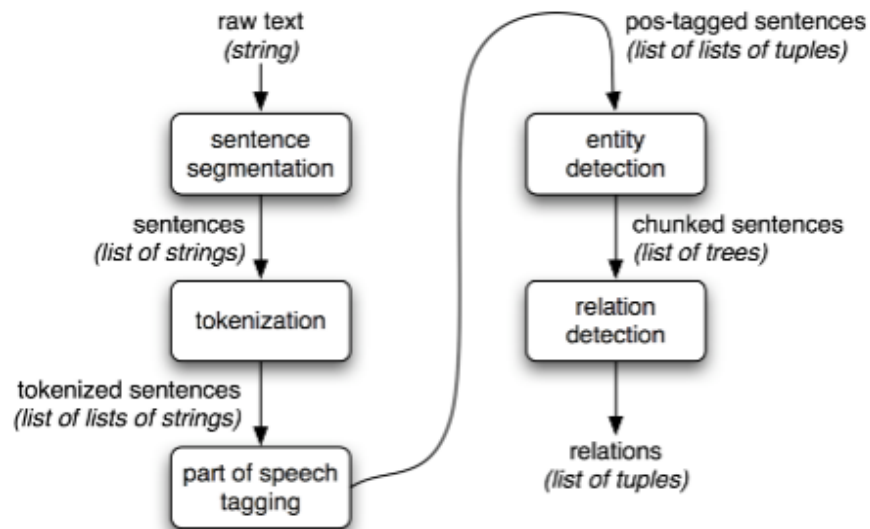


Figure : Simple Pipeline Architecture for an Information Extraction System. This system takes the raw text of a document as its input, and generates a list of (entity, relation, entity) tuples as its output.

5.4.2 Applications of IE

- Enterprise
- News tracking
- Customer care
- Data cleaning
- Personal information management
- Scientific applications
- Web oriented applications
- Citation databases
- Opinion databases
- Community websites
- Comparison shopping
- Ad placement on webpages
- Structured web searches

5.5 NATURAL LANGUAGE PROCESSING

The field of study that focuses on the interactions between human language and computers is called Natural Language Processing, or NLP for short. It sits at the intersection of computer science, artificial intelligence, and computational linguistics. Natural Language Processing is a field that covers computer understanding and manipulation of human language, and it's ripe with possibilities for news gathering.

NLP is used to analyze text, allowing machines to understand how human's speak. This human-computer interaction enables real-world applications like automatic text summarization, sentiment analysis, topic extraction, named entity recognition, parts-of-speech tagging, relationship extraction, stemming, and more. NLP is commonly used for text mining, machine translation, and automated question answering.

NLP is characterized as a difficult problem in computer science. Human language is rarely precise, or plainly spoken. To understand human language is to understand not only the words, but the concepts and how they're linked together to create meaning. Despite language being one of the easiest things for the human mind to learn, the ambiguity of language is what makes natural language processing a difficult problem for computers to master. NLP algorithms have a variety of uses. Basically, they allow developers to create software that understands human language. Due to the complicated nature of human language, NLP can be difficult to learn and implement correctly.

In fact, a typical interaction between humans and machines using Natural Language Processing could go as follows:

1. A human talks to the machine
2. The machine captures the audio
3. Audio to text conversion takes place
4. Processing of the text's data
5. Data to audio conversion takes place
6. The machine responds to the human by playing the audio file

5.5.1 Challenge of Natural Language

- Working with natural language data is not solved.
- It has been studied for half a century, and it is really hard.
- Natural language is primarily hard because it is messy.

Natural Language Processing is **the driving force behind the following common applications:**

- Language translation applications such as Google Translate
- Word Processors such as Microsoft Word and Grammarly that employ NLP to check grammatical accuracy of texts.
- Interactive Voice Response (IVR) applications used in call centers to respond to certain users' requests.
- Personal assistant applications such as OK Google, Siri, Cortana, and Alexa.

5.5.2 NLP Terminology

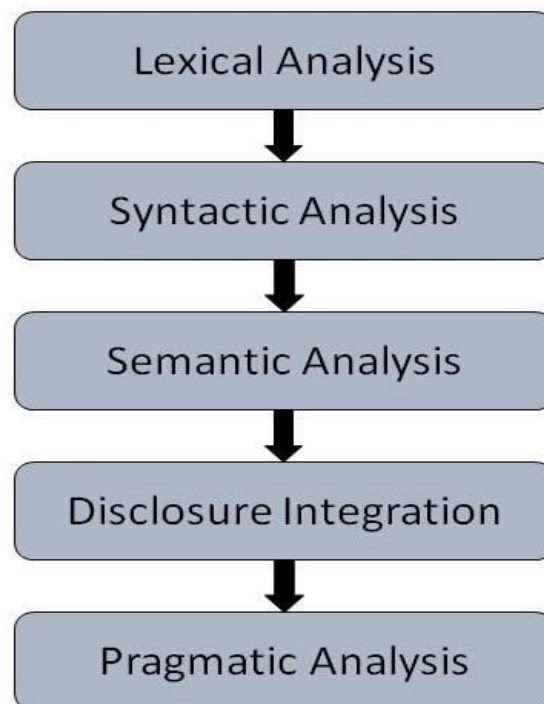
- **Phonology** – It is study of organizing sound systematically.
- **Morphology** – It is a study of construction of words from primitive meaningful units.
- **Morpheme** – It is primitive unit of meaning in a language.
- **Syntax** – It refers to arranging words to make a sentence. It also involves determining the structural role of words in the sentence and in phrases.
- **Semantics** – It is concerned with the meaning of words and how to combine words into meaningful phrases and sentences.
- **Pragmatics** – It deals with using and understanding sentences in different situations and how the interpretation of the sentence is affected.
- **Discourse** – It deals with how the immediately preceding sentence can affect the interpretation of the next sentence.
- **World Knowledge** – It includes the general knowledge about the world.

5.5.3 Steps in NLP

There are general five steps:

- 1) **Lexical Analysis** – It involves identifying and analyzing the structure of words. Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.
- 2) **Syntactic Analysis (Parsing)** – Syntax refers to the arrangement of words in a sentence such that they make grammatical sense. In NLP, syntactic analysis is used to assess how the natural language aligns with the grammatical rules. Computer algorithms are used to apply grammatical rules to a group of words and derive meaning from them. Here are some syntax techniques that can be used:

- **Lemmatization:** It entails reducing the various inflected forms of a word into a single form for easy analysis.
- **Morphological segmentation:** It involves dividing words into individual units called morphemes.
- **Word segmentation:** It involves dividing a large piece of continuous text into distinct units.
- **Part-of-speech tagging:** It involves identifying the part of speech for every word.
- **Parsing:** It involves undertaking grammatical analysis for the provided sentence.
- **Sentence breaking:** It involves placing sentence boundaries on a large piece of text.
- **Stemming:** It involves cutting the inflected words to their root form.



3) **Semantic Analysis** – It draws the exact meaning or the dictionary meaning from the text. The text is checked for meaningfulness. It is done by mapping syntactic structures and objects in the task domain. The semantic analyzer disregards sentence such as “hot ice-cream”. Semantics refers to the meaning that is conveyed by a text. Semantic analysis is one of the difficult aspects of Natural Language Processing that has not been fully resolved yet. It involves

applying computer algorithms to understand the meaning and interpretation of words and how sentences are structured. Here are some techniques in semantic analysis:

- ✓ **Named entity recognition (NER):** It involves determining the parts of a text that can be identified and categorized into preset groups. Examples of such groups include names of people and names of places.
- ✓ **Word sense disambiguation:** It involves giving meaning to a word based on the context.
- ✓ **Natural language generation:** It involves using databases to derive semantic intentions and convert them into human language.
- ✓ **Discourse Integration** – The meaning of any sentence depends upon the meaning of the sentence just before it. In addition, it also brings about the meaning of immediately succeeding sentence.
- ✓ **Pragmatic Analysis** – During this, what was said is re-interpreted on what it actually meant. It involves deriving those aspects of language which require real world knowledge.

5.5.4 Implementation Aspects of Syntactic Analysis

There are a number of algorithms researchers have developed for syntactic analysis, but we consider only the following simple methods –

- Context-Free Grammar
- Top-Down Parser

Context-Free Grammar

It is the grammar that consists rules with a single symbol on the left-hand side of the rewrite rules. Let us create grammar to parse a sentence.

“The bird pecks the grains”

Articles (DET) – a | an | the

Nouns – bird | birds | grain | grains

Noun Phrase (NP) – Article + Noun | Article + Adjective + Noun

= DET N | DET ADJ N

Verbs – pecks | pecking | pecked

Verb Phrase (VP) – NP V | V NP

Adjectives (ADJ) – beautiful | small | chirping

The parse tree breaks down the sentence into structured parts so that the computer can easily understand and process it. In order for the parsing algorithm to construct this parse tree, a set of rewrite rules, which describe what tree structures are legal, need to be constructed. These rules say that a certain symbol may be expanded in the tree by a sequence of other symbols. According to first order logic rule, if there are two strings Noun Phrase (NP) and Verb Phrase (VP), then the string combined by NP followed by VP is a sentence. The rewrite rules for the sentence are as follows:

$S \rightarrow NP VP$

$NP \rightarrow DET N \mid DET ADJ N$

$VP \rightarrow V NP$

Lexocon:

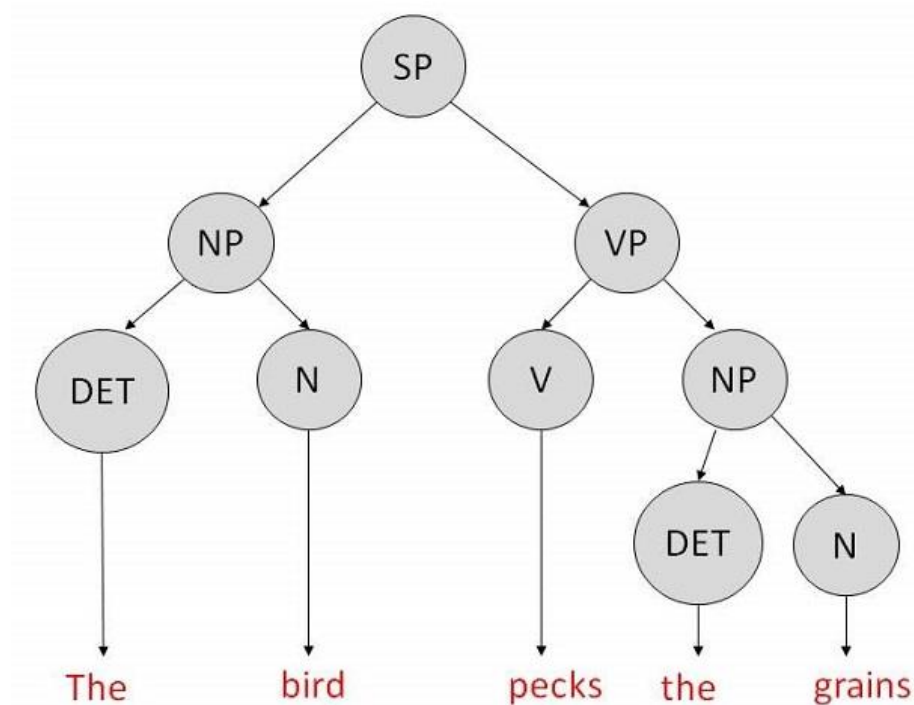
$DET \rightarrow a \mid the$

$ADJ \rightarrow beautiful \mid perching$

$N \rightarrow bird \mid birds \mid grain \mid grains$

$V \rightarrow peck \mid pecks \mid pecking$

The parse tree can be created as shown:



Now consider the above rewrite rules. Since V can be replaced by both, "peck" or "pecks", sentences such as "The birds pecks the grains" can be wrongly permitted. i. e. the subject-verb agreement error is approved as correct.

Merit:

- The simplest style of grammar, therefore widely used one.

Demerits:

- They are not highly precise. For example, "The grains peck the bird", is a syntactically correct according to parser, but even if it makes no sense, parser takes it as a correct sentence.
- To bring out high precision, multiple sets of grammar need to be prepared. It may require a completely different sets of rules for parsing singular and plural variations, passive sentences, etc., which can lead to creation of huge set of rules that are unmanageable.

Top-Down Parser

Here, the parser starts with the S symbol and attempts to rewrite it into a sequence of *terminal symbols* that matches the classes of the words in the input sentence until it consists entirely of terminal symbols. These are then checked with the input sentence to see if it matched. If not, the process is started over again with a different set of rules. This is repeated until a specific rule is found which describes the structure of the sentence.

Merit:

- It is simple to implement.

Demerits:

- It is inefficient, as the search process has to be repeated if an error occurs.
- Slow speed of working

5.6 MACHINE TRANSLATION

Machine translation is the automatic translation of text from one natural language (the source) to another (the target). It was one of the first application areas envisioned for computers, but it is only in the past decade that the technology has seen widespread usage. Here is a sentence from this book: “AI is one of the *newest fields* in science and engineering.”

And here it is translated from English to Tamil by an online tool, Google Translate:

AI அறிவியல் மற்றும் பொறியியலில் புதிய துறைகளில் ஒன்றாகும்.

For those who don’t read Tamil, here is the Tamil translated back to English. The words that came out different are in italics: “AI is one of the *new disciplines* in science and engineering.”

The differences are all reasonable paraphrases, such as *frequently mentioned* for *regularly cited*. This is typical accuracy: of the two sentences, one has an error that would not be made by a native speaker, yet the meaning is clearly conveyed.

5.6.1 Types of Translation

Historically, there have been three main applications of machine translation. *Rough translation*, as provided by free online services, gives the “gist” of a foreign sentence or document, but contains errors. *Pre-edited translation* is used by companies to publish their documentation and sales materials in multiple languages. The original source text is written in a constrained language that is easier to translate automatically, and the results are usually edited by a human to correct any errors. *Restricted-source translation* works fully automatically, but only on highly stereotypical language, such as a weather report.

Translation is difficult because, in the fully general case, it requires in-depth understanding of the text. This is true even for very simple texts even “texts” of one word. Consider the word “Open” on the door of a store. It communicates the idea that the store is accepting customers at the moment. Now consider the same word “Open” on a large banner outside a newly constructed store. It means that the store is now in daily operation, but readers of this sign would not feel misled.

The problem is that different languages categorize the world differently. For example, the French word “doux” covers a wide range of meanings corresponding approximately to the English words “soft,” “sweet,” and “gentle.” A translator (human or machine) often needs to understand the actual situation described in the source, not just the individual words. For example, to translate the English word “him,” into Tamil, a choice must be made between the humble and honorific form, a choice that depends on the social relationship between the speaker and the referent of “him.”

5.6.2 Machine translation systems

Some systems attempt to analyze the source language text all the way into internal knowledge representation and then generate sentences in the target language from that representation. This is difficult because it involves three unsolved problems:

- creating a complete knowledge representation of everything;
- parsing into that representation; and
- generating sentences from that representation.

Other systems are based on a transfer model. They keep a database of translation rules, and whenever the rule matches, they translate directly, at lexical, syntactic, or semantic level.

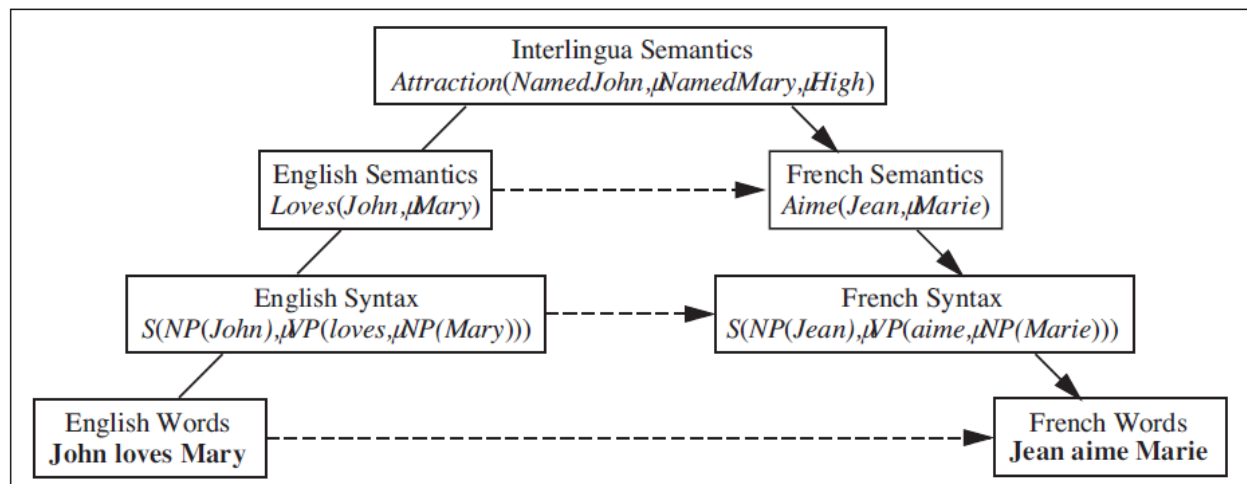


Figure 23.12 The Vauquois triangle: schematic diagram of choices for machine translation system

5.6.3 Statistical machine translation

Statistical machine translation needs data sample translations from which a translation model can be learned. To translate a sentence in, say, English (e) into French (f), we find the string of words f^* that maximizes.

$$f^* = \operatorname{argmax}_f P(f | e) = \operatorname{argmax}_f P(e | f) P(f)$$

Here the factor $P(f)$ is the target **language model** for French; it says how probable a given sentence is in French. $P(e|f)$ is the **translation model** based on Baye's rule; it says how probable an English sentence is as a translation for a given French sentence. Similarly, $P(f | e)$ is a translation model from English to French.

Baye's rule translation is applicable in few domains. But statistical machine translation optimizes using a more sophisticated model that takes into account many of the features from the language model. The translation model is learned from a **bilingual corpus**—a collection of parallel texts, each an English/French pair. Now, if we had an infinitely large corpus, then translating a sentence would just be a lookup task. But of course our resources are finite, and most of the sentences we will be asked to translate will be novel. Most sentences composed of **phrases**. Translation is a matter of three steps:

1. Break the English sentence into phrases.
2. For each phrase, choose a corresponding French phrase. We use the notation $P(f_i | e_i)$ for the phrasal probability that f_i is a translation of e_i .
3. Choose a permutation of the phrases. For each f_i , choose a **distortion** d_i , which is the number of words that phrase f_i has moved with respect to e_i .

$$P(f, d | e) = \prod_i P(f_i | e_i) P(d_i)$$

5.6.4 Translation Procedure

1. **Find parallel texts:** First, gather a parallel bilingual corpus.
2. **Segment into sentences:** The unit of translation is a sentence, so we will have to break the corpus into sentences.
3. **Align sentences:** For each sentence in the English version, determine what sentence(s) it corresponds to in the French version. It is possible the order of two sentences need to be swapped, so align them.
4. **Extract distortions:** Once we have an alignment of phrases we can define distortion probabilities.
5. **Improve estimates with EM:** Compute the best alignments with the current values of these parameters in the E step, then update the estimates in the M step and iterate the process until convergence.

5.7 SPEECH RECOGNITION

Speech recognition is the task of identifying a sequence of words uttered by a speaker, given the acoustic signal. Speech recognition is difficult because the sounds made by a speaker are ambiguous and, well, noisy.

Challenges

1. **Segmentation:** written words in English have spaces between them, but in fast speech there are no pauses.
2. **Coarticulation:** when speaking quickly the “s” sound at the end of “nice” merges with the “b” sound at the beginning of beach,” yielding something that is close to a “sp.”
3. **Homophones:** words like “to,” “too,” and “two” that sound the same but differ in meaning.

Speech recognition is the problem of computing the most likely sequence of state variables, $x_{1:t}$, given a sequence of observations $e_{1:t}$.

$$\operatorname{argmax}_{word_{1:t}} P(word_{1:t} | sound_{1:t}) = \operatorname{argmax}_{word_{1:t}} P(sound_{1:t} | word_{1:t}) P(word_{1:t})$$

Here $P(sound_{1:t}|word_{1:t})$ is the **acoustic model**. It describes the sounds of words such as “ceiling” that begins with a soft “c” and sounds the same as “sealing.” $P(word_{1:t})$ is known as the **language model**. It specifies the prior probability of each utterance—for example, that “ceiling fan” is about 500 times more likely as a word sequence than “sealing fan.”

This approach was named the **noisy channel model** by Claude Shannon (1948). He described a situation in which an original message (the *words* in our example) is transmitted over a noisy channel (such as a telephone line) such that a corrupted message (the *sounds* in our example) are received at the other end. Shannon showed that no matter how noisy the channel, it is possible to recover the original message with arbitrarily small error, if we encode the original message in a redundant enough way.

5.7.1 Acoustic model

Sound waves are periodic changes in pressure that propagate through the air. When these waves strike the diaphragm of a microphone, the back-and-forth movement generates an electric current. An analog-to-digital converter measures the size of the current, which approximates the amplitude of the sound wave at discrete intervals called the **sampling rate**.

Speech sounds, which are mostly in the range of 100 Hz to 1000 Hz, are typically sampled at a rate of 8 kHz. The precision of each measurement is determined by the quantization factor. We only need to distinguish between different speech sounds. Linguists have identified about 100 speech sounds, or **phones**, that can be composed to form all the words in all known human languages. Roughly speaking, a phone is the sound that corresponds to a single vowel or consonant, but there are some complications: combinations of letters, such as “th” and “ng” produce single phones, and some letters produce different phones in different contexts (e.g., the “a” in *rat* and *rate*).

Let us see a brief overview of the features in a typical system. First, a Fourier transform is used to determine the amount of acoustic energy at about a dozen frequencies. Then we compute a measure called the **mel frequency cepstral coefficient (MFCC)** or MFCC for each frequency.

We also compute the total energy in the frame (signal over a time slice). That gives thirteen features; for each one we compute the difference between this frame and the previous frame, and the difference between differences, for a total of 39 features. These are continuous-valued; the easiest way to fit them into the HMM (Hidden Markov Model) framework is to discretize the values.

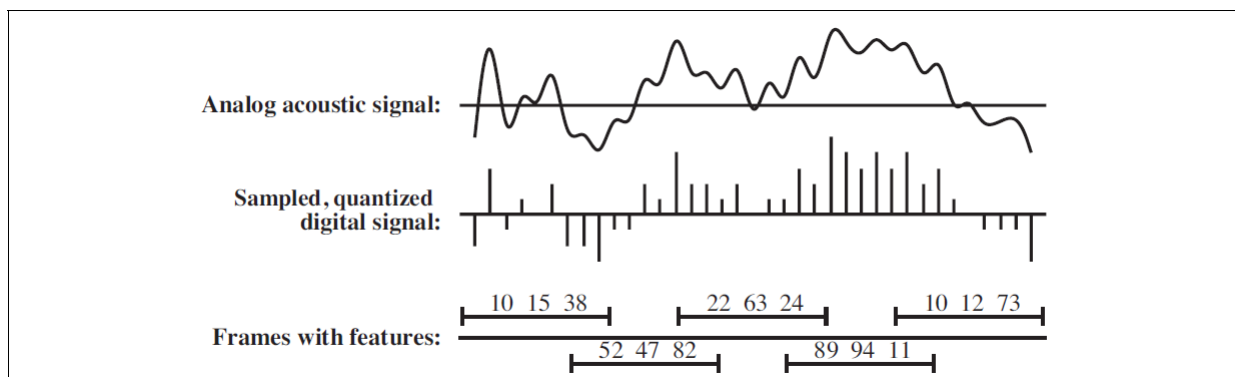


Figure 23.15 shows the sequence of transformations from the raw sound to a sequence of frames with discrete features.

5.7.2 Language model

For general-purpose speech recognition, the language model can be an n-gram model of text learned from a corpus of written sentences. However, spoken language has different characteristics than written language, so it is better to get a corpus of transcripts of spoken

language. For task-specific speech recognition, the corpus should be task-specific: to build your airline reservation system, get transcripts of prior calls. It also helps to have task-specific vocabulary, such as a list of all the airports and cities served, and all the flight numbers.

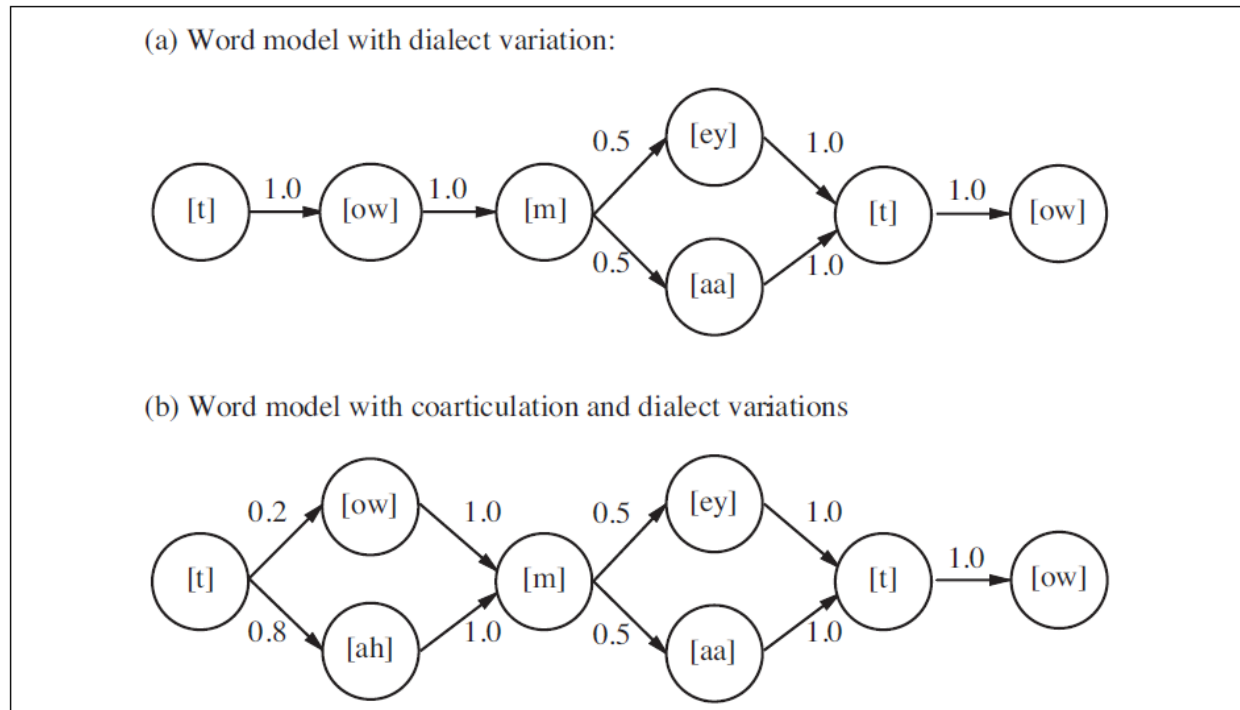


Figure 23.17 Two pronunciation models of the word “tomato.”

5.7.3 Building a speech recognizer

The quality of a speech recognition system depends on the quality of all of its components the language model, the word-pronunciation models, the phone models, and the signal processing algorithms used to extract spectral features from the acoustic signal. As usual, we will acquire the probabilities from a corpus, this time a corpus of speech. These probabilities can be viewed as *uncertain labels*.

From the uncertain labels, we can estimate new transition and sensor probabilities, and the EM procedure repeats. The method is guaranteed to increase the fit between model and data on each iteration, and it generally converges to a much better set of parameter values than those provided by the initial, hand-labeled estimates.

5.8 ROBOTICS

Robots are physical agents that perform tasks by manipulating the physical world. To do so, they are equipped with **effectors** such as legs, wheels, joints, and grippers. Robots are also equipped with **sensors**, which allow them to perceive their environment, including cameras and lasers to measure the environment, and gyroscopes and accelerometers to measure the robot's own motion. Most of today's robots fall into one of three primary categories.

1. **Manipulators**, or **robot arms** (Figure 25.1(a)), are physically anchored to their workplace, in a factory assembly line. Manipulator motion usually involves a chain of controllable joints, enabling robots to place effectors in any position within workplace.
2. **Mobile robots** move about their environment using wheels, legs, or similar mechanisms. They have been put to use delivering food in hospitals, moving containers at loading docks, and similar tasks. **Unmanned ground vehicles**, or UGVs, drive autonomously on streets, highways, and off-road. The **planetary rover** explores planets.
3. The third type of robot combines mobility with manipulation, and is often called a **mobile manipulator**. Humanoid robots mimic the human torso. Figure 25.1(b) shows two humanoid robots, both manufactured by Honda Corp. in Japan.

The **limitations** of robotics are:

1. Real robots must cope with environments that are partially observable, stochastic, dynamic, and continuous.
2. Many robot environments are sequential and multi-agent.
3. Robot cameras cannot see around corners, and motion commands are subject to uncertainty due to gears slipping, friction, etc.
4. The real world stubbornly refuses to operate faster than real time. In a simulated environment, it is possible to use simple algorithms to learn in a few CPU hours from millions of trials. In a real environment, it might take years to run these trials.
5. Furthermore, real crashes really hurt, unlike simulated ones.
6. Practical robotic systems need to embody prior knowledge about the robot, its physical environment, and the tasks that the robot will perform so that the robot can learn quickly and perform safely.

5.9 ROBOT HARDWARE

The agent architecture consists of sensors, effectors, and processors. The success of real robots depends as much on the design of sensors and effectors that are appropriate for the task.

Sensors

Sensors are the perceptual interface between robot and environment. **Passive sensors**, such as cameras, are true observers of the environment. They capture signals that are generated by other sources in the environment. **Active sensors**, such as sonar, send energy into the environment. **Range finders** are sensors that measure the distance to nearby objects. In the early days of robotics, robots were commonly equipped with **sonar sensors**. Sonar sensors emit directional sound waves, which are reflected by objects, with some of the sound making it back into the sensor. The time and intensity of the returning signal indicates the distance to nearby objects.

A second important class of sensors is **location sensors**. The **Global Positioning System** (GPS) measures the distance to satellites that emit pulsed signals. GPS receivers can recover the distance to these satellites by analyzing phase shifts. By triangulating signals from multiple satellites receivers can determine their absolute location on Earth to within a few meters. Third is **imaging sensors**, the cameras provide us with images of the environment using computer vision techniques.

The fourth important class is **proprioceptive sensors**, which inform the robot of its own motion. To measure the exact configuration of a robotic joint, motors are often equipped with **shaft decoders** that count the revolution of motors in small increments. On mobile robots, shaft decoders that report wheel revolutions can be used for **odometry**—the measurement of distance traveled. Unfortunately, wheels tend to drift and slip, so odometry is accurate only over short distances.

Other important aspects of robot state are measured by **force sensors** and **torque sensors**. These are indispensable when robots handle fragile objects or objects whose exact shape and location is unknown.

Effectors

Effectors are the means by which robots move and change the shape of their bodies. We count one **degree of freedom** for each independent direction in which a robot, or one of its effectors, can move. For example, a rigid mobile robot such as an AUV has six degrees of freedom, three for its (x, y, z) location in space and three for its angular orientation, known as

yaw, *roll*, and *pitch*. These six degrees define the **kinematic state** or **pose** of the robot. The arm in Figure 25.4(a) has exactly six degrees of freedom, created by five **revolute joints** that generate rotational motion and one **prismatic joint** that generates sliding motion.

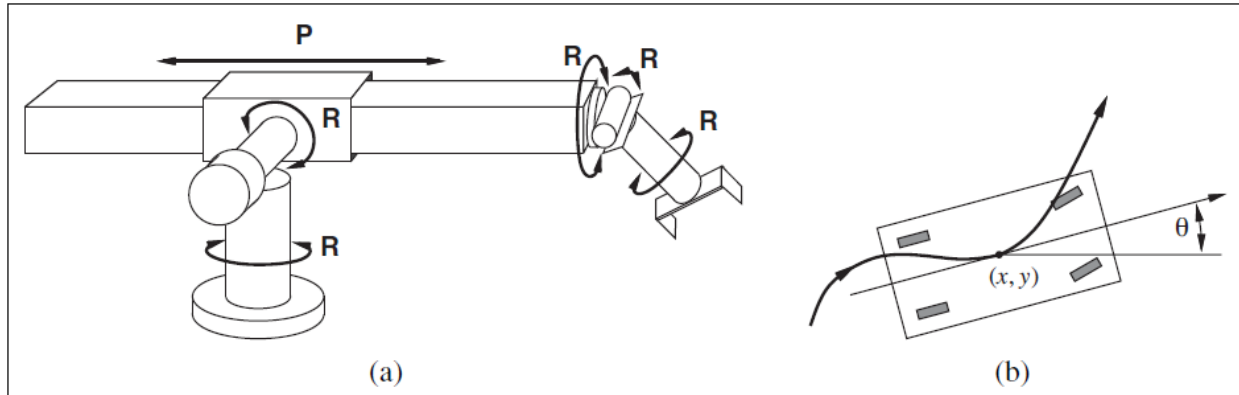


Figure 25.4 (a) The Stanford Manipulator with six degrees of freedom.

(b) Motion of a non-holonomic-four-wheeled vehicle with front-wheel steering.

The car has three **effective degrees of freedom** but two **controllable degrees of freedom**. We say a robot is **non-holonomic** if it has more effective DOFs than controllable DOFs and **holonomic** if the two numbers are the same. Holonomic robots are easier to control, i.e., it would be much easier to park a car that could move sideways as well as forward and backward—but holonomic robots are also mechanically more complex.

Differential drive robots possess two independently actuated wheels (or tracks), one on each side, as on a military tank. If both wheels move at the same velocity, the robot moves on a straight line. If they move in opposite directions, the robot turns on the spot. An alternative is the **synchro drive**, in which each wheel can move and turn around its own axis.

Legged robots have been made to walk, run, and even hop. This robot is **dynamically stable**, meaning that it can remain upright while hopping around. A robot that can remain upright without moving its legs is called **statically stable**. The **electric motor** is the most popular mechanism for both manipulator actuation and locomotion, but **pneumatic actuation** using compressed gas and **hydraulic actuation** using pressurized fluids also have their application.

5.10 ROBOTIC PERCEPTION

Perception is the process by which robots map sensor measurements into internal representations of the environment. Perception is difficult because sensors are noisy, and the environment is partially observable, unpredictable, and often dynamic. In other words, robots have all the problems of **state estimation** (or **filtering**). As a rule of thumb, good internal representations for robots have three properties: they contain enough information for the robot to make good decisions, they are structured so that they can be updated efficiently, and they are natural in the sense that internal variables correspond to natural state variables in the physical world.

Robot perception can be viewed as temporal inference from sequences of actions and measurements, as illustrated by this dynamic Bayes network. For robotics problems, we include the robot's own past actions as observed variables in the model. Figure 25.7 shows the notation used in this chapter: \mathbf{X}_t is the state of the environment (including the robot) at time t , \mathbf{Z}_t is the observation received at time t , and A_t is the action taken after the observation is received.

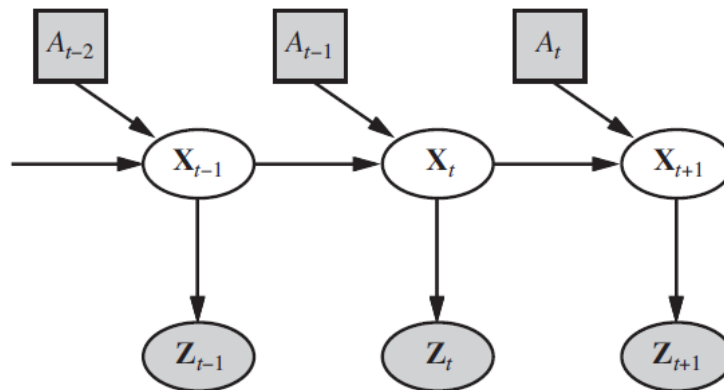


Figure 25.7 Dynamic Bayes network.

The task of filtering or updating the belief state is integration rather than summation:

$$\begin{aligned} & \mathbf{P}(\mathbf{X}_{t+1} \mid \mathbf{z}_{1:t+1}, a_{1:t}) \\ &= \alpha \mathbf{P}(\mathbf{z}_{t+1} \mid \mathbf{X}_{t+1}) \int \mathbf{P}(\mathbf{X}_{t+1} \mid \mathbf{x}_t, a_t) P(\mathbf{x}_t \mid \mathbf{z}_{1:t}, a_{1:t-1}) d\mathbf{x}_t \end{aligned}$$

Localization and mapping

Localization is the problem of finding out where things are present, including the robot itself. Navigating robots must know where they are to find their way to goal location. Localization problem comes in three flavors of increasing difficulty. If the initial pose of the object to be localized is known, localization is a **tracking** problem.

More difficulty is in **global localization** problem, in which the initial location of the object is entirely unknown. In **kidnapping** problem we are mean to our robot and kidnap the object it is trying to localize. It is used to test the robustness of a localization under extreme conditions.

Next, we need a sensor model. We will consider two kinds of sensor model. The first assumes that the sensors detect *stable, recognizable* features of the environment called **landmarks**. For each landmark, the range and bearing are reported. Suppose the robot's state is \mathbf{x}_t and it senses a landmark whose location is known to be $(x_i, y_i)^T$. Without noise, the range and bearing can be calculated by simple geometry.

$$\hat{\mathbf{z}}_t = h(\mathbf{x}_t) = \begin{pmatrix} \sqrt{(x_t - x_i)^2 + (y_t - y_i)^2} \\ \arctan \frac{y_i - y_t}{x_i - x_t} - \theta_t \end{pmatrix}$$

```

function MONTE-CARLO-LOCALIZATION( $a, z, N, P(X'|X, v, \omega), P(z|z^*), m$ ) returns
a set of samples for the next time step
  inputs:  $a$ , robot velocities  $v$  and  $\omega$ 
            $z$ , range scan  $z_1, \dots, z_M$ 
            $P(X'|X, v, \omega)$ , motion model
            $P(z|z^*)$ , range sensor noise model
            $m$ , 2D map of the environment
  persistent:  $S$ , a vector of samples of size  $N$ 
  local variables:  $W$ , a vector of weights of size  $N$ 
                     $S'$ , a temporary vector of particles of size  $N$ 
                     $W'$ , a vector of weights of size  $N$ 

  if  $S$  is empty then      /* initialization phase */
    for  $i = 1$  to  $N$  do
       $S[i] \leftarrow$  sample from  $P(X_0)$ 
    for  $i = 1$  to  $N$  do    /* update cycle */
       $S'[i] \leftarrow$  sample from  $P(X'|X = S[i], v, \omega)$ 
       $W'[i] \leftarrow 1$ 
      for  $j = 1$  to  $M$  do
         $z^* \leftarrow$  RAYCAST( $j, X = S'[i], m$ )
         $W'[i] \leftarrow W'[i] \cdot P(z_j | z^*)$ 
       $S \leftarrow$  WEIGHTED-SAMPLE-WITH-REPLACEMENT( $N, S', W'$ )
  return  $S$ 

```

Figure 25.9 A Monte Carlo localization algorithm.

Localization using particle filtering is called **Monte Carlo localization**, or MCL. The MCL algorithm is an instance of the particle-filtering algorithm. The operation of the algorithm

is illustrated in Figure 25.10 as the robot finds out where it is inside an office building. In the first image, the particles are uniformly distributed based on the prior, indicating global uncertainty about the robot's position. In the second image, the first set of measurements arrives and the particles form clusters in the areas of high posterior belief. In the third, enough measurements are available to push all the particles to a single location.

The Kalman filter is the other major way to localize. A Kalman filter represents the posterior $P(\mathbf{X}_t | \mathbf{z}_{1:t}, \mathbf{a}_{1:t-1})$ by a Gaussian. The mean of this Gaussian will be denoted $\boldsymbol{\mu}_t$ and its covariance $\boldsymbol{\Sigma}_t$. The main problem with Gaussian beliefs is that they are only closed under linear motion models f and linear measurement models h .

Sometimes the navigating robot will have to determine its location relative to a map it doesn't quite know, at the same time building this map while it doesn't quite know its actual location. This mapping problem is often called as **Simultaneous localization and mapping**, abbreviated as **SLAM**.

Other types of perception

Not all of robot perception is about localization or mapping. Robots also perceive the temperature, odors, acoustic signals, and so on. Many of these quantities can be estimated using variants of dynamic Bayes networks. All that is required for such estimators are conditional probability distributions that characterize the evolution of state variables over time, and sensor models that describe the relation of measurements to state variables.

One common approach is to map high dimensional sensor streams into lower-dimensional spaces using unsupervised machine learning methods. Such an approach is called **low-dimensional embedding**. Machine learning makes it possible to learn sensor and motion models from data, while simultaneously discovering suitable internal representations.

5.11 PLANNING TO MOVE

The hardest part is deciding how to move effectors. The **point-to-point motion** problem is to deliver the robot or its end effector to a designated target location. A greater challenge is the **compliant motion** problem, in which a robot moves while being in physical contact with an obstacle. An example of compliant motion is a robot that pushes a box across a table top.

The complication is added by path planning in *continuous* spaces. There are two main approaches: **cell decomposition** and **skeletonization**. Each reduces the continuous path-planning problem to a discrete graph-search problem.

5.11.1 Configuration space

Consider a simple representation for a simple robot motion problem. The robot arm shown in Figure has two joints that move independently. Moving the joints, alters the (x, y) coordinates of the elbow and the gripper. This suggests that the robot's configuration can be described by a four dimensional coordinate: (x_e, y_e) for the location of the elbow relative to the environment and (x_g, y_g) for the location of the gripper. Clearly, these four coordinates characterize full state of robot. They constitute what is known as **workspace representation**, since the coordinates of the robot are specified in the same coordinate system as objects it seeks to manipulate (or to avoid). Paths adhere to workspace is **configuration space** representation.

The problem with the workspace representation is that not all workspace coordinates are actually attainable, even in the absence of obstacles. This is because of the **linkage constraints** on the space of attainable workspace coordinates.

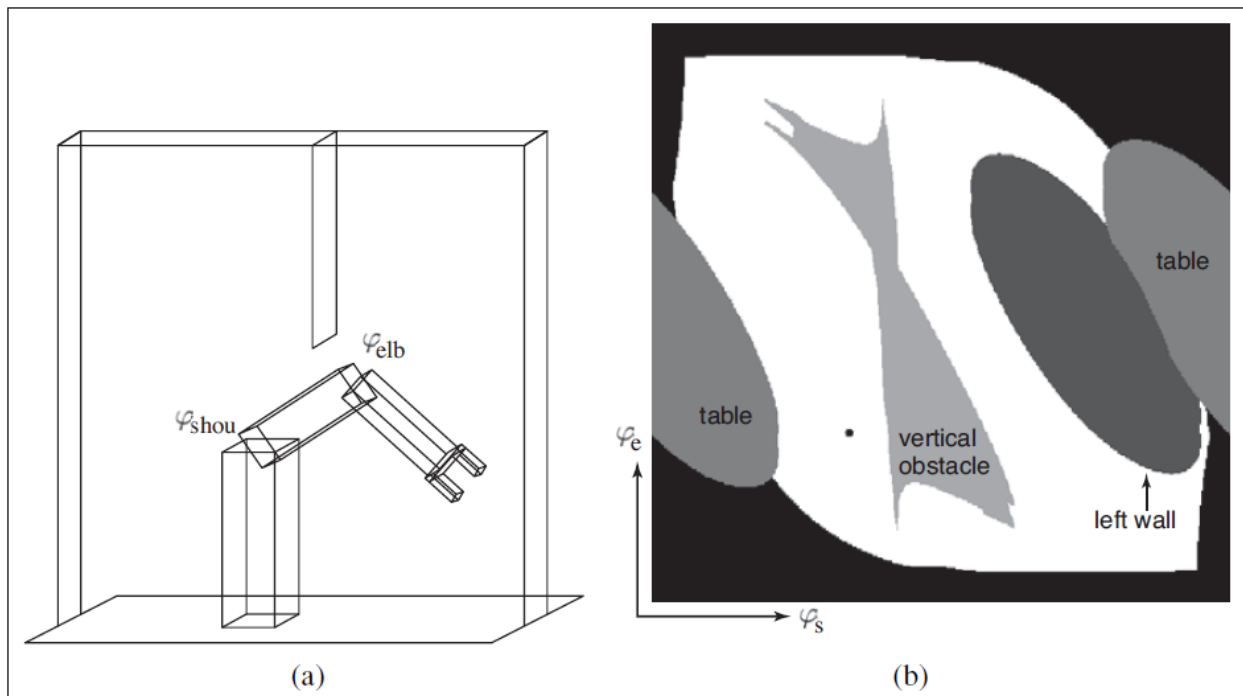


Figure 25.14 (a) Workspace representation of a robot arm with 2 DOFs. The workspace is a box with a flat obstacle hanging from the ceiling.

(b) Configuration space of the same robot.

Transforming configuration space coordinates into workspace coordinates is simple: it involves a series of straightforward coordinate transformations. These transformations are linear for prismatic joints and trigonometric for revolute joints. This chain of coordinate transformation is known as **kinematics**. The inverse problem of calculating the configuration of a robot whose effector location is specified in workspace coordinates is known as **inverse kinematics**. Calculating the inverse kinematics is hard, especially for robots with many DOFs. The space of all configurations that a robot may attain, commonly called **free space**, and the space of unattainable configurations, called **occupied space**.

5.11.2 Cell decomposition methods

Cell decomposition decomposes the free space into a finite number of contiguous regions, called cells. The path-planning problem within a single region can be solved by simple means. It is extremely simple to implement, but it also suffers from three limitations.

1. It is workable only for low-dimensional configuration spaces, because the number of grid cells increases exponentially with the number of dimensions. This is the **curse of dimensionality**.
2. Second, there is the problem of what to do with cells that are “mixed”, that is, neither entirely within free space nor entirely within occupied space.
3. And third, any path through a discretized state space will not be smooth.

5.11.3 Modified cost functions

A **potential field** is a function defined over state space, whose value grows with the distance to the closest obstacle. The potential field can be used as an additional cost term in the shortest-path calculation. This induces an interesting tradeoff. On the one hand, the robot seeks to minimize path length to the goal. On the other hand, it tries to stay away from obstacles by virtue of minimizing the potential function. With the appropriate weight balancing the two objectives are satisfied and the resulting path is longer, but it is also safer.

5.11.4 Skeletonization methods

Skeletonization algorithms reduce the robot’s free space to a one-dimensional representation, for which the planning problem is easier. This lower-dimensional representation is called a **skeleton** of the configuration space. Figure shows an example skeletonization: it is a **Voronoi graph** of the free space—the set of all points that are equidistant to two or more obstacles.

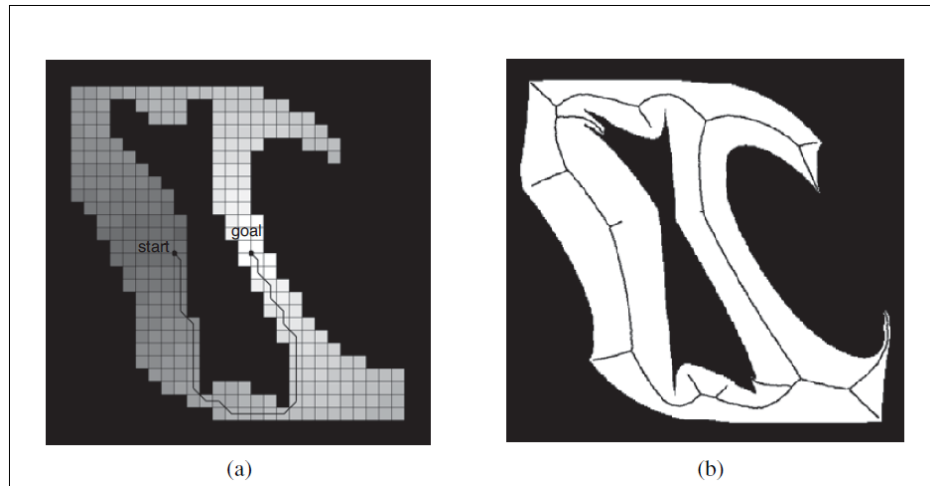


Figure 25.16 (a) Discrete grid cell approximation (b) The Voronoi graph.

5.11.5 Planning Uncertain Movements

In robotics, uncertainty arises from partial observability of the environment and from the stochastic effects of the robot's actions. Most of today's robots use deterministic algorithms for decision making, to extract the **most likely state** from the probability distribution produced by the state estimation algorithm. Many robots plan paths online during plan execution, with **online replanning** technique.

Robust methods

Uncertainty can also be handled using **robust control** methods. A robust method is one that assumes a *bounded* amount of uncertainty in each aspect of a problem, but does not assign probabilities to values within the allowed interval. A robust solution is one that works no matter what actual values occur, provided they are within the assumed interval. An extreme form of robust method is the **conformant planning** approach which produces plans that work with no state information at all.

Fine-motion planning (or FMP) is a robotic assembly task. Fine-motion planning involves moving a robot arm in very close proximity to a static environment object. A fine-motion plan consists of a series of **guarded motions**. Each guarded motion consists of (1) a motion command and (2) a termination condition, which is a predicate on the robot's sensor values, and returns true to indicate the end of the guarded move. The motion commands are typically **compliant motions** that allow the effector to slide if the motion command would cause collision with an obstacle.

5.12 MOVING

Dynamics and control

Dynamic state extends the kinematic state of a robot by its velocity. For example, in addition to the angle of a robot joint, the dynamic state also captures the rate of change of the angle, and possibly even its momentary acceleration.

The transition model for a dynamic state representation includes the effect of forces on this rate of change. Such models are typically expressed via **differential equations**, which are equations that relate a quantity (e.g., a kinematic state) to the change of the quantity over time (e.g., velocity).

Controllers are techniques for generating robot controls in real time using feedback from the environment, so as to achieve a control objective. If the objective is to keep the robot on a preplanned path, it is often referred to as a **reference controller** and the path is called a **reference path**. Controllers that optimize a global cost function are known as **optimal controllers**. Optimal policies for continuous MDPs are, in effect, optimal controllers.

1. Controllers that provide force in negative proportion to the observed error are known as **P controllers**. The letter ‘P’ stands for *P proportional*, indicating that the actual control is proportional to the error of the robot manipulator.
2. A robot is said to be **strictly stable** if it is able to return to and then stay on its reference path upon such perturbations. The simplest controller that achieves strict stability in our domain is a **PD controller**. The letter ‘P’ stands again for *proportional*, and ‘D’ stands for *derivative*.
3. A controller that calculates the integral of the error over time is called a **PID controller** (for proportional integral derivative). PID controllers are widely used in industry, for a variety of control problems.

Potential-field control

Potential-field control defines an attractive force that pulls the robot towards its goal configuration and a repellent potential field that pushes the robot away from obstacles. Its single global minimum is the goal configuration, and the value is the sum of the distance to this goal configuration and the proximity to obstacles.

Reactive control

Reactive control is a reflex agent architecture. For example, picture a legged robot that attempts to lift a leg over an obstacle. We could give this robot a rule that says lift the leg a small height h and move it forward, and if the leg encounters an obstacle, move it back and start again at a higher height. On rugged terrain, obstacles may prevent a leg from swinging forward. This problem can be overcome by a remarkably simple control rule: *when a leg's forward motion is blocked, simply retract it, lift it higher, and try again.* The resulting controller is shown in Figure 25.24(b) as a finite state machine.

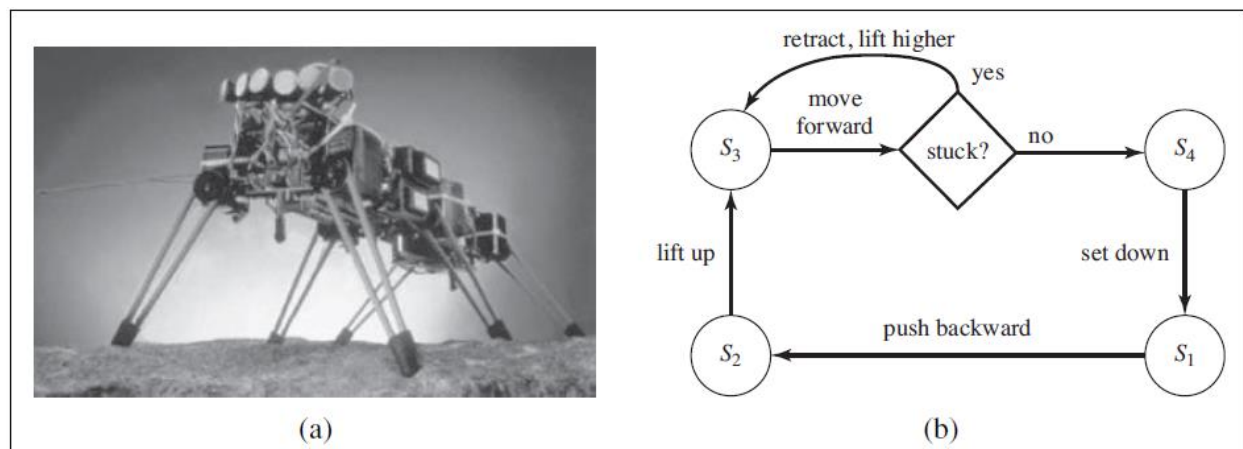


Figure 25.24 (a) Genghis, a hexapod robot. (b) An augmented finite state machine (AFSM) for the control of a single leg.

Reinforcement learning control

One particularly exciting form of control is based on the **policy search** form of reinforcement learning. Policy search is the simplest of all the methods in this chapter: the idea is to keep twiddling the policy as long as its performance improves, then stop.

1. Difficulties of Machine translation systems

Machine Translation is difficult because it involves three unsolved problems:

- creating a complete knowledge representation of everything;
- parsing into that representation; and
- generating sentences from that representation.

2. Give steps in Translation Procedure?

1. Find parallel texts
2. Segment into sentences
3. Align sentences
4. Extract distortions
5. Improve estimates with EM.

3. What is Speech recognition?

Speech recognition is the problem of computing the most likely sequence of state variables, $x_{1:t}$, given a sequence of observations $e_{1:t}$.

$$\operatorname{argmax}_{word_{1:t}} P(word_{1:t} | sound_{1:t}) = \operatorname{argmax}_{word_{1:t}} P(sound_{1:t} | word_{1:t}) P(word_{1:t})$$

4. What is noisy channel model?

This approach was named the noisy channel model by Claude Shannon (1948). He described a situation in which an original message (the *words* in our example) is transmitted over a noisy channel (such as a telephone line) such that a corrupted message are received at the other end. Shannon showed that no matter how noisy the channel, it is possible to recover the original message with arbitrarily small error, if we encode the original message in a redundant enough way.

5. Give full form of MFCC? What is it?

Then we compute a measure called the mel frequency cepstral coefficient (MFCC) or MFCC for each frequency. We also compute the total energy in the frame (signal over a time slice).

6. Define Robots?

Robots are physical agents that perform tasks by manipulating the physical world. To do so, they are equipped with effectors such as legs, wheels, joints, and grippers. Robots are also equipped with sensors, which allow them to perceive their environment, including cameras and lasers to measure the environment, and gyroscopes and accelerometers to measure the robot's own motion.

7. What is a manipulator in robot?

Manipulators or robot arms are physically anchored to their workplace, in a factory assembly line. Manipulator motion usually involves a chain of controllable joints, enabling such robots to place their effectors in any position within the workplace.

8. List the limitations of robotics?

- Real robots must cope with environments that are partially observable, stochastic, dynamic, and continuous.
- Many robot environments are sequential and multi-agent.
- Robot cameras cannot see around corners, and motion commands are subject to uncertainty due to gears slipping, friction, etc.

9. What are the two types of sensors in robotics?

Passive sensors, such as cameras, are true observers of the environment. They capture signals that are generated by other sources in the environment. Active sensors, such as sonar, send energy into the environment.

10. What is odometry?

On mobile robots, shaft decoders that report wheel revolutions can be used for odometry—the measurement of distance traveled. Unfortunately, wheels tend to drift and slip, so odometry is accurate only over short distances.

11. Write about SLAM?

Sometimes the navigating robot will have to determine its location relative to a map it doesn't quite know, at the same time building this map while it doesn't quite know its actual location. This mapping problem is often called as Simultaneous localization and mapping, abbreviated as SLAM.

12. What do you mean by Cell decomposition method?

Cell decomposition decomposes the free space into a finite number of contiguous regions, called cells. The path-planning problem within a single region can be solved by simple means.

13. What is Skeletonization method?

Skeletonization algorithms reduce the robot's free space to a one-dimensional representation, for which the planning problem is easier. This lower-dimensional representation is called a skeleton of the configuration space.

14. Write on Robust method?

A robust method is one that assumes a *bounded* amount of uncertainty in each aspect of a problem, but does not assign probabilities to values within the allowed interval. A robust solution is one that works no matter what actual values occur, provided they are within the assumed interval.

15. What are the three major of IRS?

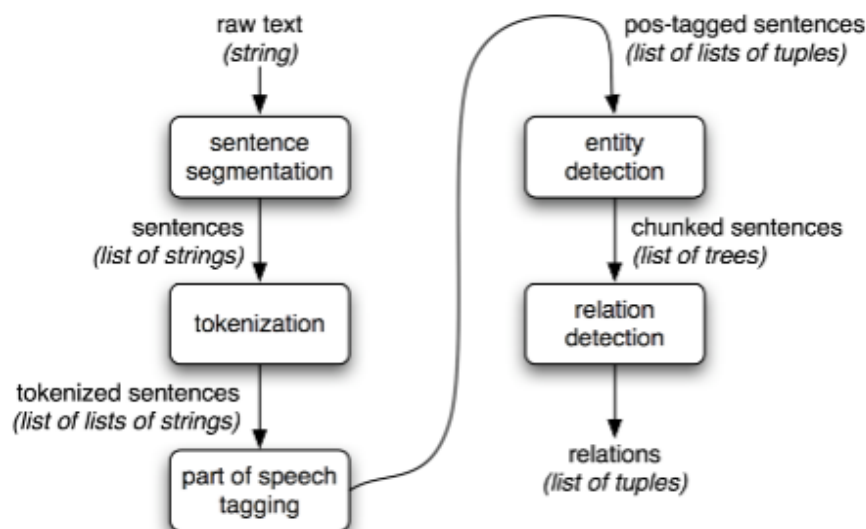
Three major components of IRS

- 1) Document subsystem
 - a) Acquisition
 - b) Representation
 - c) File organization
- 2) User sub system
 - a) Problem
 - b) Representation
 - c) Query
- 3) Searching /Retrieval subsystem
 - a) Matching
 - b) Retrieved objects

16. Define Information Extraction

Information extraction (IE) is the automated retrieval of specific information related to a selected topic from a body or bodies of text. Information extraction is the process of extracting specific (pre-specified) information from textual sources. One of the most trivial examples is when your email extracts only the data from the message for you to add in your Calendar.

17. Draw the architecture of Information Extraction system.



18. Define NLP.

Natural Language Processing is a field that covers computer understanding and manipulation of human language, and it's ripe with possibilities for news gathering.

NLP is used to analyze text, allowing machines to understand how human's speak. This human-computer interaction enables real-world applications like automatic text summarization, sentiment analysis, topic extraction, named entity recognition, parts-of-speech tagging, relationship extraction, stemming, and more. NLP is commonly used for text mining, machine translation, and automated question answering.

19. What is Lexical Analysis?

Lexical Analysis – It involves identifying and analyzing the structure of words. Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.

20. What is Language Model?

The goal of a language model is to assign a probability to a sequence of words by means of a probability distribution. Formal grammars (e.g. regular, context free) give a hard “binary” model of the legal sentences in a language.

PART B (13 MARK QUESTIONS)

1. Write the procedure to create a Translation machine?
2. Write in detail about Speech Recognizer?
3. What is the use of AI in robotics domain?
4. Give the hardware issues in robots?
5. Explain perception in robotics?
6. Write a short note on robots plan making method to move?
7. Explain the operations involved in moving a robot?
8. Explain in detail about Information Extraction
9. Discuss the process involved in Information Retrieval.
10. What is Natural Language processing? Explain it in detail.