NLP TERM REPORT

Natural Language Processing

Saranya M S (CS15D006) *

Correspondence:
cs15d006@cse.iitm.ac.in
Department of CSE, IITM,
Chennai, India
Full list of author information is
available at the end of the article
*Single contributor

Contents

Chapter 1

1 Overview of NLP

Languages born when human tried to communicate more information that can be conveyed by their sign languages and sounds. The languages evolve over a period of time. Though the vocabulary size of a language increases, the basic unit of sounds (phonemes) produced to form those vocabulary are confined. The basic unit of speech is called phones. The set of phones confined to a language is called phonemes. Once humans start to civilize and think, he analyzed the nature and started to discover the science behind everything. His quest did not stop at the understanding, he also tried to apply it in his day to day life.

This analyzing quest turned towards the language which is the basic mode of communication when he tried to make the system to understand the same. During 1950's this natural language processing starts to get more attention as it intersected with the another vast domain called artificial intelligence (AI).

The natural languages are very vast and un-restrictive in nature with many ambiguities This makes it tougher to come up some set of hand-written rules to feed to the systems. Two main problems that are needed to be faced while using handcrafted rules are [1],

- Extracting meaning from the text (semantics) is hard.
- Hard to get even the hand-written rules to come up with human comprehensible, 'ungrammatical' phrases.

In recent years, the real time applications like

- Language Translation: translating the sentence from one language to the other,
- Language Identification: identifying a language from the speech, or from text,
- Voice commanding machines: detecting or understanding the emotion or context,
- Context based spell checking, etc

makes the natural language processing (NLP) a rigorous research field. NLP deals with text. We can describe NLP as the process of understanding and producing the understandable texts in any human language either as text or speech.

The NLP can be split into two components namely natural language understanding (NLU) and natural language generation (NLG) as shown in Figure 1

The NLU is the process of interpreting a text and converting it into some symbolic representation for machine understanding and usage. The NLG is the process of taking some source or representation like discourse model of a language and generating a text or speech or embodied part of a document. Few examples for existing NLG are

- Traffic mapping: Interpreting the satellite road maps and giving voice instructions accordingly.
- Language translation: Converting text from one language to the other like *G*oogle translator
- Text to speech (speech synthesis) or speech to text conversion (speech recognition)

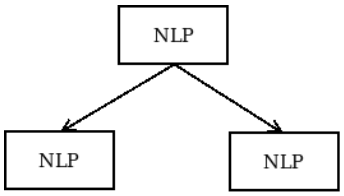

Figure 1: Components of NLP

Figure 2: Natural language understanding

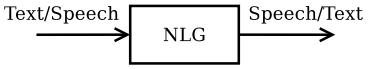

Figure 3: Natural language generation

2 System Vs Human Intelligence

The general assumption is that NLU is harder problem than NLG. Because NLG does not require much precision as NLU. For example, one can compare automated flight take model as the NLG model where the choices and corresponding situations can be given in prior. It has to select one among the existing choices in according to the situation. But NLU is like trying to automate the aircraft landing where many unexpected situations can appear due to the evolving nature of a language.

Another reason for NLU is the ambiguity in the source. The text which serves as the source for the NLU component, exists in both structured and unstructured ways. Humans do interpretation even from unstructured form of resources due to the complicated and still unexplored nervous system. Expecting the same from system is nearly impossible until we find out a way to interpret all those stuffs. Interpreting those information are even hard to humans because **"we do many stuffs but do not know how it is being done?"**. This is because of the **non-conscious** knowledge in our mind. It is hard to represent something clearly without much understanding. And to make the systems work on it, it requires more precision as discussed before.

2.1 Structure Function Correspondence

The human thinking can be related to a black box, where the input is fed and output is obtained without bothering much about its internal functionality. But machines need a sequence of steps which converts the input to an output. The process is well structured. For the successful modelling of a scenario via a machine, the whole process has to be structured well. The modelling will be successful if the structure function correspondence is high. To be more clear, the way in which the human brain reads a newspaper by selecting the required column or section out of all the paper and interpreting the information can be considered as a function. But the same is not so simple with the system. Making the machine to understand the semantics of a phrase is a harder problem. It requires the information to be in a more structured form like tables.

2.2 Model Diagnosis

The other main problem in the structure in structure-function correspondence is **"notion of a structure is not universal"**. It is based on the application and

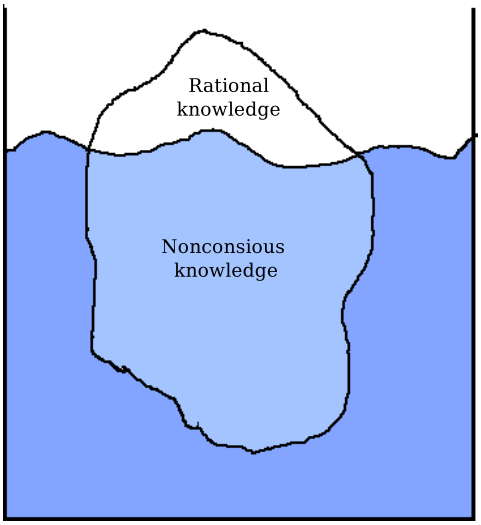

Figure 4: Conscious vs non-conscious knowledge

its context. The system model of a scenario need not to be perfect in its first attempt. The structure of the model can be updated iteratively by observing how far its behaviour is deviating from the empirical behaviour. The model is the abstract behaviour of the system and it can be incomplete. Given the observation of a system (real-time), the model simulates the behaviour of the system. The actual observations and predicted observations are compared to understand how far the model has deviated from the system as in 5.

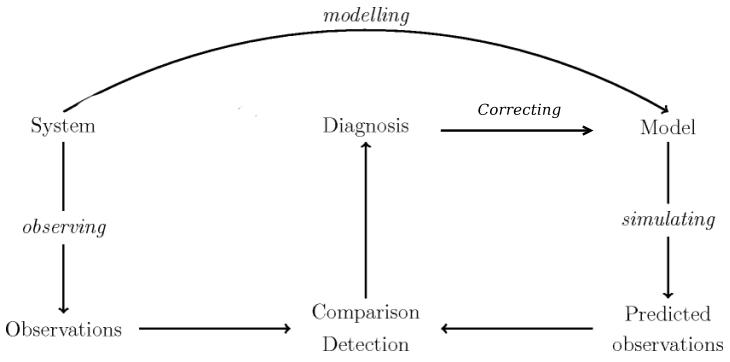

Figure 5: Model Diagnosis

Classical artificial intelligence (AI) can be used for better modelling of a system. The model of a system is developed by certain rules. The rule based systems may not be perfect due to certain level of uncertainty prevailing in the structure. In such cases "rules are padded with certain probabilities". The two ways in structuring a model are

- bottom-up-representation: the models are formulated from set of rules.

- top-down-representation: first model is formulated, then rules are derived from the models.

Top down approach ensures that one is much aware of what is needed and what is to be done in a much promising way.

2.3 Classical AI

The study of understanding the intelligent entities of human and an attempt to represent/model it, is called artificial intelligence (AI) [2]. AI helps to learn about ourselves and our thinking process. AI tries to address the functionality of a tiny brain. AI turned the philosophers study on how the inputs of sense organs are being remembered by this little brain into real experimental and theoretical discipline. The two broad sub-fields of AI are

- General purpose: logical reasoning, perception etc
- Application specific: chess games, poetry, disease diagnosis etc

Over the decades, many definitions have been evolved to describe AI. Though there are many definitions they vary in only two dimensions namely "reasoning" and "behaviour".

Acts/Thinks	Rationally	like Humans
Rationally	Act rationally & think rationally	Act rationally & think like human
like Humans	Act like human & think rationally	Act like human & think like human

The definitions are based on *"human"* performance and ideal concept of *" intelligence"* called as *"rationality"*. A human-centered approach is an empirical science with lot os hypothesis and experimental confirmation, while the rationalist approach is defined by the combination of mathematics and engineering.

2.4 AI system and Conventional system

The fundamental difference between the normal conventional system and AI system is **decision making**. In conventional system the process of decision making is encoded completely in prior with all possibilities and appropriate decisions at every possible conditions. Eg:- tree parsing

On the other hand, in AI system *"what to do"* is specified but "how to do" is not specified. That decision is to be made by the AI system.

3 Applications of NLP

NLP is the field that deals with most of our day-to-day life products like emails, web pages, tweets, social media, newspapers, scientific articles in many languages across the world. It has the unavoidable usage right from spell check, grammar check to language translation, toy applications like automatic question answer , text comprehension etc.

Information extraction: Reading a mail or passage identifying the information like date, place and create a calendar entry automatically.

Sentiment analysis: Getting user feelings from the comments given for a product online. For example, when the consumer page of a camera which has lots of comments by various users about the products, analysing and splitting it into good and bad reviews can be done by sentiment analysis.

Machine translation: Converting a text from one language to another language. The best and simpler example for this is the *"google translator"*. Though it does not work upto the mark for some uncommon words or vernacular speech.

Text summarization: This produce abstract version of a document with user specific relevant information. Eg: Reading a documnet and producing the headlines, summary, outline etc. This summarization can be done on multiple documents also where a gist of the documents, or the stories that are common across the documents, or identifying the set of web pages with same content can be retrieved.

Parsing: Parsing is the process of representing a sentence in the form of a tree represented by the parts of speech of that sentence. Using grammar and parts of speech for parsing is a poor idea. As the grammar structure increases, the parsing tree structure will grow exponentially. If we restrict the grammar to control the parse structure it reduces the robustness nature of it. *Penn's tree bank* came with an idea of annotated parsing which reduces the complication of parsing by using annotated parse tree. The annotated parse tree contains the information of parts of speech , frequency of the words, and the distributional information.

IBM Watson System: This is a *"question-answer"* system, based on cognitive computing. It behaves like humans by observing the data fed to it, evaluating the data based on grammar and huge set of rules programmed into it followed by decision making. Do not need structured data. It can understand from unstructured data like blogs, newspapers, websites etc. Watson does not simply looks for a synonym or keywords alone as he search engines. It interprets the text like a person by breaking the sentence grammatically, relationally and structurally by understanding the context. Watson performs corpus collection which is overlooked by human to remove anything that is out of date. This process is called as "curating the content". The data is preprocessed and an analogy graph is created. Training is done with the help of machine learning to understand the human interpretation via question answer pairs. Watson keeps on reading in every Q&A process.

Synchronize-3-application: This applicationalso called as *"applink"*, allows the consumer's smart phones to synchronize with the vehicles thereby giving access to their smart phones seamlessly. Once synchronization is done, the user can utilize the smart phone completely in a hands free conversation mode while driving the vehicle.

Ring (reads text for blind): This is an innovative invention of a ring which can be worn on a finger to read the text on a page or book. This will be very useful for the blind people as it converts text to speech.

4 Critics of NLP in history

The specific branch of AI that deals with natural language processing is called computational linguistics. The work of Norwig on computational logic is criticized by Chomsky as *"a method of doing linguistics without involving language and linguistic science"*. But Norwig comments on Chomsky's work by saying it as a "notorious work in linguistics without any special regard for real-data. Chomsky wants to represent language with elegant theory which clearly surpasses the human erroneous nature and to get the simple structure lying under his complex brain structure. Meanwhile Norwig tries to represent everything by simplistic statistics. Chomsky

commented on the statistical machine learning methods that mimics the behaviour of some real world scenarios, as to simulate the bee dance without understanding the reason behind that dance. But Norvig claims that why should we complicate the process instead of following statistical model which is simpler and efficient. Norwig argues that every word in a language occurs with certain frequency and that frequency can be learned from the humongous amount of data.

5 Morphology

The source of NLP is "TEXT". The basic element of text is character. A sequence of character giving a proper meaning is called as a word. The sequence of words forming conveying an information is called sentence. Sentences together form the paragraph also know as discourse. This structure can be linked with chemistry and it looks like the Figure 8.

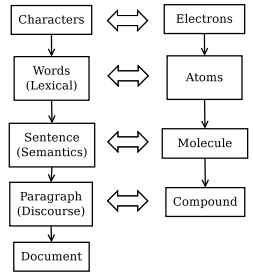

Figure 6: Morphology

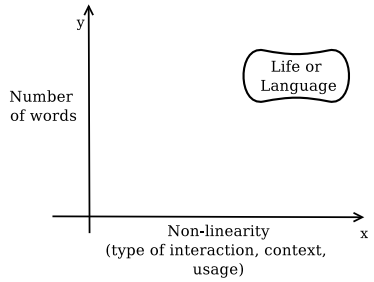

Figure 7: Words-Language relationship

The characters are similar to electrons which are the basic elements and do not convey any meaning when it is used alone. The combination of electrons leads to atoms and atoms are bonded to get a valid molecule. The combination is determined by the valency bond. Similarly the combination of words to make a meaningful sentence is determined by grammar and structure of the language.

The words evolve over a period of time. One word can be used to form the other by many process like *derivation* (Eg: educate –¿ education), *inflexion* (sing –¿ singing), etc. When words used in sentence, the role of it in the sentence is determined by *parts of speech*. The meaning of two individual words change completely when they come together. *Eg: ice-cream, honeymoon, pass-away ...*

Lexical analysis comes into picture in the formation of words from characters and semantics comes in to picture when words are combined to form a sentence. Sentences together form a paragraph or passage. Analysis the paragraph is totally a different task known as *discourse analysis*.

Words evolve over a period of word, which leads to higher dimensionality problem when documents are processed on word basis. As the words of a language increases it is not necessary for the language also to grow in same rate. This is because, all the words in the language are not used in daily interactions or in all context base don the life-style. Thus the growth of words of language an d the language are non-linearly related.

5.1 Word Co-occurrence

The relation between two words is defined as word co-occurrences if both words are in association with each other in some sense. That is, if there is a document with a word heart surgery, it is more likely to find the word by-pass surgery or angiography in the same document. If the related word are found in same document, then they are referred to as *"first order co-occurrences"*. If the related works are found in two different documents then they are called as *"second order co-occurrences"*.

5.2 Ambiguity and its types

Analysing and understanding a text/speech segment is pretty tough for a machine. Humans understand it duet to the cognitive knowledge that acts as a very big database formed right from one's child hood. When some one asks you *Is there water in jug?* it means if the water is there we have to pass that jug to the person who asked that question. Such sentences are called **pragmatics**. But system will not understand pragmatics. The process of understanding a word in more than one way is referred to as **ambiguity**. Ambiguity can occur at various levels of NLP. Ambiguity could be Lexical, Syntactic, Semantic etc [3].

- **Syntactic ambiguity:** This ambiguity involves operators and quantifiers. *Eg*: Every man loves a woman. Here there are two kinds of interpretations. The one is "For every man there is a woman", and the other is "there is one particular woman who is loved by every man". The scope of quantifiers are not clear and it can create ambiguity.
- **Semantic ambiguity:** This occurs when the meaning of the word itself is interpreted in more than one ways. *Eg: We saw her duck*. Duck can refer to the person's bird or to a motion he made. Semantic ambiguity happens when a sentence contains an ambiguous word or phrase.
- **Discourse analysis:** This analysis gives the information about the influence and relationship of across the sentences.This analysis includes *(i)Co-reference resolution* and *(ii) information extraction*. Co-reference identifies the phrases in a document that refers to same information. Information extraction locates specific piece of data from a document.

6 Types of Knowledge

Humans infer the language through the knowledge they acquire from there child-hood [4]. The process of a child getting the worldly knowledge from its care-takers and surroundings is termed as **"child knowledge acquisition**. Similar to this knowledge source, system needs knowledge to process the natural language. The knowledge that contributes the database for NLP can be divided into three types namely

- **Linguistic knowledge:** The dictionary is an example for linguistic knowledge from where one can get the common knowledge about the use of language. Explanations, meanings and synonyms of a word and their usage helps us in understanding the language in a proper way. *Eg: Oxford dictionary*
- **Introspective knowledge:** The knowledge within the given documents or reports are considered as introspective knowledge. The information like co-occurrences of words at different order (statistical co-occurrences) constitutes the introspective knowledge.

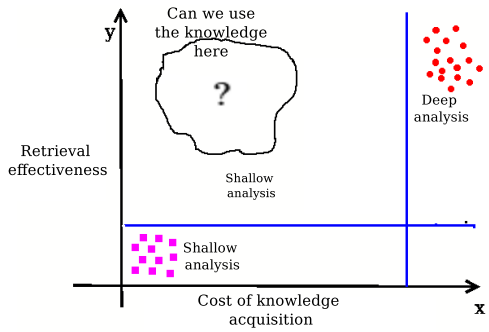

Figure 8: Shallow and deep linguistic analysis

- **External knowledge:**The world knowledge or external knowledge can be divided under variety of various fields. This knowledge also includes the common information shared across the fields also. The effective representation of knowledge via human media such as pictures, diagrams, voice, knowledge drawn from textbooks, encyclopedia, glossaries etc. *Eg: Wikipedia*

7 Linguistic processing

Language processing needs theoretical and descriptive analysis. Based on the level of analysis the linguistic processing is classified into (i) deep and (ii) shallow processing.

- **Shallow linguistic processing (SLP):** SLP also known as machine learning approaches altered the fundamental processing of NLP. Many rapid and robust creation of tools have been developed for this SLP which requires lesser time and manual effort. So the SLP gained more interest than deep linguistic processing. This can also be termed as *knowledge light NLP*.
- **Deep linguistic processing (DLP):** DLP is concerned with more computational development. The grammars for DLP were manually developed maintained and are computationally expensive. But the cost can be trade off with the effective performance given by DLP than SLP.

Chapter 2

8 History of NLP

In 1948 the first NLP application was developed for searching a word in dictionary using a look-up system. In 1950 the first machine translation was implemented to translate Russian language to English. The NLP really took its intense phase of growth from 1957 after the syntactic structures proposed by Noam Chomsky. Many researches are started after that and still evolving. After the introduction of machine learning and artificial intelligence the research development in NLP took a vertical growth.

The unavoidable usage of internet and technologies, made NLP as an essential part of peoples every day life. Chatter-bots like PARRY, Recter, and Jabberwacky were written and developed during early times of NLP research. The IBM Watson discussed in Section 3 was invented in 2011 which is a question and answer (Q&A) application, which makes the user so transparent to the system that responds from the other end. It is a general purpose artificial intelligence based system, which can hold a human-level conversation.

An enormous quantity of preprogrammed knowledge concerning both the language and the domain under examination is the prerequisite of any natural language processing application. The increase in computer power made this transformation possible thereby allowing more research and programs to be written for NLP.

9 Why NLP is hard?

Sheer complexity of sentence structure of our text/spoken language makes the NLP problem harder to get implemented. The indicatives also known as *known facts*, and our rationally questioning nature of *"what if...?* in many cases increases complexity in formulating the rules for these kind of applications. Processing text in becoming more hard because of the various methods of formulating the text. For example it is rule to use periods (.) after the title like *Mr., Mrs.,* etc and at the end of the sentences. its is harder to find and differentiate between these two periods, wherein other unnecessary and unwanted periods are being inserted in many places due to lack of awareness which makes the **sentence boundary detection** problem more complicated.

Apart from the lack of common notation in writing a text information, the syntactic ambiguity because of the polysemous words makes it tough to understand and differentiate the words. The major and still unsolved problem is the context detection in the language. The meaning of the sentences vary depending on the intent of the speaker, listener, location, prior, time and social content as shown in some scenarios given below [5].

- Depends on the people present e.g. "How far is it?" (miles or km)
- Depends on the social context: "That was bad!"
- Depends on the location, e.g. "It is playing in upstairs" (It refers to song that is being played in upstairs.)
- Depends on the time of day, e.g. "Let's go eat" (eat may refer to breakfast, lunch or dinner)
- Depends on prior sentences: "The third one" (depends on what other two are?)

Apart from this, recognising the names of people, place, animal, object and differentiating the various slangs, jargon, sarcasm, spelling mistakes, humour, grammar mistakes etc makes the natural language processing the more challenging research problem.

10 Artificial Intelligence

Artificial intelligence can be literally defined as the intelligence possessed by machines or software. AI is the separate engineering field where the study deals with making intelligible machines and software. Intelligence in here describes the ability of the system that perceives the environment and takes the action such that it maximizes the chance of success in whatever application it work towards. AI has its own part from generalized applications to specific applications like gaming. The AI is used where there is the need for reasoning, planning and learning. Since NLP has all these in its field, AI is used widely in NLP. Humans have the ability to precisely describe their need and making the system to simulate it. But the range of precision clearly affects the performance of the system. As far as the rules are clear, systems will work fine with it.

AI leads to many latest and admirable inventions like smart phones, automatic vehicles, smart homes, auto-bots, language learning tools, hardware interactions etc. Many IVR based systems are developed for ubiquitous interaction of humans with the machines and electronics on the fly. This is made possible only because of the fusion of two most important and complicated fields by thte name AI nad NLP.

11 Conscious and non-conscious Knowledge

Right from the child hood we humans do many task with the help of our organs controlled by the only CPU in our head called brain. The tasks like understanding a language, picking up the required sound from crowded hall, summarizing the text document, inferring the image and storing the details, everything are done just like that without much knowledge of "actually how it is done?". These actions are done as an involuntary reaction for the instructions given by the brain. If we decode these set of actions and understand it, then it will be easier for us to instruct the machines to follow the same set of instructions to simulate the human behaviour.

But it is hard to explain something that one does not understand black and white. Those unexplainable knowledge is termed as *"non-conscious-knowledge"*. If one can able to answer a question clearly by description or diagram or by any means the same can be interpreted by system if the way of our interpretation is given to it as input. For example, if some one asks to define force, it can be defined empirically and mathematically. This is because of *conscious knowledge* which we obtain through our rational understanding. If someone asks to explain a color, it is impossible to explain unless one see many object with same color and classifies it internally with that class label "green". This happens because of the *non-conscious* knowledge that resides in our brain due to the humongous processed database collected right from our childhood.

12 Morphology

The smallest meaningful grammatical unit in a language is called *morpheme*. The study of morpheme sis called morphology. The morpheme and words are not same. Morpheme may or may not stand alone. But words always stand-alone [6]. There are different types of morphemes as described below:

- **Free morphemes:** The morphemes that are stand alone and act as words. Eg: dog, cat, town, city etc.
- **Bound morphemes:** Appears as a part of a word. Always occurs in conjunction with a root word as affixes, prefixes, and suffixes(tion, ing, ation etc).
- **Derivational morphemes:** The morpheme together with the standalone word called as root word, changes the meaning or parts of speech of that root word. *Eg: happy+ness=happiness. Happy (adjective), happiness(noun).*
- **Inflectional morphemes:** Modifies the verb's tense or a noun's number without affecting the meaning. *Eg: Sing+ing=singing. Root: sing, singing(current ongoing action).*

As discussed in Section 5, the character forms the basic of a written language from where the words, phrases, sentences and paragraphs are formed. In spoken language the same sequence can be compared with phonemes, syllables, words, sentences and paragraphs. Phonemes are like the basic elements. The combination of vowels (V) and consonants (C) as CV, VC, CVC, CCV, etc forms the syllables. The syllable structure of every language varies. The basic sound units of a language are termed as *"phones"* and the sounds that are confined to a language are called as *"phonemes"*.

The words of a language keep on evolving. The one brilliant idea followed or insisted by our ancestors are to keep the basic sound units, i.e., phonemes constant and evolving new words as per the requirement based on these phonemes. Apart from the semantic ambiguity or context difference, difference and complications starts to occur right from the word level of a language. There are different words with same meaning(synonyms), different or same spelling words with same pronunciation (homonyms), a word having more specific meaning than the other (hyponym), a metaphorical word, that used to refer the other word which actually points to the part of a whole (meronym) etc. Understanding and differentiating these words are not a trivial task.

These problems can be solved by having large number of examples for each word in the database and tagging them with descriptions called synsets. This helps us to resolve the syntactic ambiguity problem to a certain extent.

13 Ambiguity

Ambiguity in NLP occurs in all levels of analysis right from the morphology to discourse analysis. Coming up with a software that decides the meaning of a given text or speech without considering the context is like a lost battle. The basic definition of ambiguity can be given as *"the ability to understand the piece of text/information in more than one way"* [7]. The most common types of ambiguity from literature are

- **Lexical ambiguity:** More than one interpretation of a word. Eg: Flies (an action of flying / an insect)

- **Syntactic ambiguity:** Variation in the interpretation of a sentence due to the grammatical structure. Eg: He saw her duck (duck may be an action/a bird).
- **Semantic ambiguity:** Different interpretations of meaning based on the based on the meaning of words in the phrase when combined. Eg: *"Colourless green ideas sleep furiously"* is a sentence composed by Noam Chomsky in his book "Syntactic Structures" as an example of a grammatically correct, but semantically nonsensical statement.
- **Pragmatic ambiguity:** The context of the phrase or sentence gives completely different meaning for the sentence's actual meaning. This commonly happens when conversation happens between two entities in different domain without any common basis. Eg: *"superfluous hair remover"*. Here we are not sure whether the superfluous describes the process of hair remover or the noun remover. Because of the improper representation of scope of specifier this pragmatic ambiguity may occur.

14 Child knowledge acquisition

Language acquisition is the process of acquiring the capacity to understand and interpret a language, and to communicate in that language. This language based communication is meant especially for homosapiens [4]. Usually the language acquisition starts right from our childhood, from the surroundings and caretakers. There are two types of language acquisition in general for a child namely (i) first language acquisition and (ii) second language acquisition. To get expertise in a language, one need to now different range of tools right from phonology, morphology to extensive vocabulary in that language. Language can be written as text or vocalized as speech.

The general approaches of child knowledge acquisition are (i) social interaction and (ii) emergent-ism. The former one happens because of the interaction interaction of a child with the linguistic-knowledge-adults. The instructive corrections given to the kid by its care-givers helps the child to correct the language. This works as a black box with feed-back mechanism. The later one can be defined as the cognitive process that emerges by the biological pressures and environment. This can be related a=to an example for better understanding. If a person familiar with only the mother tongue is relocated to another place with completely different language, he will learn that new language quickly for his better survival in that environment. The theory of child and adult knowledge acquisition process is detailed in [8].

15 Other applications of NLP

- **Co-reference resolution:** This is the process of analysing the given text document or a chunk of text and identifying the words referring to same information. For examples, matching up the pronouns with the nouns that are referring to. It is also called as "bridging-relationships" while referring the structure of one thing in a sentence. *Eg: She peeped in to Rosy's house via the glass window.* In this sentence, the glass window is the structure of Rosy's house. It indirectly refers to Rosy's house.

- **Morphological segmentation:** The process of identifying the morphed words, separating it from the root word and classifying the morphemes is called morphological segmentation. For few languages like English which has simple morphology, all possible words can be listed instead of going for morphological segmentation. But this is not the case in highly agglutinated languages like Tamil, Turkish etc.
- **Named Entity Recognition (NER):** Given a stream of text, predicting the upcoming word or next word or completing the present typing the word is called NER. Forming a common way of writing the text to identify the starting of hte word will also fail in case of representing the name of a a person, place or animal. Handling this is also another difficult problem in NLP.
- **Optical Character Recognition (OCR):** Scanning text from a scanned image of a document/book. There is a problem of mis-interpreting two closely written characters. *Eg: Interpreting "cl" as "d".*
- **Sentence boundary disambiguation:** Usually the written texts are bounded to some rules like the usage of periods and punctuations for better readability and sense. But the same can also be used to represent abbreviation. To make the difference between the sentence period and abbreviation-al notations, and segmenting the sentence accordingly is called as sentence boundary disambiguation. *Eg: I woke up at 5 A.M. I went for a walk and met the C.E.O of a company in the park.* Here the difference between the period after 5 A.M and the punctuation between A.M and C.E.O needs to be differentiated.

16 Summary and Conclusion

The Sections 1 to 7 in Chapter 1, gives the verbatim of the introduction of NLP taught in the first week classes. These chapters helped us to understand the basics and origin of NLP field and the current trending application of it. The NLP evolved when humans try to simulate the language behaviour through the system for faster and easier communication in this technological world. The mapping between the human knowledge and that of the systems (artificial intelligence) are discussed in Section 2. The general structure of any language is learnt from morphology discussed in Section 5. The type of knowledge required by the system to perform any NLP tasks is elaborated in Section 6 and the methods of language processing by the systems is discussed in Section 7.

The remaining Sections from 8 to 15 discusses the information I gained through the self-study suggested during class hours. Section 8 is just the output of a curiosity in digging the origin of NLP and its evolution in the current trend. Section 9 is the suggestion for defining natural language processing as a hard problem. Section 10, 11 and 13 is about my views on AI, conscious vs non-conscious knowledge and ambiguity. The term child knowledge acquisition mentioned in class and through the video lecture from TED talk made me to search and understand the process of language knowledge acquired by a child, which is explained in Section 14. Apart from the NLP applications discussed in class, few more applications are elaborated in Section 15 which caught my interest.

Chapter 3

17 Introduction

The natural language processing as over-viewed in first two chapters has loads of applications in real-time which need to be un-earthed to reach realize it's full effectiveness in our day to day life. This chapter comprises of problems the cause, the symptoms and solutions of real time problems that are attempted and solved (in many cases) till now. From the duration of past 6 months with the help of the course instructor, I felt the following are the important take-aways that I understood and can implement in real from the help of related published materials.

- Spell checking
- Information retrieval
- Semantic relatedness
- Word sense disambiguation
- Circularity problem
- Parsing and
- Machine translation

Each of the above enlisted topics are discussed in detail in other sections of this subject.

18 Spell Checker

Spell check is the process of identifying misspelled word in the given phrase or sentence. This can be implemented with different granularity like

- word-level
- phrase-level and
- sentence-level.

If a word is mis-spelled we have to play with just 26 characters in different combinations to come up with correct-word [9]. A word is decided as a proper word if it exist in dictionary. With this this, the spell-corrections at word-level can be done.

But in case of phrase level the other problems occurs. *Whether to identify all mis-spelled word in the phrase? or a single word* . The other problem pops up, when the correct spelled word occurs in wrong context like *I saw her near the see / A peace of cake* etc.

The other important requirement to perform spell check is the dictionary. This dictionary can be any kind of thesaurus, digital dictionary or a huge collection of real-time text data which can act as a repository to check with all the possible alternates for the mis-spelled word. This set of alternatives are termed as **"Candidate List"**.

18.1 Edit Distance to Generate Candidate List

From one word we can get another word by any one of the following process

- Insertion (port $->$ sport)
- Deletion (Spark $->$ park)
- Transposition (hte $->$ the)
- Substitution (spine $->$ swine)

Edit distance algorithm follows the strategy of dynamic programming, where the solution of a problem depends on the solution of its sub-problem. This nature is defined as the *optimal sub-structure*. This algorithm also has the *"overlapping sub-problem nature"* which can be illustrated by the example of Fibonacci series represented as

$$Fib(n) = Fib(n-1) + Fib(n-2) \tag{1}$$

The computation can be made faster by solving the sub-problem and saving the result in a look-up table which can be used for the new computations.

18.1.1 Applications of Edit Distance Algorithm

The edit distance algorithm has its applications in different fields of science and engineering as listed below:

- Spell check (character level comparison)
- Machine translation (word level comparison)
- Speech recognition (word level comparison)
- Computational biology (DNA sequence comparison)

Algorithm 1 Edit distance algorithm

1: {**Input:** Two words w_1 and w_2
Output: Edit distance between the words}
2: Initial condition or base condition

$$D(i,0) = i; \quad D(0,j) = j;$$

3: When j $\neq 0$ or i $\neq 0$

$$D(i,j) == min \begin{cases} D(i-1,j)+1 \\ D(i,j-1)+1 \\ D(i-1,j-1) + \begin{cases} 1, & if \quad s_1(i) \neq s_2(j) \\ 0, & if \quad s_1(i) = s_2(j) \end{cases} \end{cases}$$

Any one of the operations mentioned in Section 18.1 is used with the cost associated with it. The edit distance of to words PARK and SPARKE is given in the following matrix

	P	A	R	K
S	1	2	3	4
P	1	2	3	4
A	2	1	2	3
R	3	2	2	2
K	4	3	3	3
E	4	3	3	3

Table 1: Edit distance matrix of two words Spark and park

The total edit distance of sparke and park is 3 (value in the right-most bottom cell of the matrix). The time complexity of the edit distance algorithm is $O(m \times n)$ where m and n are the lengths of *word* w_1 and *word* w_2.

18.2 Bayesian Probability

Once the candidate list of words with corresponding edit distance is obtained, then Bayesian probability as mentioned in Equation 2 [10]. $P(C|T)$ in the Equation 2 is

referred as *posterior* probability calculated from $P(T|C)$, the *likelihood* of the typo given correct word, $P(C)$ the prior of correct word from the dictionary or database, and $P(T)$, the evidence of the typos i.e, the mis-spelled word.

$$P(C|T) = \frac{P(T|C) * P(C)}{P(T)} \tag{2}$$

The likelihood ratio is computed by the ratio of bi-grams or tri-grams of the typo and the correct words. This can be explained with the example of two words say "acress" and "actress". The tri-gram sequence of "acress" and "actress" are,

- acress (T): acr,cre,res,ess
- actress (C): act,ctr,tre,res,ess
- $|T \cap C|$: — res,ess — = 2
- $|T \cup C|$: — acr,cre,res,ess,act,ctr,tre— = 7
- Likelihood term : $P(T|C) = \frac{|T \cap C|}{|T \cup C|} \frac{2}{7}$
- Prior term: $P(C) = \frac{n}{N}$ where n is the frequency of C in corpus and N is the total number of words in the corpus.

18.2.1 Smoothing

The major problem in calculating prior is to handle the unseen words. If a word which is no tin the database or corpus used as dictionary is encountered the whole *Posterior* term goes to zero because of the zero prior. In order to handle this the process of smoothing is invoked. The most commonly used smoothing technique is the **Laplace Smoothing** also known as **add-one smoothing**.

- **Laplace Smoothing** is the process of adding certain value to the prior of every word found from the corpus. The smoothed prior is calculated as

$$P(C) = \frac{n + 0.5}{N + 0.5V}$$

The other alternate ways of smoothing also present. One such method is **Good Turing Discounting**.

- **Good Turing Discounting:** Here the unseen classes are termed as *zero training* classes and Good Turing probability for zero training classes are calculated as

$$P_{GT} = \frac{N_1}{N} \tag{3}$$

where N_1 is frequency of classes or sub-population with minimal frequency identified so far, and N is the sum of all class frequencies identified so far. Once P_{GT} is computed the probabilities of each class is re-computed as follows:

$$\hat{C} = (C+1)\frac{N_{C+1}}{N_C} \tag{4}$$

$$P(C) = \frac{\hat{C}}{N} \tag{5}$$

where C refers to the class and N_C refers to the frequency of class C.

The candidate word with maximum posterior is considered as the correct replacement for the mis-spelled word.

18.3 Phrase level spell check

Edit distance is used to identify the appropriate candidate list for the mis-spelled word. Once the candidate list is generated the possible phrases are generated with all the candidate elements and the probabilities of all possible phrases along with the context are computed from the corpus as

$$P(w|c_{-k}.c_{-k-1}.\cdots c_{-1}.c_1.c_2\cdots c_k) = \frac{P(c_{-k}.c_{-k-1}.\cdots c_{-1}.c_1.c_2\cdots c_k).P(w)}{P(c_{-k}.c_{-k-1}.\cdots c_{-1}.c_1.c_2\cdots c_k)} \tag{6}$$

where C_k are the collocation words. Collocation words are the words located in close proximity with the considered word. But context words are the need not be close in spatial terms. It can be anywhere in the document and the it is assumed that if we find that word, it is more likely to find the other word related to that in the document.

This scenario of context based spell checking can be explained with following example. Eg: From the eath to the moon

- Candidate List: death, earth
- Possible phrases:
  - From the death to the moon
  - From the earth to the moon
- Calculate the probability of each phrase from the corpus. The phrase with maximum probability is considered as the correct phrase.

The other scenario is handling the *confusion* words. Consider the example **"*peace* of cake"**. Here though the spelling of "peace" is correct, it does not suit for the phrase. In such cases, we have to compare the phrase with the list of confusable words. All possible sentences with confusion words are formed and the sentence or phrase with maximum probability is chosen as the correct phrase as usual.

19 Information Retrieval

Information retrieval (IR) can be broken down to prior and likelihood of the documents involved. Different formalism for information retrieval are there. A non-probabilistic approach for information retrieval is a kind of string matching. Estimating the relevance between the query and a documents matches the representation of our intent to a representation of all possible intents, that computationally leads to the documents or parts there of. While giving a query instead of giving a well-formulated sentence one can form a bag of words with core and relevance words as shown below

$$Q : w_1, w_2, w_3, \cdots, w_n \tag{7}$$

Words from the documents are also considered as the bag of words as D : $w_{D1}, w_{D2}, \cdots, w_{D_n}$. Different documents are represented as D_i and the query is represented as Q. The total number of vocabulary is represented as $|V|$. The documents and words can be treated as vector space models with documents as basis vectors and the words in the space spanned by the documents. The vice-versa is also possible as in the Figure **??**. The main aim is to avoid the correlation as much as possible such that the basis vectors are the orthonormal basis vectors. The relation

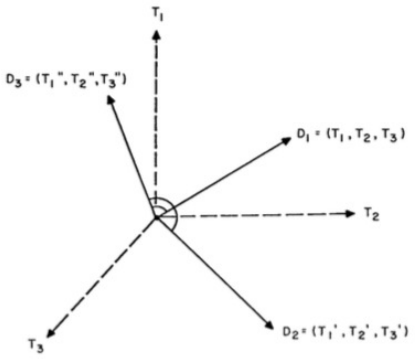

Figure 9: Term and document representation in vector space models

between the words(terms) and document is represented by the $term - dcument$ matrix (Equation 19).

$$\begin{array}{c} \\ D_1 \\ D_2 \\ \vdots \\ D_n \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & \cdots & w_{|V|} \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & 1 \\ 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \cdots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix} \end{array} \tag{8}$$

where n is the total number of documents. The number of occurrences of a term is a document is termed as $term\ frequency$. The term frequency is considered as the two different measures namely local measures and global measures. The *inverse document frequency* (IDF) for a word k is defined as mentioned in the equation 19.

$$IDF_k = 1 + log_2 \frac{n}{d_k} \tag{9}$$

where the d_k is the document frequency of k. The saturation of the IDF values leads to the word sense dominance (WSDo) which is a property to be avoided. To measure the relevance of a term to a document, both local and global measures are used as follows:

$$\begin{aligned} measure\ of\ a\ relevance\ of &= local\ measure \times global\ measure \\ a\ term - document &= tf \times idf \\ &= tf \times log\left(\frac{N}{n}\right) \end{aligned}$$

The document vector is typically longer in size because the words in the document is redundant and more. In query we will not have redundant information. If the document vector and query are of same size then we might have gone for Euclidean measure. But because of this size variation, we usually use cosine measure.

$$cos(\theta) = \frac{\vec{Q} \times \vec{D}}{||\vec{Q}||.||\vec{D}||} \tag{10}$$

where Q is the query and D is the document. The sophisticated similarity measures are used to avoid repeating same operations. If search of 'cat' occurs twice, it must use the results of the previous computation. This can be handled with any one of the following techniques

- Linguistic resources (dictionary) and
- background knowledge (Wikipedia)

The functional words has to be eliminated based on the linguistic knowledge like stemming, document frequency, word frequency etc. Evaluating an IR system is hard because of measuring the utility using similarity. Similarity is used as a measure to proxy the utility. Usually the queries are subjective and the retrieved information may not be of same usefulness for all. For example the query "Jaguar" may return both the car and the animal. Both both may not be interested to all the users. Out of top n ranked document in retrieved solution for a query we have to know whether one document is related or complemented with the other documents. Subjectivity is the personalized but the most important additional dimension. The suggested list of items for each user in amazon. The utility includes the time factor, search space factor of query etc. The IR gives the result on the basis of the rank order. The effectiveness of IR depends on the number of retrieved results (RET) are relevant (REL) and the number of relevant results out of all retrieved documents [11]. These information are measured through

$$\begin{aligned} Precision &= \frac{|RET \cap REL|}{|RET|} \\ Recall &= \frac{|RET \cap REL|}{|REL|} \end{aligned}$$

The recall score is difficult to calculate because its tough to calculate the relevance factor in the real sample space. The precision is typically easier than recall. The human relevant judgment (HRJ) is done by mapping a set of documents for the set of query manually. This is done by a group of people since the query is subjective. The HRJ is necessary to calculate the recall. Given a TREC topic, the goal of this task is to locate relevant and new information from a set of documents. P_k is defined as the precision at k preferring only top k of retrieved documents. The harmonic mean of precision and recall is called as F-measure or F1-score determined by the Equation 19.

$$F_1 = 2.\frac{precision.recall}{precision + recall} \tag{11}$$

The preference over the precision and recall is application based. Recall is preferred for recall centered applications like the criminal case, hotel search, patent search. etc. Precision-centric work like spell checking, google search, yellow pages uses precision. The trade-off between precision and recall is achieved by using weights based on the application instead of using F-measure always. Based on the degree of relevance of a retrieved document with respect to query and rank of the relevance documents are determined. Some other measures like average precision , and mean-average precision are also used.

19.1 Shortcomings of IR

The major problem of IR is the curse of dimensionality [12]. The bag of words used to represent the query or document do not preserve the sequence information. To model the similarity or relevance utility requires to handle these shortcomings. The number of topics and or concepts must be significantly lower than the number of words in order to handle the dimensionality curse. Relevance of features to a documents are decided by adhoc weights given by the IR committee.

20 Word Sense Disambiguation

Word sense disambiguation (WSD) aims at making explicit meaning lying under the words in the context of computational manner. For example, the word "bass" gives different meaning in the following two sentences. 1. He caught the bass fish 2. She played bass guitar. Identifying the actual meaning of a word with respect to the sentence apart form the literal/linguistic meaning is really hard problem which is compared with the artificial intelligence problems [13]. The problem definition of WSD is itself difficult because of

- different problem formalization
  - domain-specific/generic
  - granularity (slight variation / homonymy type)
  - one word per sentence / all words of a sentence
  - enumerate finite set / rule-based generation
- the knowledge source
  - corpora

- labeled or unlabeled data
- structure sources like thesauri, dictionaries

Knowledge acquisition bottleneck: Though the importance of WSD is high the lack of application to the real world tasks makes this problem a harder problem to understand and solve. Since the internet had gone viral, the WSD is mandatory to treat the mass of information scattered across billions of servers by automated mechanisms. The manual creation of data source is time consuming and expensive. It has to be changed every time manually in case of any strategical changes. This repetition may also lead to further disambiguation. This fundamental problem of source and time in WSD knowledge base creation is termed as knowledge acquisition bottleneck.

If WSD is built based on lexico-syntactic analysis, the WSD problems are resolved only at the surface level. It will fail to identify relevant information formulated with different wordings. It also fails to discard the irrelevant documents.

20.1 Applications of WSD

Though the applications of WSD are not well-experienced in real world, it does not mean that WSD is not having any applications in NLP. The following are few well-known applications of WSD under development to improvise the effectiveness.

- Semantic web: web-search based on well-defined information or meaning of the given query.
- Parts of speech tagging.
- Identifying the name entity relationship.
- Machine translation: English word "pen" has different meaning in Italian language like "author" and "feather".
- text categorization.
- document summarization.

Even the usage of artificial intelligence did not help to derive a generalized solution for WSD. The detail description of WSD problem can be given as this: Consider a text document (T) with 'i' different words. Appropriate subset of senses of a word w_i is defined as the mapping A from the words to senses as $A(w_i) \subset Senses_D(w_i)$ where $Senses_D(W_i)$ is defined as the set of senses of the word w_i encoded in the dictionary D.

20.2 WSD - A Classification Task

The set of senses of a word can be assumed as a class, and the words can be assumed as the data points which has to be assigned to any one of the classes based on the evidence from the "context" and "external knowledge sources". Based on the words to which the sense has to be identified the WSD problem can be divided into two tasks namely

- Targeted WSD : where a system is required to identify the sense of a set of words usually that occurs once in a sentence. Supervised systems are used to train the manually labeled sentences and classifies the unlabeled sentences.
- all word WSD : is a open set problem where the system has to distinguish all the words in the given text or document. Supervised system suffers from data sparseness and thus semi-supervised learning systems are used to develop this system.

Four main elements of WSD are listed as

- selection of word senses (determining sense inventory)
- use of external knowledge sources
- representation of context and
- selection of automatic classification methods

The sense inventory partitions different range of meaning of a word into its senses.

20.3 Selection of word-senses

Granularity of WSD problems varies based on the distinctions of the sense of the words in interest. The two types of approach to solve the granularity problems are

- Enumerative approach
- Generative approach

Enumerative approach: This approach can be explained with an example. Let us consider the usage of knife in two different senses. 1. She cut the vegetables with knife 2. He got killed after stabbed with knife. In first sentence the word knife is used as a tool and in the second sentence it is used as a weapon. In this approach all possible senses of a word are listed in a set and each possibility is enumerated with a number. The enumerated senses of a word forms the *sense inventory*.

The major disadvantages of this system is determining the optimal granularity for better results, and organization of dictionary senses. Apart from this enclosing all senses for every change of a word is also not possible.

Generative approach is the approach where the senses are generated by a set of rules. This rules captures the generality among the senses. These set of rules are called as "qualia roles" where the semantic featured of the senses of an entity is structured. The parts of speech (POS) and lexicon (L) are associated to get the senses of a word from the dictionary D as

$$Senses_D : L \times POS \leftarrow 2^C \tag{12}$$

where 2^C describes the power set of the concepts of the word under consideration. A word w is said to be *monosemous* when it has only one sense. Senses of a word which can convey unrelated meanings are called *homonymous*.

20.4 External knowledge sources

The resources for WSD can be broadly classified as shown in the figure **??**. The thesauri, ontologies and machine readable dictionaries forms the structured resources of knowledge. Thesauri provides information about relationships between words. Ontologies give the specifications of concepts of specific domains of interest. Raw corpora, sense-annotated corpora and collocation resources forms the unstructured resources. Brown corpus with a balanced collection of million words and Wall street journal corpus with 30 million words are few examples of raw corpora. SemCor is the largest and most used sense-tagged corpus. There are multiple corpus which tags the senses of different languages. This can be used in machine translation. Collocation resources has the tendency to register the words that commonly occur with other words. This resources are mainly derived from web data.

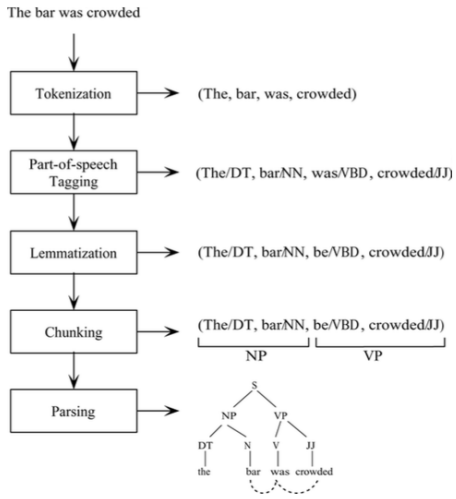

Figure 10: Preprocess steps for formatting unstructured source

20.5 Representation of Context

Unstructured source of information are pre-processed as displayed in the diagram. Tokenization splits up the text into set of tokens. Parts of speech tagging tags every word with its parts of speech usually noun, verb, determinant, and adjectives. Then lemmatization reduces the morphological variations to the basic form and chunking divides the text into syntactically correlated parts [14]. Finally parsing is used to identify the syntactic structure of the sentence. Usually a set of features are chosen to represent the context. The commonly used features are

- local features: represents the local context of the word usage based on the small number of words surrounding the target word.
- topical features: is the complete opposite of local feature and describes the general topic of text or discourse.
- syntactic features: explains the syntactic cues and argument head relations between the target word and other words within the same sentence.
- semantic features: explains the semantic representation based on the sense of words in the context.

20.6 Types of Classification

As we discussed already WSD can be considered as the classification task and three types of classification systems are:

- Supervised WSD
- Unsupervised WSD
- Semi-supervised WSD

20.6.1 Supervised WSD

This approach uses machine learning techniques to learn the class properties from labeled training data. It encodes the features of same class label together and check for the set of such features for the unlabeled data. It assigns the unlabeled data to the class that gives maximum similarity of features. Many approaches like

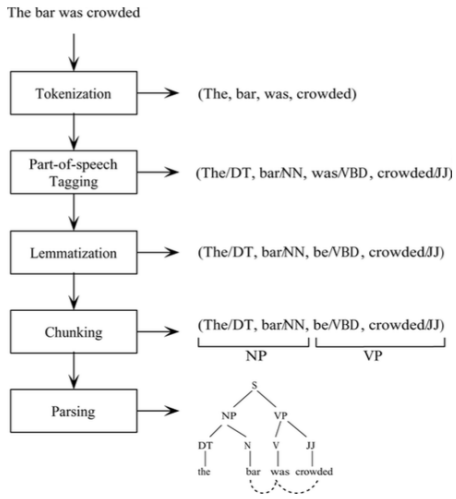

Figure 11: CHANGE THE DIAGRAM

- Decision list
- Decision tree
- Naive Bayes approach
- Neural network and
- SVM

comes under the supervised classification of words and its senses.

- **Decision List:** A given word w is represented as a feature vector and the decision list is prepared. A decision list is an ordered set created by a set of weighted "if-then-else" rules. The feature set (sense) with highest score for the input feature vector (word) of a test word is assigned to that sense (feature set). This is the basic way of scoring to assign a word with its senses list.

$$\hat{S} = argmax_{S_i} \in Senses_{D(w)} score(S_i) \tag{13}$$

  Many variations of this scoring is found in literature including probability, logarithms etc.
- **Decision Trees:** are the predictive models used to represent the classification rules in a tree structure where the leaf nodes have the classified words. Each internal node represents a test and each branch represents the outcome of the test. Based on the leaf nodes the predictions are made. The tree is based on the result of yes or no for the rules represented in the internal nodes. The empty leaf node indicates that no choice is made based on the rule in its parent node.
- **Naive Bayes:** is the probabilistic approach based on the application of Bayes' theorem [15]. It calculates the conditional probability of different senses of the word using

$$\hat{S} = argmax_{s_i \in Sensed_D(w)} P(S_i) \prod_{j=1}^{m} P(f_j|S_i) \tag{14}$$

and the sense with maximum $\hat{S}$ is considered as the appropriate sense of the word.

- **Neural network:** is the computational model generated by the interconnection of artificially created nodes called neurons. It uses the features of input words to partition the contexts into non-overlapping sets corresponding to the desired outputs. The neuron producing the desired output has larger activation value than others. The sample neural network is shown in the Figure 12.

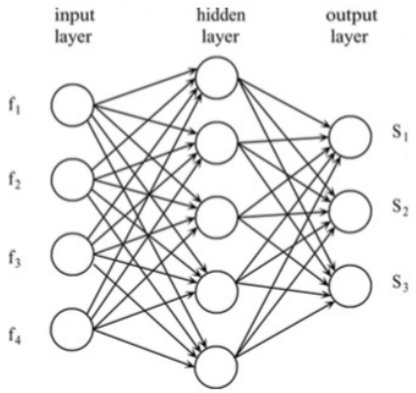

Figure 12: A sample neural network with input, hidden and output layers

- **Support Vector Machines (SVM):** SVM is based on cover theorem which says on projecting the lower dimension to higher dimension leading to better classification of the data as shown in the Figure 13. The SVM aims at finding

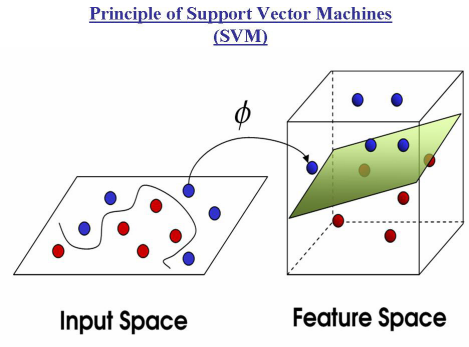

Figure 13: SVM projection from 2D to 3D

the separating hyperplane with margins. The data points that lies on the margin hyperplanes of the separating hyperplane are called support vectors. The separation is shown in the Figure 14.

20.6.2 Unsupervised WSD

This method completely uses the unlabeled data. It does not uses the manual tags to identify the sense of word in the context. The main advantage of this method

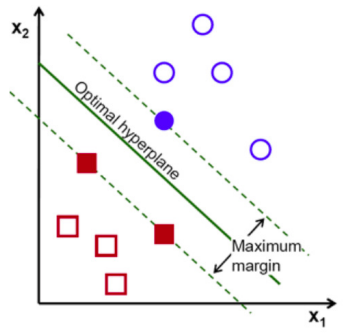

Figure 14: SVM with separating hyperplanes and support vectors

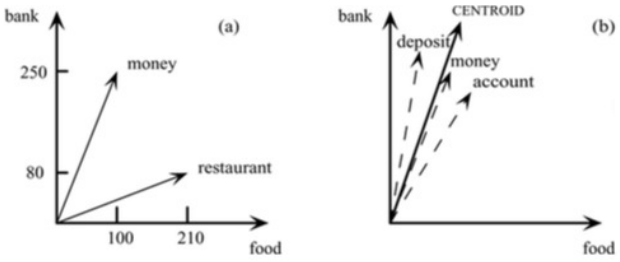

Figure 15: Figure (a) is an example of two vectors. Figure (b) is a context vector for stock, and the centroid of the vectors of word

lies in the fact that it overcomes the knowledge acquisition bottleneck discussed in the beginning of the Section 20. The basic idea of this process is that a word and its neighbor will have same sense. This is similar to K-nearest neighbor algorithms concept [16]. It is based mainly on the cluster allocation. It identifies the clusters instead of believing the assigned class. The different methods of unsupervised WSD is listed and explained one by one below.

- **Context clustering:** A given sentence is taken as a bag of words and treated as context vector. Similarity between the two context vectors are mapped to the number of features. The similarity of the two vectors can be calculated by cosine similarity as shown in 20.6.2.

$$sim(u,v) = \frac{u.v}{|u|.|v|} = \frac{\sum_{i=1}^{m}}{\sqrt{\sum_{i=1} mu_i^2 . \sum_{i=1}^{m} v_i^2}} \tag{15}$$

As the number of words increases the set of vectors corresponding to the words goes to infinity. The context clustering results in the word vector matrix as shown below

$$\begin{array}{c} \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & \cdots & v_n \end{array} \\ \begin{pmatrix} 1 & 0 & \cdots & 1 \\ 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \cdots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix} \end{array} \tag{16}$$

Latent semantic analysis (LSA) is based on this context clustering.

- **Word clustering:** Word clustering is the clustering process between two words say w_i and w_j. This clustering is a 3 step process. The first step is the $S - step$ where the similarity between the two words is calculated as $sim(w_i, w_j)$ where the information in the feature vector. The second step is the $C - step$ determining the set of clusters for each word. This $C - step$ commits the word with each cluster. The last step is the $D - step$ known as discrimination step $w_i \in E$ the set of clusters.
- **Co-occurrence graphs:** Let G be the graph with V vertices and G edges. The words are the vertices and the co.occurrence syntactic relation are the edges. Let w be the target or ambiguous word and G_w is the local graph around w. The adjacency matrix has to be created and normalized to get the Markov chain. The markov clustering algorithm has two steps namely $E - step$ (Expectation step) and $I - step$ (Inflamation step). HyperLex is the hyper lexicon where the similarity between the two words w_i and w_j as

$$w_{ij} = 1 - max P(w_i|w_j), P(w_j|w_i) P(w_i|w_j) = \frac{freq_{ij}}{freq_j} P(w_j|w_i) = \frac{freq_{ij}}{freq_i} \tag{17}$$

Minimum spanning tree (MST) is another representation generated from the co.occurrence graph to remove the ambiguity in the graph. To generate MST first arrange the ascending weights such that the co.occurrence graphs order is preserved.

21 Semantic Relatedness

TO estimate the similarity between a pair of words, semantic relatedness is used to identify something apart from the literal meaning of both the words. The set of possible synonyms for a word collected from external world knowledge is called as Synonym-set or simply **synset**. Each synset is associated with a small description called as **gloss**. Synsets are easy to encode because they are direct words. Synset is referred using ID's. Each synset is associated with a series of glosses. Preferable cardinality of synset to be greater than one. The main advantage of this is that we can have more words and context. The normal functional relations like symmetry, reflexive etc. The estimation of semantic relativeness between words is measured is determined by three different ways as given below.

- Wordnet-based
- Distributional similarity based on corpora
- Introspective distributional similarity

21.1 WordNet based method

Word sense disambiguity (WSD) is used to come -up with appropriate synsets for each word from the synset word. To find the relationship between synsets we use the taxanomy of wordnet. One way to do without WSD is find out all possible similarity for w_i from the synset-word relationship. If the synsets of two words are more related, then the words are said to be related too. The measures of similarity between synsets can be measured in more than one ways. Two such methods are

- Path-based measures
  - Wu-Palmer Measure (WPM)
  - Liu similarity measure
- Information content based measures

21.1.1 Path-based measures

- **Wu-Palmer Measures (WPM):** The sum of distance to the lowest common ancestor (LCA) in a tree structure with respect to two leaf nodes (word1 and word2) under consideration and the sum of the distance to the root as mentioned in the Equation

$$WPM = 1 - \frac{sum\ of\ the\ distance\ to\ LCA}{sum\ of\ the\ distance\ to\ the\ root} \tag{18}$$

  Bigger the WPM measure, the LCA measure is the similarity measure and it must be preferably higher and the root distance measures the distance which must be preferably lower [17]. The similarity between two classes C1 and C2 is measured by *Wu-Palmer similarity score* as

$$\begin{aligned} sim(C1, C2) &= \frac{2 \times N_3}{N_1 + N_2 + 2N_3} \\ &= \frac{N_1 + N_2}{N_1 + N_2 + 2N_3} \end{aligned}$$

  where N_1, N_2 are the number of links from $C1, C2$ to their most specific common superclass and N_3 a=is the number of links from the most specific common superclass to the root.
- **Liu similarity measure:**

$$sim(C1, C2) = \frac{e^{\alpha L} e^{\beta H} - e^{-\beta H/}}{e^{\beta H} + e^{-\beta H/}} \tag{19}$$

where $\alpha \geq 0, \beta \geq 0$ are constants, L is the shortest path between C_1, C_2 and H is the depth in taxonomy of the most specific common concepts in a non-linear function.

21.1.2 Information theoretic based measure

This method is proposed to overcome the disadvantages of the path-based measures discussed in Section 21.1.1 like similar treatment for the words at same level and the skewness based on the tree structure. The tree is constructed manually and it may not reflect all world knowledge. This measure also uses LCA as

$$IC(S_1, S_2) = log\left(\frac{1}{P(LCA(S_1, S_2))}\right) \tag{20}$$

The denominator term is estimated based on the corpora. The disadvantage of this method is the word in search may not be ithere directly but many related synsets will be there. Such similar synsets are also needed to be considered to calculate the probability.

21.2 Distribution similarity based on corpora

Wikipedia is a generic resource which explains the representation of knowledge and how the data related data are linked to it. The reference to one page may trigger many related memories leading to a search for the related topics. For example if we search for bishop it may lead to rook so that one may know that too. But both are related through a game named Chess.

21.2.1 Explicit semantic analysis

Explicit semantic analysis (ESA) is an analysis method used to identify or handle the distributional similarity based on corpora like Wikipedia. This analysis is based on *vector-space modeling* [18]. ESA is used for two tasks like

- Estimating relativeness between two pieces of text
- estimating relativeness between two words

Input texts are represented as weighted vectors of concepts called *interpretation vectors*. Semantic relatedness of texts are computed by comparing the vectors defined in the concepts space. The explicit meaning is intended to capture the concepts grounded in human cognition. The inputs are given in the form of plain text and the conventional classical algorithms are used to rank the concepts represented by the document. Each document is represented as a vector of words and conventional classification algorithms to rank the concepts represented by these documents. Each documents is assigned with a weight vector identified by the TFIDF discussed in Section 19. The vector space modeling of the words and its concepts will look very similar to the Figure 15. The vector space model can be spanned either by word vectors as basis vectors or concept vectors as basis vectors.

Mapping of words from word space to concept space are done offline irrespective of query using inverted index. This mapping is working on the basis of information retrieval on classification data. If a single words leads to many vectors in concept space, take a centroid of all those vectors and use it as the equivalent of that word in the concept space. The input document is treated as a bag of words $(w_1, w_2, w_3, \cdots, w_n)$ and mapped to $(v_1, v_2, v_3, \cdots, v_n)$. Let k_j be the TFIDF values of concept c_j then the page rank of each document is calculated according to this TFIDF. Relevance measure of a concept c_j to the document is measured as the sum of products of v_i and k_j.

21.3 Distribution (Introspective) Approach

Estimated semantic analysis has the potential to do dimensionality reduction. Similar to spell check application this too requires feature selection [19]. Word-word relation gives the ability to compress the dimension and relate it to a cluster. The basic idea of this distributional approach is no external knowledge like wordNet or wikipedia etc. Minimum description length (MDL) has connection with Bayesian classification/inferences. MDL is the information theoretic approach whereas the Bayesian classification is the probabilistic approach. MDL concentrates on minimizing the length of encoding. Bayesian classification will maximize the posterior. Sometimes during the training of models we will not concentrate on over training to avoid over-fitting problem. Decision trees are used to define hypothesis. FUlly grown decision tree is the over-fitting kind of problem while shallo decision tree has the risk of over-generalization. KL-Divergence is the asymmetric measure between two probability distributions. Here KLD is used to measure the distributional similarity between two words. The KLD between two distribution P and Q are defined as in the equation given below.

$$D_{KL}(P,Q) = \sum_{x} P(x) log \frac{P(x)}{Q(x)} \tag{21}$$

KLD gives expected number of bits to represent the samples from P using Q. This measure is based on entropy. Entropy is a measure of uncertainty and it is used to assign an item x to any one of the classes under consideration. For a subset with all examples belonging to one class has zero entropy. If number of examples belonging to the positive and negative classes are same then the entropy will be 1. Usually entropy varies between 0 to 1. Entropy of a set S with (+,-) classes is given as

$$H(S) == P_{+} log_2 P_{+} - P_{-} log_2 P_{-} \tag{22}$$

Introspective measures are mainly based on orders of co-occurrence patterns. Let us consider two words A and B. If both occurs in same document . Then they are said to be in first order co-occurrence. If those words are connected through some other word say C in two different documents then A and B are said to be second order co.occurrences and so on. The semantic relatedness between words using co.occurrences of words is done if it is a weighted graph. Semantic relatedness is calculated as the sum of all co-occurrences between a pair of words.

Let us assume a set of words forming a directed graph as shown in figure given below. If 'B' is used in definition of A,D,C the edges will be like as in graph. The word with more incoming edges will be the foundational word. Or, the words of ancient origin are considered as foundational words.The alternate way is to prefer topological sort using page rank algorithm which helps in calculating the age of acquisition of words in child knowledge.

22 Circularity Problem

Documents are similar, if they have similar words. Words are similar if they occur in similar documents. The previous sentence defines a problem of circularity. The problem where something is needed to define one thing and that one thing is needed to define that same thing. The circularity problem is seen everywhere in NLP applications like word sense disambiguation, page ranking, information retrieval, co.occurrence patterns etc. The circularity can be resolved by using any one of the machine learning techniques discussed in following subsections.

22.1 K-Means Algorithm

K-means algorithm is the simplest clustering algorithm to cluster the data in unsupervised learning. This classifies the data point into clusters based on *apriori*. One the cluster centroids are given the data point will be assigned to each cluster and cluster centroid is changed iteratively based on the data points. In NLP, K-means clustering is used in term-document clustering. Say the documents are represented in the space spanned by the terms. Out of many documents $\vec{x}$ in the space, it can be represented by the centroid $\vec{\mu}$ of the document cluster ω as

$$\vec{\mu}(\omega) = \frac{1}{|\omega|} \sum_{\vec{x} \in \omega} \vec{x} \tag{23}$$

To achieve this the documents to be compared are assumed to be length normalized. Ideally the clusters must not overlap to each other. The process can be completed after fixed number of iterations or until there is no much difference between the clusters formed by subsequent iterations. Since there is only a finite set of possible clusterings, a monotonically decreasing algorithm will eventually arrive at a (local) minimum.

Algorithm 2 K-means algorithm

1: {**Input:** Documents in term space or terms in document space
Output: Clusters and cluster centroids}
2: Randomly initialize k clusters.
3: Calculate distance between each data points and all the cluster centroids
4: Assign the data point to the closest cluster
5: Recalculate the mean r centroid of each cluster with assigned data points
6: Repeat the steps from 2 to 4 until termination condition.

Advantages: Robust, simple, relatively efficient. Effective when the data points are distinct and well separated.

Disadvantages: We have ti determine the number of clusters or centroids in prior. This has to be related with the actual classes or varities of data. Else the resulting clusters may not be appropriate. Fails to handle noisy data, outliers and non-linear data.

22.2 Expectation Maximization Algorithm

EM algorithm is used to handle the circularity problem. EM algorithm is used as a classifier. The EM algorithm is applied on GMMs. The weights of each Gaussian mixture component is calculated along with the mean and the variance. The

mixture weights and parameters are estimated in Estimation step (E-step) and the parameters are used to maximize the posterior probability in Maximization (M-step).

EM algorithm has a wide application range in the statistics of natural language process, such as the forward-backward algorithm in HMM, the inside- outside algorithm in PCFG, EM clustering algorithm and no supervision semantic disambiguation algorithm, which are the specific applications of EM algorithm for parameter estimation problems.

EM might look like a heuristic method. However, as we show now, it has a rigorous foundation: EM is guaranteed to find a local optimum of data log likelihood. EM is an iterative procedure to maximize the marginal log likelihood $l(\theta)$.

Step 1: The step 1 of EM algorithm (E step) is to find logarithm likelihood function $logp(X, Y|\theta)$, while given observation data set X and the current parameter set $\theta(i-1)$, the expectation value about unknown data set Y is the value to calculate the next expression: $Q(\theta, \theta(i-1)) = E[logp(X, Y|\theta(i-1))]$, here into θ is the new parameter set after optimization and makes the value of function Q increasing with the new parameter.

Step 2: The step 2 (M step) of EM algorithm is to maximize expectation value of part 1, that is next expression $\theta(i) = argmax_Q(\theta, \theta(i-1))$, this θ two steps constantly iterates, each iteration will ensure to increase the logarithm likelihood function values and ensure that likelihood function converges to a local maximum value point.

22.3 Page ranking

The Google search engine uses Page Rank algorithm to exploit the linked structure of the web. This page rank helps in computing the global importance scores that can be used to influence the ranking of search results. The rapid growth in web increased the demand for greater flexibility in ranking and page rank algorithm is proven to be effective in achieving this. Ideally each user should be able to define his own notion of importance for every query posed by him.

In principle personalized page ranking algorithm can be effective to achieve this. But it requires naive implementation and computing resources far beyond the realm of feasibility. Page rank assigns score to the web pages. A page is important if it is pointed-to by several important pages. Based on the number of link pointing a page and from who the reference is coming the weightage of a page will be altered.

From the diagram $P_4 = P_1 + p_2 + P_3$ where P is a vector that only gets scaled or stretched but it does not rotate when the operator acts on it. Page rank has the solution to the circularity problem but not the optimal one. PCA can handle circularity with optimal solution.

22.4 Singular Value Decomposition

Singular value decomposition (SVD) is used to solve the problem of circularity by using the covariance matrix as a feature matrix. SVD follows matrix diagonalization theorem splits a matrix into three where the middle one is the diagonal matrix with the eigen values and the other matrices has eigen vectors in it. If a matrix A has a matrix of eigenvectors P that is not invertible then A does not have an eigen

decomposition. However, if A is an $m \times n$ real matrix with $m > n$, then A can be written using a so-called singular value decomposition of the form $A = UDV^(T)$.

Singular values are the square root of eigenvalues. Eigenvectors computed from the covariance matrix represents the maximum energy varying directions and eigenvalues gives the magnitude of these vectors. If the eigenvalues are unique and independent the eigenvectors will be linearly independent of each other. If the covariance matrix is symmetric and positive definite the eigenvectors will be orthogonal to each other. Note that there are several conflicting notational conventions in use in the literature. U to be an $m \times n$ matrix, D as $n \times n$, and V as $n \times n$. However, the Wolfram Language defines U as an $m \times m$, D as $m \times n$, and V as $n \times n$. In both systems, U and V have orthogonal columns so that $U^(T)U = I$ and $V^(T)V = I$. In NLP SVD is performed on word-word covariance matrix obtained by the distribution of words over the documents. The words are the linear combination of the eigen values of the word-word covariance matrix.

$$\begin{array}{c} \\ D_1 \\ D_2 \\ \vdots \\ D_n \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & \cdots & w_{|V|} \end{array} \\ \begin{pmatrix} ... & ... & \cdots & ... \\ ... & ... & \cdots & ... \\ \vdots & \vdots & \cdots & \vdots \\ ... & ... & \cdots & ... \end{pmatrix} \end{array} \quad \begin{array}{c} \\ w_1 \\ w_2 \\ \vdots \\ w_n \end{array} \begin{array}{c} \begin{array}{cccc} w_1 & w_2 & \cdots & w_{|V|} \end{array} \\ \begin{pmatrix} ... & ... & \cdots & ... \\ ... & ... & \cdots & ... \\ \vdots & \vdots & \cdots & \vdots \\ ... & ... & \cdots & ... \end{pmatrix} \end{array} \quad \begin{array}{c} \\ D_1 \\ D_2 \\ \vdots \\ D_n \end{array} \begin{array}{c} \begin{array}{cccc} e_1 & e_2 & \cdots & e_{|V|} \end{array} \\ \begin{pmatrix} ... & ... & \cdots & ... \\ ... & ... & \cdots & ... \\ \vdots & \vdots & \cdots & \vdots \\ ... & ... & \cdots & ... \end{pmatrix} \end{array}$$

a) Term Document Matrix b) Covariance Matrix c) EigenVector Document Matrix

e_i is the eigen vector with the linear combination of words and hence it can be termed as *eigendocuments* . Now we get 3 more documents which are not in document collection. Now we have $D_n + 3$ documents along with the initial documents. These documents can also be represented along with the initial document (D_i).

22.5 Principle Component Analysis

Principle component analysis (PCA) also known as *u*nimodal factor analysis (UFA). PCA performs orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principle components. The number of principle components are less than or equal to number of original variables. This transformation is defined in such a way that the first principal component has the largest possible variance and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. The resulting vectors are an uncorrelated orthogonal basis set. The principal components are orthogonal because they are the eigenvectors of the covariance matrix, which is symmetric. PCA is sensitive to the relative scaling of the original variables.

Algorithm 3 Principle component analysis

1: {**Input:** Data points / terms / words
Output: Principle components and its directions}
2: Subtract mean of the data from all the data points
3: Calculate the covariance matrix
4: Calculate eigenvalues and eigen vectors of the covariance matrix
5: Choosing components and forming the feature vector

22.6 Latent Semantic Analysis

Matrix diagonalization theorem applied on a symmetric covariance matrix gives eigenvectors will be orthogonal in nature to each other. sometimes orthogonality is the essential property. Latent Semantic Analysis (LSA) is a theory and method for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of text. The underlying idea is that the aggregate of all the word contexts in which a given word does and does not appear provides a set of mutual constraints that largely determines the similarity of meaning of words and sets of words to each other. The adequacy of LSA's reflection of human knowledge has been established in a variety of ways. Word and passage meaning representations derived by LSA have been found capable of simulating a variety of human cognitive phenomena, ranging from developmental acquisition of recognition vocabulary to word-categorization, sentence-word semantic priming, discourse comprehension, and judgments of essay quality.

LSA can be construed in two ways: (1) simply as a practical expedient for obtaining approximate estimates of the contextual usage substitutability of words in larger text segments, and of the kinds of as yet incompletely specified meaning similarities among words and text segments that such relations may reflect, or (2) as a model of the computational processes and representations underlying substantial portions of the acquisition and utilization of knowledge. LSA represents the meaning of a word as a kind of average of the meaning of all the passages in which it appears, and the meaning of a passage as a kind of average of the meaning of all the words it contains. LSA's ability to simultaneously derive representations of these two interrelated kinds of meaning depends on an aspect of its mathematical machinery that is its second important property. LSA assumes that the choice of dimensionality in which all of the local word- context relations are simultaneously represented can be of great importance.

22.6.1 LSA Application

Cross language retrieval and document preparation are the major applications of LSA. Cross language retrieval will have same text in both language will give us the difference between the languages after proper model training through LSA. To make a document of sentences and there translation and LSA will come up with interesting pattern of co.occurrences.

23 Parsing

Since the human language is highly recursive and ambiguous, syntax are needed. Parsing is the process of analysing the syntax and determining the meaning. The man goals are to get the structure to validate the sentence meaning, to identify the parts of speech, to co-reference resolution within the sentence, and to get the meaning of the sentence. Inducing grammar rules in a language is a big problem. If we downsize the problem to some basic set of rules along with the probabilities alongside it. EM algorithm is used to learn the probabilities associated with rules.

The advantage of parts of speech tagging is "re-usability" but it has its limitations. The context free grammars can be used to implement parsers, and discuss chart parsing, which allows efficient processing of strings containing a high degree of

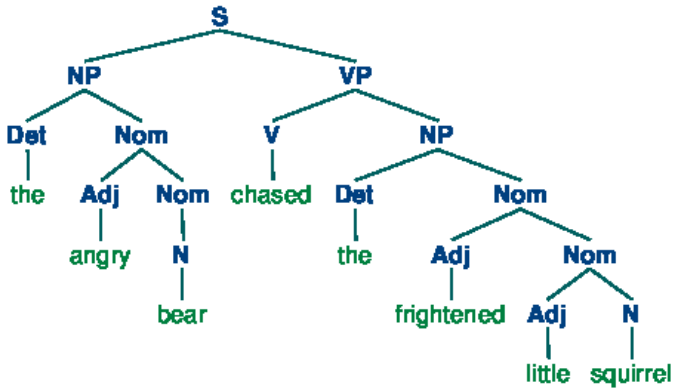

Figure 16: Example of a parse tree

ambiguity. The idea of a context-free grammar (CFG) should be familiar from formal language theory. A CFG has four compo- nents, described here as they apply to grammars of natural languages:

- a set of non-terminal symbols (e.g., S, VP), conventionally written in uppercase;
- a set of terminal symbols (i.e., the words), conventionally written in lowercase;
- a set of rules (productions), where the left hand side (the mother) is a single non-terminal and the right hand side is a sequence of one or more non-terminal or terminal symbols (the daughters);
- a start symbol, conventionally S, which is a member of the set of non-terminal symbols.

23.1 Classical Parser

Classical parsing is the process of just parsing the input sentence based on the formulated general grammars. Generally this is classified into *top-down parsing* and *bottom-up parsing*.

- **Top down parsing:** A top-down parser starts with the root of the parse tree. It is labeled with the start symbol or the goal symbol of the grammar. TO build the parse tree, it repeats the following steps until the leaves of the parse tree matches the input string.
  - At a node labeled A, select s production with A on its left-hand-side and for each symbol on its right-hand-side, construct the appropriate child.
  - When a terminal is added to the leaf that does not match the input string backtrack.
  - Find the next node to be expanded. It must have a label in non-terminal.

  The main key in this top-down parsing is to select the right production in Step - 1.
- **Bottom up parsing:** Bottom up parsing has same grammar and the procedures. But it starts at the leaves and grow in-wards. Starting in a valid state for legal first token is the first step in bottom up parsing. As an input is consumed, the state is changes to encode the possibilities. That is to recognize the valid preferences. Bottom up parsing uses a stack data structure to store both state and sentential forms.

23.2 Parse Generation

Classical parse trees are used to check the validity of a existing or formulated grammar [20]. Generating the parse tree automatically based on the given grammar is called as parse generation and it can be done in two ways namely *constituency parsing* and *dependency parsing*.

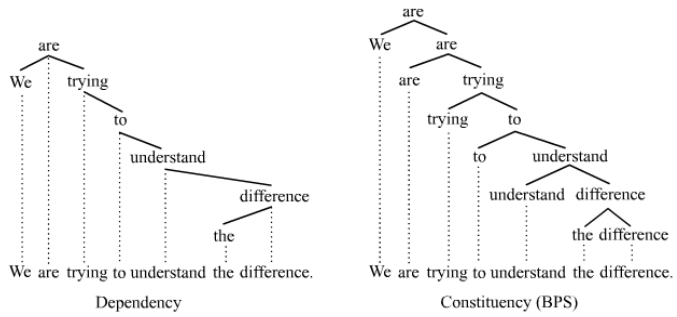

Figure 17: Constituency and dependency parsing

A constituency parse tree breaks a text into sub-phrases. Non-terminals in the tree are types of phrases, the terminals are the words in the sentence, and the edges are unlabeled. For a simple sentence "We are trying to understand the difference", a constituency parsing and dependency parsing would look like as in the Figure 17. A dependency parse connects words according to their relationships. Each vertex in the tree represents a word, child nodes are words that are dependent on the parent, and edges are labeled by the relationship. Out of these two parsers the parser which may take us close ot the assigned goal has to be chosen.

23.3 Probabilistic parser

Probabilistic parsing is using dynamic programming algorithms to compute the most likely parse(s) of a given sentence, given a statistical model of the syntactic structure of a language. As we see in previous chapter same grammar may lead to more than one parser. The problem of this ambiguity and chaos can be get-rid by probabilistic context free grammars. Probabilistic grammars are used to

- determine the sentence and
- choose the speedier parser

Tree bank: The pure grammar induction approaches tend not to produce the parse trees that people want. A fairly obvious approach to this problem is to give a learning tool with some examples of the kinds of parse trees that are wanted. A collection of such example parses is referred to as "tree bank". Because of the usefulness of collections of correctly-parsed tree banks, but by far the most widely used one, reflecting both its size and readily available status *Penn Treebank"* is used more commonly.

24 Machine Translation

Machine Translation (MT) is the task of automatically converting one natural language into another, preserving the meaning of the input text, and producing fluent

S	→	NP	VP	1.0
VP	→	Vi		0.3
VP	→	Vt	NP	0.5
VP	→	VP	PP	0.2
NP	→	DT	NN	0.8
NP	→	NP	PP	0.2
PP	→	IN	NP	1.0

Vi	→	sleeps	1.0
Vt	→	saw	1.0
NN	→	man	0.1
NN	→	woman	0.1
NN	→	telescope	0.3
NN	→	dog	0.5
DT	→	the	1.0
IN	→	with	0.6
IN	→	in	0.4

Figure 18: Probabilistic CFG

text in the output language. The ideal aim of machine translation systems is to produce the best possible translation without human assistance. Basically every machine translation system requires programs for translation and automated dictionaries and grammars to support translation.

Machine translation is handled with the assumption that while transferring the data over a noisy channel it may get corrupted or altered. To get back the original message from the received message we need to understand the channel features and the noise in it. The same scenario is used here where the analogy of channel model maps to the language model here. In figure 19 e represents English which acts as a target language and f represents French which is our source language.

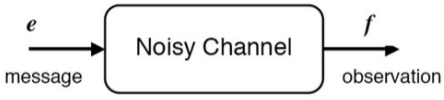

Figure 19: Noisy channel model of Machine translation

Using Bayes theorem we have to identify

$$P(e|f) = \frac{P(e).P(f|e)}{P(f)} \tag{24}$$

and the best Englllish translation for the French sentence in hand is obtained by $\hat{e} = argmax_e P(e)P(f|e)$. Now the translation problem is reduced to a language problem of identifying the *Prior* $P(e)$ and a *translation* task of identifying $P(f|e)$. IBM model is used to identify the translation task alone whereas the language model can be built using $Markovian$ based *Hidden-Markov-Model*.

Statistical machine translation handles complete data directly and the incomplete data through EM algorithm (Section 22.2). Word-pair alignments gives rise to the sentence pairs. The sentence pairs do not have the information of the word-pair alignments given rise to them. In all cases, the translation probability $P(f|e)$ is seen as the sum on all alignments of the conditional probabilities $P(f, a|e)$, where a is an alignment between the French and the English sentences:

$$P(f|e) = \sum_{a} P(f, a|e) \tag{25}$$

and the conditional probability $P(f, a|e)$ is defined as

$$P(f, a|e) = P(m|e) \prod_{j=1}^{m} P(a_1^{j-1}, f_1^{j-1}, m, e).P(f_j|a_1^j, f_1^{j-1}, m, e) \tag{26}$$

The best alignment is calculated as

$$\hat{a} = argmax_a P(a, f|e) \tag{27}$$

The translation model informs us on what sentences are good translations, while the language model ensures that these sentences are well-formed. By combining these models, we thus get better results than if we were to look directly for the sentence that maximizes $P(e|f)$.

24.1 IBM Model 3

IBM Model 3 is an example for statistical machine translation.

Figure 20: An example for machine translation

IBM model 3 is a generative model which calculated the best alignment and translation probability as discussed previously. The three important parameters needed for this model are

- Fertility parameter : $n(\phi_j|e_j)$
- Translation parameter : $t(f_i|e_{aj}$
- Distortion parameter : $d(f_{pos} = i|e_{pos} = j, I, J)$

where I and J are the total words presents in the English and French sentences respectively and i, j refers the words in the respective languages. The transformation of English sentence to French is shown in the Figure 21. Fertility parameter explains whether any word is replicated/deleted during the transformation and the translate parameter calculates the translation probability as discussed earlier. The distortion parameter measures the amount of re-ordering of the translated words needed to get a proper translated sentence.

Knowing the parameter values the probability can be calculated easily for sentence pair. If the parameters are unknown, IBM model 3 is tough to use. So we have to align words using IBM model 1 to get the best alignment and the parameters are learned using the same model. Then transition models are built using IBM model 2 and the final parameter set is used as the seed to IBM model 3 to calculate the probability.

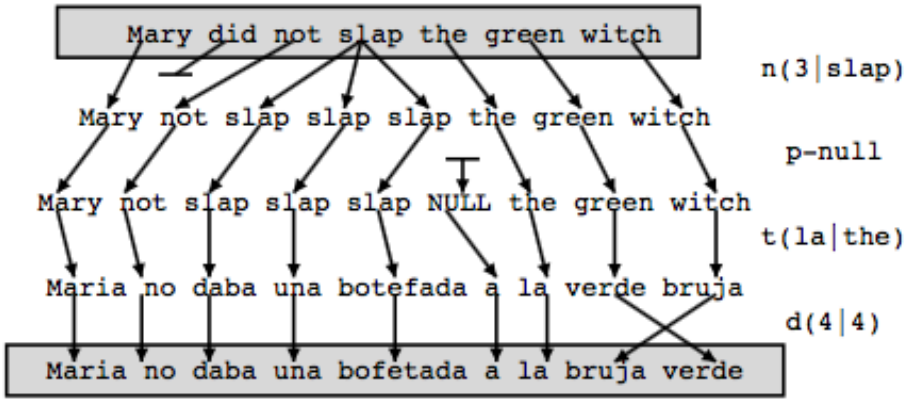

Figure 21: An transformation of English to French by the IBM Model 3 with parameter set

Chapter 4

25 Choosing the words in computer-generated weather forecasts

Weather forecasts and daily weather reports are inevitable needs in our day-to-day life to know the climatic changes. Satellite images are the source of the weather reports. These satellite images can be interpreted only by the experts. Their interpretation are so crisp and too technical for the common people to understand. Transferring the technical information listed by the weather experts from the satellite images to the understandable text format can be done by employing humans for the task. The major problem with this is the inconsistent and ambiguous way of an individual in representing the content. This can be avoided by using an automated system to do the translation of information to the complete report.

The humans have lot of ambiguity in understanding the things and there is no commonality between the understanding of two different persons. The color referred as "pink" by two persons may differ even in same lighting and with same visual perception capacity. This difference is termed as "idiolect differences" which raises an important question in implementing the computer based NLG system. The major problem due to the idiolect differences is "*How does the text produced by the system in English (or other human languages) from non-linguistic input data ensures that the texts they produce are correctly interpreted by their readers?*".

This can be ensured by a common set of rules that are needed to be followed in choosing the words and phrasing the sentences and the resultant must be consistent and understandable to all type of people without any ambiguity. In [21], an automated weather report generator named SUMTIME-MOUSAM is discussed elaborately. SUMTIME-MOUSAM is the text-generator based on NLG concepts voted as the best choice by users than the human-authored texts.

25.1 Stages of text generation

SUMTIME-MOUSAM is generated after 3 different stages of detailed analysis of the content to be translated and the content to be presented in the generated weather-reports. The three important stages are

- Document planning
- Micro-planning and
- Surface realization

Document planning: decides on what content to be included and in the structure of that content in the report. Mainly how the numeric information are going to be communicated as the text. A linear segmentation algorithm is used for this purpose which uses the pragmatic analysis of appropriate content for a weather forecast report. The structure of the report will be based on a human understandable schema or template on which the transcribed numerical information are included to make it readable and understandable.

Micro-planning: plans how subtle the content and structure must be expressed linguistically. This step uses the NLP concepts like *lexicalisation, aggregation and expression generation. Lexicalization* determines which words are to be used to represent the information and *aggrgation* decides how to distribute the information across the sentences. And the last task of expression generation deals with the problem of referring back the entities introduced in the generated text.

Surface realization: This phase generates the actual text according to the decisions made in previous stages. The key problem in this phase is to generate the special rules to generate the text in the weather sub-language called "weatherese" instead of conventional English.

25.2 Word choice

Human forecasters usually use their own choice of words and lexical rules. When it comes to automatic method with common representation of texts, the possible problems are explained with few examples.

Time	Wind dir	Wind speed 10 m	Wind speed 50 m	Gust 10 m	Gust 50 m
06:00	W	10.0	12.0	12.0	16.0
09:00	W	11.0	14.0	14.0	17.0
12:00	WSW	10.0	12.0	12.0	16.0
15:00	SW	7.0	9.0	9.0	11.0
18:00	SSW	8.0	10.0	10.0	12.0
21:00	S	9.0	11.0	11.0	14.0
00:00	S	12.0	15.0	15.0	19.0

Figure 22: Part of an input data set from SumTime-Mousam

Consider the table represented as the Figures 23 generated from the information shown in the Figure 22. Each sentence in Figure 23 leads to a serious number of questions like

- Should the direction "West" be expressed as W or W'LY?
- Should the speed 8 knots be expressed as 8 or 08?
- Should the verb "backing" or "becoming" be used to describe the change in wind direction?
- Should the time pharse "by evening", "by late evening", or "by midnight" be used to express the time 0000?

Field	Text
WIND(KTS) 10 M	W 8–13 backing SW by mid afternoon and S 10–15 by midnight.
WIND(KTS) 50 M	W 10–15 backing SW by mid afternoon and S 13–18 by midnight.
WAVES(M) SIG HT	0.5–1.0 mainly SW swell.
WAVES(M) MAX HT	1.0–1.5 mainly SW swell falling 1.0 or less mainly SSW swell by afternoon, then rising 1.0–1.5 by midnight.
WAVE PERIOD (SEC)	Wind wave 2–4 mainly 6 second SW swell.
WINDWAVE PERIOD (SEC)	2–4.
SWELL PERIOD (SEC)	5–7.
WEATHER	Mainly cloudy with light rain showers becoming overcast around midnight.
VISIBILITY (NM)	Greater than 10.
AIR TEMP(C)	8–10 rising 9–11 around midnight.
CLOUD (OKTAS/FT)	4–6 ST/SC 400–600 lifting 6–8 ST/SC 700–900 around midnight.

Figure 23: Extract from forecast generated by SumTime-Mousam, from the input data set partially shown in Figure 22.

TO resolve this chaos, humongous human written forecast data are collected and analyzed. This analysis leads to a interesting observation that the data-to-text mapping depends on contextual factors like

- Preferences of individual writer and
- Linguistic context

Preferences of individual writer refers to the usage of different words by different persons to represent same chunk of data. The linguistic content represents the choice of word is influenced by linguistic factors such as the position of a word in a sentence.

25.3 Choice of time phrases

A set of rules are needed to express which time phrase should be used in a weather forecast to communicate a numerical time from the input data file. This is done by an algorithm which first learns the way of representation from the human-forecaster' s text and then uses that to generate the rules. This analysis is done on the aligned corpus, where the content of the report as well as the phrases used by the forecasters in that content are aligned like a parallel corpus. The aligned corpus is generated from the normal data as follows.

- All similar statements that covers a period of 24 hours or less is collected
- These statements are parsed and the linguistic structure of the wind statements are tuned.
- Each phrase from the corpus corresponding to those time formats are aligned to the tuned linguistic structure.

Tests on time phrases with unambiguous meanings (such as by midday) suggested that the alignment process was 86% accurate.

25.4 Classifier Analysis

Once the alignments are done, the next problem of what to choose when pops up? A classifier can handle this issue by tagging each scenario with certain probabilities to the possibilities in the database. The machine learning algorithms are used to learn classifiers which predicted which time phrase would be used in wind phrases extracted from the corpus. The classifier was trained only on wind phrases which had been successfully aligned with the data file. Similar to wind features, different features can also be aligned and the classifiers are trained to meet the end-users requirement. The other set of features that can be used are

- semantic features: information from the data file, such as the actual wind speed and direction
- author feature: which forecaster wrote the text;
- collocation features: the preceding and subsequent words in the text
- repetition feature: the previous word of this type (time phrase) in the text
- surface features: the length and position in the sentence of the phrase
- temporal features: when the forecast was issued, and how far in the future the prediction was from the forecast issue date.

25.5 Summary

People considerably vary with the usage of words as well as in the interpretation of words. Since we have crude understanding of our own language this variations do not affect the understanding. But when it comes to the system, what people mean and think with every word has to be represented clearly. SUMTIME-MOUSAM, produced texts avoided the words which only occurred in one idiolect, and words

whose meanings varied in different idiolects. But despite these flaws, human subjects still considered SUMTIME- MOUSAM's texts to be more appropriate than human-written texts. It is possible to prepare the more generic report as well as the more idiolect specific reports. The texts generated by the NLG systems are more unambiguous and cheaper compared to the text reports produced by the human writers.

Acknowledgements
I sincerely acknowledge to Prof. Sutanu Chakraborti for teaching us the concepts of NLP and guiding us in a righteous way to develop the subject knowledge that we acquire through the NLP classes. The cited articles and web-pages are acknowledged by referring them in reference section.

References

1. Nadkarni, P.M., Ohno-Machado, L., Chapman, W.W.: Natural language processing: an introduction. Journal of the American Medical Informatics Association : JAMIA **18**, 544–51 (2011). doi:10.1136/amiajnl-2011-000464
2. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. Prentice Hall Press, Upper Saddle River, NJ, USA (2009)
3. Anjali M K, B.A.: Ambiguities in natural language processing. International Journal of Innovative Research in Computer and Communication Engineering **2**, 392–394 (2014)
4. Taylor, M., Esbensen, B.M., Bennett, R.T.: Children's understanding of knowledge acquisition: The tendency for children to report that they have always known what they have just learned. Child Development **65**(6), 1581–1604 (1994)
5. problem in NLP, H.: http://nlp.abodit.com/home/nlp-is-hard
6. Morphology:: https://en.wikipedia.org/wiki/morpheme
7. Shemtov, H.: Ambiguity management in natural language generation. PhD thesis, Stanford, CA, USA (1997). AAI9901643
8. Taylor, M., Esbensen, B.M., Bennett, R.T.: Children's understanding of knowledge acquisition: The tendency for children to report that they have always known what they have just learned. Child Development **65**(6), 1581–1604 (1994)
9. Neha Gupta, P.M.: Spell checking techniques in nlp: A survey. International Journal of Advanced Research in Computer Science and Software Engineering Research Paper **2**, 217–21 (Decemebr 2012)
10. Golding, A.R.: A bayesian hybrid method for context-sensitive spelling correction. CoRR **cmp-lg/9606001** (1996)
11. Hand, D.J., Smyth, P., Mannila, H.: Principles of Data Mining. MIT Press, Cambridge, MA, USA (2001)
12. Hastie, T., Tibshirani, R., Friedman, J., New York, NY, USA
13. Navigli, R.: Word sense disambiguation: a survey. ACM COMPUTING SURVEYS **41**(2), 1–69 (2009)
14. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 1st edn. Prentice Hall PTR, Upper Saddle River, NJ, USA (2000)
15. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Machine Learning **29**(2-3), 131–163 (1997)
16. Cunningham, P., Delany, S.J.: k-Nearest Neighbour Classifiers (2007)
17. Pedersen, T., Patwardhan, S., Michelizzi, J.: Wordnet::similarity: Measuring the relatedness of concepts. In: Demonstration Papers at HLT-NAACL 2004. HLT-NAACL–Demonstrations '04, pp. 38–41. Association for Computational Linguistics, Stroudsburg, PA, USA (2004). http://dl.acm.org/citation.cfm?id=1614025.1614037
18. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: In Proceedings of the 20th International Joint Conference on Artificial Intelligence, pp. 1606–1611 (2007)
19. Pajkossy, K.: Studying feature selection methods applied to classification tasks in natural language processing. MSc thesis, Eötvös Loránd University (2013)
20. Charniak, E.: Statistical parsing with a context-free grammar and word statistics. In: Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference, AAAI 97, IAAI 97, July 27-31, 1997, Providence, Rhode Island., pp. 598–603 (1997)
21. Reiter, E., Sripada, S., Hunter, J., Davy, I.: Choosing words in computer-generated weather forecasts. Artificial Intelligence **167**, 137–169 (2005)