Tutorial on Natural Language Processing

Saad Ahmad
Artificial Intelligence (810:161) Fall 2007
University of Northern Iowa
Ahmads09@uni.edu

Abstract

Natural languages are languages spoken by humans. Currently we are not yet at the point where these languages in all of their unprocessed forms can be understood by computers. Natural language processing is the collection of techniques employed to try and accomplish that goal.

The field of natural language processing (NLP) is deep and diverse, This paper will introduce natural language understanding and generation to the reader then go in depth on how these topics work and relate to NLP as a whole. Furthermore, this paper will discuss the applications and challenges of NLP, namely duplicate error report detection, tutoring systems, and database interfaces.

1. Introduction

Natural language processing (NLP) is a collection of techniques used to extract grammatical structure and meaning from input in order to perform a useful task as a result, natural language generation builds output based on the rules of the target language and the task at hand. NLP is useful in the tutoring systems, duplicate detection, computer supported instruction and database interface fields as it provides a pathway for increased interactivity and productivity.

2. Natural Language Understanding

> There are many advantages of natural language as a communication channel between a man and a machine. One of them is that the man already knows the natural language, so that he does not have to learn an artificial language nor bear the burden of remembering its conventions over periods of disuse… There arise occasions where he(the user) knows what he wants the machine to do and can express it in natural language, but does not know exactly how to express it to the machine. A facility for machine understanding of natural language could greatly facilitate the efficiency of expression in such situations -- both in speed and convenience, and in decreased likelihood of error.(Woods p.18)

The chapter on NLP in *Artificial Intelligence Illuminated* discusses how a system would be able to parse and interpret input. Natural language (understanding) consists of four main sections: Morphology, Syntax, Semantics and Pragmatics.

Morphology is the first stage of analysis once input has been received. It looks at the ways in which words break down into their components and how that affects their grammatical status.

Syntax involves applying the rules of the target language's grammar, its task is to determine the role of each word in a sentence and organize this data into a structure that is more easily manipulated for further analysis.

Semantics are the examination of the meaning of words and sentences. Semantics convey useful information relevant to the scenario as a whole.

Pragmatics are the sequence of steps taken that expose the overall purpose of the statement being analyzed. This will be broken down into Ambiguity and Disambiguation to facilitate understanding.

2.1 Morphology

Morphology is mainly useful for identifying the parts of speech in a sentence and words that interact together. The following quote from Forsberg gives a little background on the field of morphology:

> A morphology is a systematic description of words in a natural language. It describes a set of relations between words' *surface forms* and *lexical forms*. A word's surface form is its graphical or spoken form, and the lexical form is an analysis of the word into its *lemma* (also known as its *dictionary form*) and its *grammatical description*. This task is more precisely called *inflectional* morphology. Yet another task, *derivational* morphology, describes how to construct new words in a language (Forsberg p.213)

Being able to identify the part of speech is essential to identifying the grammatical context a word belongs to. In English, regular verbs have a ground form with a limited set of modifications, however, irregular verbs do not follow these modification rules, and greatly increase the complexity of a language. The information gathered at the morphological stage prepares the data for the syntactical stage which looks more directly at the target language's grammatical structure.

2.2 Syntax

Syntactical analysis is the application of the languages grammar to the application's input. This information is used to develop a parse tree for the sentence which will later be used to uncover the meaning of the sentence in the semantic analysis.

2.2.1 Grammar

In English, a statement consists of a noun phrase, a verb phrase, and in some cases, a prepositional phrase. A noun phrase represents a subject that can be summarized or identified by a noun. This phrase may have articles and adjectives and/or an embedded verb phrase as well as the noun itself. A verb phrase represents an action and may include an imbedded noun phrase along with the verb. A prepositional phrase describes a noun or verb in the sentence. The majority of natural languages are made up of a number of parts of speech mainly: verbs, nouns, adjectives, adverbs, conjunctions, pronouns and articles.

2.2.2 Parsing

Parsing is the process of converting a sentence into a tree that represents the sentence's syntactic structure. The statement: "The green book is sitting on the desk" consists of the noun phrase: "The green book" and the verb phrase: "is sitting on the desk." The sentence tree would start at the sentence level and break it down into the noun and verb phrase. It would then label the articles, the adjectives and the nouns. Parsing determines whether a sentence is valid in relation to the language's grammar rules.

2.3 Semantics

After determining the structure of a sentence the next step is to determine the meaning of the sentence. This process builds up a representation of the objects and actions that a sentence is describing and includes the details provided by adjectives, adverbs and propositions. This process gathers information vital to the pragmatic analysis in order to determine which meaning was intended by the user.

2.4 Pragmatics

Pragmatics is "the analysis of the real meaning of an utterance in a human language, by disambiguating and contextualizing the utterance"(Coppin). This is accomplished by identifying ambiguities encountered by the system and resolving them using one or more types of disambiguation techniques.

2.4.1 Ambiguity

Ambiguity is explained as "the problem that an utterance in a human language can have more than one possible meaning. Types of ambiguity include lexical, semantic, syntactic, referential and local"(Coppin).

Lexical Ambiguity results when a word has more than one possible meaning such as in the case of "board", it could mean the verb "to get on" or it could refer to a flat slab of wood.

Syntactic Ambiguity is present when more than one parse of a sentence exists. "He lifted the branch with the red leaf." The verb phrase may contain "with the red leaf" as part of the imbedded noun phrase describing the branch or "with the red leaf" may be interpreted as a prepositional phrase describing the action instead of the branch, implying that he used the red leaf to lift the branch.

Semantic Ambiguity is existent when more than one possible meaning exists for a sentence as in "He lifted the branch with the red leaf." It may mean that the person in question used a red leaf to lift the branch or that he lifted a branch that had a red leaf on it.

Referential Ambiguity is the result of referring to something without explicitly naming it by using words like "it", 'he" and "they." These words require the target to be looked up and may be impossible to resolve such as in the sentence: "The interface sent the peripheral device data which caused it to break", it could mean the peripheral device, the data, or the interface.

Local Ambiguity occurs when a part of a sentence is unclear but is resolved when the sentence as a whole is examined. The sentence: "this hall is colder than the room," exemplifies local ambiguity as the phrase: "is colder than" is indefinite until "the room" is defined.

2.4.2 Disambiguation

There are many techniques and tools to decide which interpretation of a word to use, some of these techniques are listed below:

Prior probabilities are rules that tell the system that a certain word phrase nearly always means a certain thing without looking at anything else, this is a purely statistical approach to disambiguation.

Conditional probability examines the scenario in reference to the origin of the phrase in order to make the decision on the meaning of a word phrase.

Context looks at the environment and incidents surrounding the phrase in order to make a decision on which interpretation to use.

World Models are needed for a good disambiguation system, to allow for the selection of the most practical meaning of a given sentence. This world model needs to be as broad as the scenarios the system would encounter in its normal operation.

3. Natural Language Generation

In the paper by McRoy, natural language greatly enhances the interactivity and effectiveness of dialogue systems. McRoy's team introduces a three step process to generating the natural language output portion of an

interactive dialogue program. This process consists of the text planning phase, the sentence planning phase and finally, the text realization phase.

3.1 Text Planning

Text planning consists of two general steps: Content planning and Discourse structuring. Content planning is concerned with deciding what needs to be addressed and retrieving this data from a knowledge base. Discourse structuring involves organizing content in a way that meets the communication goal of the content planning step and ensure coherence and understandability.

3.2 Sentence Planning

The goal of sentence planning is to increase the fluency of the output using a three section process: Lexicalization, Aggregation, and Referring expression generation. Lexicalization is the process that chooses the words and phrases required to convey the content, Aggregation decides how to arrange phrases in sentence size chunks, and Referring expression generation selects a pronoun or phrase to set the tone for the entire sentence.

3.3 Text Realization

Text realization is the mapping of a sentence plan into a sentence structure, this process may be either simple or complex depending on how fluent the desired output should be and how quickly the system should run.

5. Applications and Challenges to NLP

Some of the proven abilities of natural language interfaces discussed in Hendrix's paper include the ability to answer direct questions, handle simple pronoun use, correct spelling errors, analyze null answers, and coordinate multiple files. Following are a few examples of NLP applied to duplicate error report detection, tutoring systems and database interfaces.

5.1 Duplicate Error Report Detection

In fields where product defect reports are common, a duplicate error report detector would save time spent identifying these duplicate reports. The prototype discussed by Alexandersson identified 40% of the total number of duplicate error reports submitted to the users of the prototype. This prototype used tokenization, stemming, stop word removal, vector space calculation and similarity calculation to identify duplicates.

Tokenization is typically accomplished by removing punctuation, capitalization and other modifications to a stream of characters in order to break them into tokens that can be analyzed further by the system.

Stemming is the process of removing lexical components (the alterations to a ground word) to reach the ground word itself. The tokens from the input stream are processed and simplified to their ground forms and passed on for further analysis.

Stop word Removal, stop words are the words such as articles that are commonly found in many phrases. These words are removed to improve the searching time efficiency.

Vector Space Representation, each remaining word becomes a vector in a multidimensional space with the search parameters as the axes. The search result is represented as a vector in relation to these axes, the vector's length along a particular axis is determined by how many words it has that are the same as the axis label. This system is used in the similarity calculation. More unique words are weighted more heavily than common words in order to produce good results.

Similarity Calculation, calculating the similarity of one result to the query involves finding the total length of the result vector from the axes and comparing this result with other results to form a relevance ranking, in which the longer the vectors are, the more similar they are to the query. This final result is then analyzed to identify duplicates in the set of error reports.

5.2 Tutoring Systems

The application of NLP to tutoring applications is a relatively complex function due to the need for a dialogue system instead of the more traditional monologue systems that have been developed. The need for dialogue is directly related to the fact that tutoring is highly interactive, and the system may need to adjust its plans to produce more effective instruction based on input from the user:

> Human tutoring is a collaborative process, in which tutor and student work together to repair errors. It is highly interactive, with the tutor providing constant feedback to support students' problem solving.(Di Eugenio p.29)

In addition to tackling the dialogue, research is being conducted on the best way to have a student learn. In the BEESIM project discussed in Di Eugenio's paper, the aim was to "operationalize" the idea that students learn best when they construct knowledge themselves. In the study, human tutors and their students were evaluated on improving test scores under two different learning conditions: Socratic and Didactic. The Socratic condition's aim was to prompt the learner with as little information as possible, the Didactic condition had the tutor explain to the student what they felt the student needed in order to proceed and then query the student to test for understanding. In short, the students in the Socratic environment showed greater improvement than the Didactic environment's students. A system utilizing the Socratic environment would need to be built in such a way that the system would be able to monitor the student and prompt the student as needed.

5.3 Database Interface

Hendrix's tutorial covers natural language database interfaces as being a practical application of the goals of NLP:

> One of the most important areas for the practical application of NLP is in accessing databases. Because conventional databases are among the few types of symbolic knowledge representations that are indexed in a computationally efficient manner, are in widespread use, and have a semantics that is well understood, providing access to them is currently the best understood commercially viable task for natural-language processing.(Hendrix p.5)

Current development has produced systems that can provide usable NL interfaces for single databases, produce simple reports, and coordinate multiple files. The largest problem in this area is "telling systems about the vocabulary, concepts, and linguistic constructions associated with new databases"(Hendrix) This problem may be solved by using the database itself to guide the linguistic processes.

5.4 Challenges

In systems that deal more directly with human users, the system must be able to handle the curve-balls thrown to them:

> Spelling correction is an important aspect of the input understanding, as students frequently misspell words, abbreviate creatively, and make word boundary errors (two words joined together or a single word split in two). Spelling correction is based on a three-way match algorithm which slides a small window simultaneously across both the unknown input word and a candidate word from the lexicon. Transpositions, elisions, substitutions, and similar errors are counted and the most likely candidate is picked.(Evens p.14)

Leidner's paper describes several major challenges to NLP from the viewpoint of software engineering. Of the challenges mentioned, accuracy, efficiency, and scalability appear to be the most vital to developing fully functional systems.

Accuracy: Natural language techniques can never guarantee a complete and correct result, and as a result, the entire system must be able to take this into account and provide the appropriate fallbacks.

Efficiency: Research shows that response times greater than four seconds renders a system too slow to be acceptable, and many NLP systems fall into this category. The natural language systems that have been developed for research have not placed an emphasis on efficiency, leaving it and other issues related with software design as "implementation detail." To date it is still unclear how efficient a system can be.

Scalability: Any system that is deployed would likely need to be able to deal with a large number of users or documents. Runtime, complexity and memory results of research projects in the field are usually not reported.

6. Conclusion

Despite the challenges against them, Natural Language systems hold great promise in areas involving human-computer interaction. Great progress has been made towards developing practical applications with NLP systems in certain areas, other areas are still in need of work and provide an open area for research for developing theories.

> It will be a long time, if ever, before we can create programs that understand and produce language as people do. But It is time for new creative possibilities. The next few years will see many new and valuable applications whose diversity and novelty will remind us once again how hard it is to predict the future from our limited interpretations of the present.(Mohammed p.30)

References:

Coppin, B. (2004). *Artificial Intelligence Illuminated*. Sudbury, Massachusetts: Jones and Bartlett Publishers.

Di Eugenio, B. (2001).Natural-Language Processing for Computer-Supported Instruction. *Intelligence*. *Winter 2001*, 22-32.

Evens, M.W., Zhang, Y., Michael, J.A., & Rovick, A.A. (1997). CIRCSIM-Tutor: An Intelligent Tutoring System Using Natural Language Dialogue. *Morgan Kaufmann Publishers*, 13-14.

Forsberg, AuthorM., & Ranta, A. (2004). Functional Morphology. *Association for Computing Machinery Press. 39*,

Hendrix, G.G., & Carbonell, J.G. (1981). A Tutorial on Natural Language Processing. *ACM. '81*, 4-8.

Leidner, J.L. (2003).Current issues in Software Engineering for Natural Language Processing. *Association for Computational Linguistics*. 8, 45-50.

McRoy, S.W., Channarukul, S., & Ali, S.S. (2001). Creating. *CreatingNatural Language Output for Real-Time Applications*. *Summer*, 21-34.

Mohammed, F.A., Nasser, K., & Harb, H.A.
A Knowledge Based Question Answering System. *Sigart Bulletin*. *4*, 21-33.

Runeson, P., Alexandersson, M., & Nyholm, O. (2007). Detection of Duplicate Defect Reports Using Natural Language Processing. *29th International Conference on Software Engineering (ICSE'07)*.

Woods, W.A. (1977).A Personal View of Natural Language Understanding. *SIGART Newsletter*. *61*, 17-20.