COMPUTATIONAL LINGUISTIC MORPHOLOGY

Daniel Harrison
Carlos Lazo

Artificial Intelligence
Fall 2006

Introduction to NLP and Linguistic Morphology

Natural Language Processing (NLP) is a hybrid discipline that takes from both computer science and linguistics. It consists of two problems: converting computer data into text and converting text into a format that is more easily understood by a computer. The latter receives, by far, the most attention.

While computers have given way to massive productivity and organization increases in recent years, the way one interfaces with them remains clunky and difficult to learn. This is especially true for those that would like to use their computer like they would a blender: with little prior instruction, no manuals, and a quick learning curve. The desktop metaphor and the command line that came before it allow experts to interact effectively and quickly, but have a steep learning curve that has prevented many would-be casual users from making the most of the capabilities of modern computers. And even the experienced users sometimes have difficulty accomplishing what they'd like to. From talking to a gas station attendant to ordering dinner at a restaurant to giving a sales presentation to interacting with friends and family, spoken and written language plays an extremely large role in our daily lives. If computers were able to understand and act on natural language inputs, many of these problems would be solved. Some examples of this are commands such as "Please find all movies on my computer that are longer than 2 hours and back them up to the 200 gigabyte external hard drive." and "Find and print my Fall 2006 class schedule." Both of these can be done using existing interfaces, but both would be easier if the computer could just be told what to do. In the former, a search must be performed for files that have a certain file extension or that have metadata designating them as video files. Then, the they have to be filtered so that only ones above the specified length remain. Finally, the drive corresponding to the external hard drive with the named size must be found, and the files copied to it. In the latter, a document that is titled similarly to "Fall 2006 Class Schedule" must be found. Then the application that is associated with printing it must be opened and told to open the file. Finally, the document is sent to (probably the default) printer. Not only is the existing way to accomplish these tasks slower, it may be too involved for some users to grasp. If this is the case, the spoken commands in a way add new functionality.

Linguistic morphology is the study of the structure of words. Text can generally be broken up into multiple categories: letters, words, sentences, paragraphs, and so on. Of these, it is commonly accepted that words are the most useful structure to study. This is for two reasons. First, they are the most basic unit with meaning. Second, words have relationships. For example: one can tell that doughnut and doughnuts are obviously related and, additionally, one can tell that they are related in the same way that book and books are. Morphology studies these relations as well as others within a language and attempts to derive rules for their formation.

NLP is a broad topic encompassing everything from parsing text into words to deriving meaning from those words to determining an overall meaning from the meanings of the individual words and the sentence structure. Morphology is a very important step in this process. Once the individual words have been parsed out, the meanings must be assigned. This is not a trivial task for reasons that include the meaning of words changing when they are placed in different contexts and that will be discussed later. Only after this is complete can the overall meaning of the input text be determined and acted upon. The key role it plays in NLP as well as the computational appeal are two of the many reasons we've decided to explore computational linguistic morphology.

A Brief History of NLP

The field of Natural Language Processing (NLP) has been of tremendous interest to researchers for quite some time. The roots of NLP can be traced back to the post World War II era that lead to the initial prototype of the computer. The rise of the automaton concept along with probabilistic and information-theoretic models by Turing in the 1940’s and 1950’s led to the development of formal language theory. Algebra and set theory were used to formally define languages as sequences of symbols, leading to the creation of context-free grammar. Implementations of early machine translation systems were relatively inadequate. Computer scientists and mathematicians found very quickly that the idea of language was far more complex then they had originally imagined. Creating knowledge databases and representations, regardless of human fluency in the languages, proved to be nearly impossible for all researchers involved.

In order to surpass the apparent roadblock, NLP merged with the Linguistics field starting in the late 1950’s. Researchers in linguistics began joining Machine Translation teams, ultimately helping scientists understand the intricacies of language and how they could potentially be modeled in a computational sense. Linguistic theories were revolutionized in 1957 by Noah Chomsky, a young American Linguist who introduced the idea of Generative Grammar: rule-based descriptions of syntactic structures. Suggestions such as pre and post-editing text were attempted in the early 1960’s in an effort to increase both accuracy and efficiency in the NLP realm. Problems were still present when analyzing a sentence such as: “She wore small shoes and socks.” In this case, one must pose the question – are the socks small as well? Such intricacies, as simple as they may seem, still needed much more exploration as fully comprehending and understanding the sense of the sentence is crucial in NLP.

Very important concepts for NLP field research were explored between the 1960’s and 1990’s. The Augmented Transition Network (ATN), a piece of software with the capability of using powerful grammars for syntax processing, was fully realized. This aided in producing parses of English sentences which could effectively be used in future syntactic analysis. Semantically speaking, case grammar was developed to express the relationship between nouns and verbs when used with linking propositions. Simultaneously, different semantic representations of languages were analyzed, such as semantic networks, language processing systems, and database systems.

The modern understanding of NLP can be divided into several sub problems: Morphological analysis, Syntactic analysis, Semantic analysis, Discourse integration, and Pragmatic analysis. Morphological analysis delves into the realm of word components, looking into elements such as prefixes, suffixes, and indicators of plurality. Meanwhile, relationships between words in a linear language sequence and their respective transformation into structures are explored in Syntactic analysis. Once word sequences are properly formulated, Semantic analysis is performed to assign them correct, distinct meanings. Meanings of word sequences have the potential of being dependent upon preceding sentences and may affect future linguistic structures - Discourse integration helps to ensure that the overall discourse of the sentence is logical.

Finally, Pragmatic analysis aids in the ensuring coherent reinterpretation of linguistic structures.

All exploration, conducted from the beginnings of NLP to today, has laid the foundation for current on-going research efforts. Comprehension in the field has sky-rocketed in the past few years, leading to many successes in areas such as data mining, language parsing, and speech recognition. Computational linguistic morphology, the study of word structures and their respective meanings, has and continues to play an important role in NLP.

Details of Linguistic Morphology

There are many things that make Natural Language Processing hard. First, and probably the most difficult of the challenges, extracting meaningful information from text often requires external information. For example, "The girl eats the apple with a smile" and "The girl eats the apple with a bruise" are very similar in structure. In fact, all the but last word is the same, and the last word is a noun in each of them. Only by having external information about girls, apples, smiles, and bruises can one infer that the girl has the smile in the first sentence and the apple has the bruise in the second sentence. Further, in a sentence such as "The girl eats hot dogs with relish," the word relish could either mean the condiment or enthusiasm. The meaning would have to be inferred from other surrounding sentences, or, in some cases, may not ever be able to be fully inferred. Second, a word can take on multiple meanings even when it is using the same definition. For example, the word tall in the phrase "a tall giraffe" can mean both a giraffe that's tall absolutely (as giraffe's are) or a giraffe that's tall compared to other giraffes. However, the word tall in the phrase "a tall poodle" almost definitely means a poodle that's tall compared to other poodles. Additionally, the object of a pronoun can be ambiguous. In the text "John took his car to the store. It was in bad shape" - it could refer to either the car or the store. The way that words and sentences are shortened also has to be considered. In the questions "Who is the teacher for CS I next semester? For Data Mining?" - the second sentence is a shortened form of "Who is the teacher for Data Mining next semester?" Finally, there is also social knowledge that comes into play. Phrases such as "Pass me the salt," "Pass me the salt, please," "Can you pass me the salt?" and "Could you pass me the salt?" all imply different levels of politeness. There are many other reasons that NLP is difficult, but these are a few of the most prominent.

Morphological analysis has truly come a long way in the overall history of the subject. This specific topic dates far back to ancient Indian times (520-460 BC), where linguist Pānini was able to develop 3,959 rules of the Sanskrit morphology. It is, to this day, the earliest known grammar of the Sanskrit language, and marks the earliest known work on linguists as a whole. The term 'morphology' itself was instated by Frenchman August Schleicher in 1859, being defined as the patterns of word formation in a particular language, including inflection, derivation, and composition.

Early approaches to morphology in the modern world commenced in the 1960's with work conducted by Noam Chomsky and Morris Halle. They were able to develop traditional phonological grammars which consisted of rewrite rules which proved to be more powerful than context-free grammatical rules developed in earlier years. Stemming from the work of Chomsky and Hale, Xerox began the important development of the Two-Level Morphology model which would be used in all future linguistic grammatical models. The model is constraint based that does not depend on a rule compiler, a composition, or any other finite-state algorithm.

Through the linear, upward progression of morphology, many fundamental and important concepts were developed that can be seen throughout linguistic analysis. The idea of lexemes and word-forms plays in integral part in understanding lexical structure. A simple example of a lexeme is 'car' vs. 'cars', where these two words are in essence the same 'word' but differ simply in number. There is a clear and special distinction between the lexeme and a word-form as can be seen in the following example: 'the car is big' vs. 'the cars are big.' In this case – the phrases are word forms since the verb 'is' must be altered to 'are' to account for the plurality change of 'car' to 'cars'.

The determination of a morphological rule being either inflectional or word-based is important when understanding linguistic meaning and syntax. A rule is categorized as inflectional when it relates different forms of the same lexemes ('car' vs. 'cars'). The example 'carwash' would characterize a word-formation, where the two lexemes 'car' and 'wash' are combined. From a high-level point of view, inflectional rules will produce different variations of the same word where word-formation rules will produce a completely new lexeme.

A paradigm, a complete list of related word-forms associated with a dictated lexeme, is absolutely crucial to morphology. Applications can easily be seen in linguistic verb conjugations for different languages. These paradigms can be arranged into their own tables utilizing different inflectional rules for organizational purposes – for example:

number, gender, tense, etc. These inflectional policies must comply and be relevant to the syntactic rules of the specific language. Morphosyntax is the part of morphology that explores the relationship between the syntax and the morphology of the given language and grammatical rules. Morphosyntax primarily concentrates on the bond between inflectional status and syntax, but not with word-formation.

There are certain distinctions in morphology that must be made when analyzing a language. In pluralization, there are certain cases where the lexeme can vary from the singular to plural form in sound and still contain the same meaning. Pluralizing ‘car’ yields the word ‘cars,’ where simple grammatical rules dictate the addition of an ‘s’ to the end. In the case of pluralizing ‘wish,’ an extra vowel (‘e’) must be added in conjunction with the ‘s,’ yielding the word ‘wishes’ with a different phonetic interpretation. Phonological lexemic changes make up what is known as allomorphy in linguistic morphology.

Linguistic morphology consists of many different intricacies that must be considered when correctly analyzing the syntax of any language. Each of the fundamental concepts must be applied specifically to each set of inflectional rules to ensure valid language translations. Although the concrete ideas listed above do not solely compose the realm of linguistic morphology, they are quintessential to fully comprehending the grammatical nature and syntactic rules of a language.

Linguistic morphology is comprised of three primary models which attempt to capture the different morphological fundamental concepts in their own distinct ways. Morpheme-based morphology analyzes word-forms known as morphemes. A morpheme is defined as any of the minimal grammatical units of a language, each constituting a word or meaningful part of a word. For example, take the word ‘insufficiently.’ This word can be split up into the following morphemes: ‘-in’, ‘-sufficient’, and ‘-ly.’ Item-and-Arrangement is a method of interpreting word-forms by stringing morphemes together. One can run into problematic situations when using Item-and-Arrangement when analyzing a word such as ‘octopi,’ the plural for of octopus. In this case, where there is no allomorphy present, the morphological interpreter will utilize null morphemes (a morpheme that has no phonological expression) in conjunction with the irregular plural stem. Preserving morpheme-based syntactic rules is imperative in ensuring consistent analysis across an entire language.

Lexeme-based morphology varies from morpheme-based morphology in that is does not utilize morphemes to produce resulting word-forms. Instead, it simply applies a set of distinct inflectional rules that will merely change the current word-form and produce the correct output. Inherently, this eliminates the necessity of the null morphemes as seen in a morpheme-based environment.

Word-based morphology differs vastly from the morpheme and lexeme based approaches. It utilizes linguistic methodology to create generalized relationships between different forms of inflectional paradigms. In essence, this idea would take a morpheme and attempt to categorize it in terms of a specific type of form, such as “first person past,” and would therefore derive the implied meaning.

All three morphological methods of analysis are effective in their respective domains. Differentiation in usage is imperative when trying to syntactically analyze and parse a language in an attempt to extrapolate meaning. The cohesion between the fundamental ideas of computational linguistic morphology and the methods of analysis will help determine the overall accuracy and strength of an agent or program that looks to implement intelligence in grammatical and language interpretation.

Algorithms for Linguistic Morphology

One of the largest contributers to modern computational linguistics and morphology was Noam Chomsky. In the 1950s, he studied the application of finite state languages to natural language syntax. He demonstrated that, while finite state languages are much simpler, more easily understood, and effectively modeled, they are not powerful enough to adequately model natural languages. As a result, he decided to use the more powerful, but less efficient, non-finite state machines, in his transformational generative grammar. Although, most researchers agree that finite state machines are useful for phonology, which is very closely related to morphology. The most successful approach to finite state phonology was developed by Kimmo Koskenniemi in the early 1980s, and is called two-level morphology. This was developed because of the complexity of the inflection system in Finnish, Koskenniemi's native language. While much less complex phonetically, English can also be modeled with two-level morphology.

The main difference between Koskenniemi's approach and those before him is parallelization. Before, researchers used a serial approach to transform between word forms. This involved one-way rules that transform a begin or intermediate state into an intermediate or final state. As a result, once a rule is applied, the earlier form is inaccessible to later rules. Also, because the rules are one-way, they essentially have to be written twice. Kosenniemi's approach is to create rules that can be applied in parallel, which allows for much more power. Because of this parallelism, there are only two states, the start and end. The power comes from this parallelism: because the word form isn't transformed, the entire start state is available for all the rules to process, instead of relying on intermediate states. Also, the lack of an imposed sequentiality allows the interactions between the rules to be much more complex. This is both an advantage and a disadvantage: the power of the rules are increased dramatically, but they are much more difficult to formulate. Finally, there is an inherent bidirectionality in the way the rules are expressed, once they are written, they can be used to convert word forms either from the underlying to surface form or from the surface to the underlying form.

One thing that the two have in common is a linear representation of the word forms, meaning that they are a list of symbols. An example of this is the phrase "have sailed the seas" which can be mapped into a predicator of HAVE + SAIL + PAST PARTICIPLE and an object of THE + SEA + PLURAL. Most modern approaches to phonology use a non-linear approach, though none have yet become as widely used as PC-KIMMO, a natural language processing tool which has a morphological parser based on Koskenniemi's method.

There are many applications for computational morphology. One of these is pluralization. The use for this mostly arises from software needing to make the verbs in its output agree with any numbers it needs to output, but it is also used in software as diverse as translation software to search engines to calendaring applications. As an example, we've decided to focus on the verb agreement problem. There are several workarounds to this: the problem could just be ignored, allowing messages such as "There were 1 errors.”; the wording could be tweaked to avoid the problem, such as “Number of errors:1”; both possibilities could be printed, such as “There were(was) 1 error(s).”; and the options could be hard coded, such as “print ‘There ‘ + (errors==1?’was ’:’were ’) + errors + (errors==1?’error.’:’errors.’)”

If a more general approach is required, algorithms can be used to pluralize words. The simplest approach would be to add -s to the end of words, but this fails on words that end in s, such as class->classes, where an -es is added. This could be taken into consideration, by adding rules that transform -s into -ses, but rules such as -y->-ies would have to be added as well. Even these have exceptions, though: -y is changed to -ys if the letter before the y is a vowel. With this in mind, we present the framework for an algorithm to pluralize English words. Three categories are applied: general rules, category exceptions, and specific exceptions. These must be applied in the order of most specific to most general. Table 1 in Appendix A is a list of general rules, and Table 2 is a list of specific exceptions. As was discussed earlier, without context there are situations where a pure search and replace algorithm such as this will not be able to always produce the correct output. This occurs in situations such as the plural of mouse: the mammal pluralizes as mice, but the computer peripheral pluralizes as mouses.

Conclusion

The effects of NLP range from smarter search engines to better spell checkers. The former can break down search terms and the content of indexed pages into morphemes to allow for searches that will consider "knife" and "knives" as the same concept. The latter could use morphological analysis to allow for more intelligent spell checking. This could be done by breaking unrecognized words into base morphemes and verifying their spellings and the way they are combined. The field of Natural Language Processing has and will continue to have a large impact on the way humans interact with computers. Not only will it allow for more intuitive and accessible interfaces, but also more efficient ones. We feel that computational linguistic morphology is a revolutionary field and have thoroughly enjoyed our exploration of this important topic.

Appendix A

TABLE 1

Singular suffix	**Anglicized plural**	**Classical plural**	**Example (see Appendix A for comprehensive lists** of words in each category)
-a	*(none)*	**-ae**	**alga -> algae**
-a	**-as**	**-ae**	**nova -> novas/novae**
-a	**-as**	**-ata**	**dogma -> dogmas/dogmata**
-an	**-en**	*(none)*	**woman -> women**
-ch	**-ches**	*(none)*	**church -> churches**
-eau	**-eaus**	**-eaux**	**chateau -> chateaus/chateaux**
-en	**-ens**	**-ina**	**foramen -> foramens/foramina**
-ex	*(none)*	**-ices**	**codex -> codices**
-ex	**-exes**	**-ices**	**index -> indexes/indices**
-f(e)	**-ves**	*(none)*	**wolf -> wolves** **life -> lives**
-ieu	**-ieus**	**-ieux**	**milieu -> mileus/milieux**
-is	*(none)*	**-es**	**basis -> bases**
-is	**-ises**	**-ides**	**iris -> irises /irides**

-ix	-ixes	-ices	matrix -> matrixes/matrices
-nx	-nxes	-nges	phalanx -> phalanxes /phalanges
-o	-oes	*(none)*	potato -> potatoes
-o	-os	*(none)*	photo -> photos
-o	*(none)*	-i	graffito -> graffiti
-o	-os	-i	tempo -> tempos/tempi
-on	*(none)*	-a	aphelion -> aphelia
-on	-ons	-a	ganglion -> ganglions/ganglia
-oo-	-ee-	*(none)*	foot -> feet tooth -> teeth
-oof	-oofs	-ooves	hoof -> hoofs/hooves
-s	-s	*(none)*	series -> series
-s	-ses	*(none)*	atlas -> altases
-sh	-shes	*(none)*	wish -> wishes
-um	*(none)*	-a	bacterium -> bacteria
-um	-ums	-a	medium -> mediums/media
-us	*(none)*	-era	genus -> genera
-us	*(none)*	-i	stimulus -> stimuli
-us	-uses	-era	opus -> opuses/opera
-us	-uses	-i	radius -> radiuses/radii
-us	-uses	-ora	corpus -> corpuses/corpora
-us	-uses	-us	status -> statuses/status
-x	-xes	*(none)*	box -> boxes
-y	-ies	*(none)*	ferry -> ferries
-zoon	*(none)*	-zoa	protozoon -> protozoa
(none)	-s	-im	cherub -> cherubs/cherubim

TABLE 2

Singular form	Anglicized plural	Classical plural

beef	**beefs**	**beeves**
brother	**brothers**	**brethren**
child	*(none)*	**children**
cow	**cows**	**kine**
ephemeris	*(none)*	**ephemerides**
genie	**genies**	**genii**
money	**moneys**	**monies**
mongoose	**mongooses**	*(none)*
mythos	*(none)*	**mythoi**
octopus	**octopuses**	**octopodes**
ox	*(none)*	**oxen**
soliloquy	**soliloquies**	*(none)*
trilby	**trilbys**	*(none)*

Links Used

http://en.wikipedia.org/wiki/History_of_linguistics

http://en.wikipedia.org/wiki/Linguistics

http://en.wikipedia.org/wiki/Morphology_(linguistics)

http://en.wikipedia.org/wiki/Natural_language_processing

http://portal.acm.org/citation.cfm?id=981131.981150

http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-155.pdf

http://www.cs.bham.ac.uk/~pjh/sem1a5/pt1/pt1_history.html

http://www.cs.colorado.edu/%7Emartin/SLP/slp-ch1.pdf

http://www.csse.monash.edu.au/~damian/papers/HTML/Plurals.html

http://www.ling.helsinki.fi/~koskenni/esslli-2001-karttunen/

http://www.sil.org/computing/catalog/show_software.asp?id=1

http://www.sil.org/pckimmo/pc-kimmo.html