MORPHEUS, A Tool for the Lexical Analysis of Corpora for Morpheme Segmentation

Chris Hall Patrick Juola Adam Boggs
University of Colorado at Boulder

1 Introduction

MORPHEUS is a tool used for segmenting morphemes, the smallest unit of a word that has meaning, for multi-lingual corpora. The ability to split a word into its individual morphemes has significance in NLP as well as cognitive modeling projects. Translation is simpler at the morpheme level, as systems can make guesses as to how new words may be formed in the target language. For example, in English, when the letters 'ly' appear at the end of the word it often signifies an adverb.

The background of MORPHEUS has its basis with a rather simple conjecture, that words consist of streams of one or more conjoined morphemes. However, predicting the boundary between two successive morphemes is a difficult problem. If the morpheme boundary can be determined, words can be split into their individual morphemes, and then the morphemes can be combined to form new words. Our approach for splitting the words uses the mathematics of Information Theory.

2 Background

2.1 Morphology

A "morpheme" can be most simply defined as the smallest unit of language that carries meaning. For example, the word *unhappiness* can be restructured and analyzed as "un- + happy + -ness", where each component carries part of the meaning of the full word. Morphological analysis provides an easy-to-use and useful tool for such problems as part-of-speech tagging, identification of novel words, and dictionary construction. Like most natural language problems, though, the problem of morpheme identification, particularly for unfamiliar languages, is difficult. Even determining what constitutes a morpheme can be difficult; Peters and Menn (1993) discuss some of the theoretical difficulties in identifying allomorphs (variant spellings of the same morpheme).

Although much work has been done (e.g. Bybee 1985) on the functional aspects of morphology, resulting in several interesting universals on the ordering and semantic roles of morphemes, little is known about the phonotactical structure of morphemes. For instance, Rumelhart and McClelland (1986) describe an attempt to infer the method of producing English past tense forms using a neural network. Borin (1991) describes a system that will infer inflectional morphology (such as tense, number, and gender morphemes) from a catalog of inflected forms, for automatic identification

of the stem and various syntactic markers. Both of these approaches rely on prior knowledge of the semantic categories to be expressed and will miss categories not explicitly shown to the system. For example, the focus on derivational morphology means that it is extremely unlikely that either system above would identify the common root in the words "visible" and "invisibility" unless the system was explicitly presented with antonymic pairs to demonstrate the "in-" prefix. Thus, neither approach is completely suited to the task of extracting morphological information from corpora without using native speakers for questioning.

This program focuses on a specific sub-problem within morphological analysis, that of *morphological segmentation*. Instead of analyzing words into a canonical representation of their morphemes, words are to be divided between letters into their morphological constituents. This is easier in some regards; for example, words like "geese" are considered to be monomorphemic as there is no easy way to unify the singular and plural forms. However, morphemes may have allomorphs, variant spellings of the same morpheme, such as "-ible" and "-able," which will not be unifiable by segmentation alone. In addition, many words in many languages may not be completely segmentable. Morphemes may overlap or there may be linking letters between the various morphemes to preserve the overall phonotactic shape of the word. For example, viewed diachronically, the word "toxicology" is actually comprised of two major morphemes, "toxic" and "logy." The "-o-" is simply a connecting vowel without meaning.[1]

The questions that arise at this point are twofold. Of what use is the representation of English (or any language) at a morphological level? And, secondly, how can this segmentation be performed? A full answer to the first would be an exhaustive list of potential NLP projects; in short, as the morphemes are by definition the informational chunks, a string of morphemes provides a more useful and accessible method of dealing with the meanings of large corpora. Juola (1994b); Juola (1994a) outlines a problem in automatic extraction of transfer functions for machine translation that would be greatly benefited by the ability to extract and translate individual morphemes rather than full lexical items. For the second, we explore an information-theoretic statistical approach to morphological segmentation.

2.2 Information Theory

Shannon (1948) developed the basic concepts of information theory. Borrowing the term "entropy" as a measure of randomness from thermodynamics, he defined the entropy of a set S of random events $1..N$ as

$$E(S) = \sum_{i=1}^{N} p(i) \log_2 p(i)$$

In intuitive terms, the entropy provides a numerical measure of the number of bits necessary to represent a sequence of random events. As the set of events becomes larger or the events become less predictable, the entropy of the sequence increases. Natural language text, with its semi-predictable sequence of characters or words, can be an example of such a sequence.

Studies of the information-theoretic structure of language(Shannon 1951; Oswald 1991) have shown that the information content of any language considered as a string of characters is vastly

[1] Of course, one could argue that synchronically, the vowel has been subsumed into a new morpheme "-ology." This argument will not work for languages with more extensive vowel harmony such as Hungarian.

higher than the content of that language considered as a string of words. In practical terms, the structure of the word system vastly constrains what can and cannot be said, and as much as 50-75% of the information carried in normal English text is redundant.

Obviously, some information is carried by the word structure of English itself—for example, the sentence fragment *John gave Mary the ...* is unlikely to be followed by any thing except an adverb, adjective, or noun. At the same time, the structural content of English is not sufficient to fully identify the rest of the sentence. Thus there is evidence of information being carried at a level between that of full words and of individual letters, i.e. at the morphological level. Further evidence for this view can be gleaned from Shannon's work; in (Shannon 1951), he describes the results of an experiment where native speakers were asked to guess the next letter of a sentence until they got it correct. In general, the only spots where subjects made incorrect guesses were at the beginnings of words and at morpheme boundaries.

There is evidence, then, that the entropy (unpredictability) of a string of letters in running linguistic text achieves some sort of a maximum at the boundaries between morphemes. By gathering co-occurrence statistics on the letters of a language, one can measure the running entropy and identify local maxima. In principle, this approach should be independent of the language or writing system chosen, but we will for the present concentrate on the extraction of English morphemes from a large corpus of English text.

3 Theory of Morphological Extraction

At a first pass, we will assume that all words in English consist of one or more conjoined morphemes and further that the morpheme string is intelligible, in the sense that no morpheme is itself a prefix of another morpheme. In this case, a better encoding (with reduced redundancy) would be to express the English string as a series of morpheme-symbols. Any string of letters within a morpheme, in turn, would express information regarding the potential morphemes which could contain those letters. As more letters are seen, the set of potential morphemes would decrease, until they would exactly constrain the set to one morpheme. At this time, of course, the word itself would not be known, but the remaining letters in the morpheme could be predicted exactly without error.[2]

In information theoretic terms, then, as more letters are shown, the informational content of those letters *in that context* decreases until the point where there is no new information. The informational content of each letter can be measured and used to identify the morpheme boundaries as occurring after those locations where the letter under consideration was more predictable than the letter immediately following, i.e. at the local minima of the entropy of the individual letter strings in context.

The notion of using probabilities to divide letter strings is not new. For example, Keene (1990) used a notion of mutual information between letter pairs to find meaningful decompositions of words for document retrieval. This approach differs fundamentally by using not strength of association, but predictability as directly measured by the entropy. Further, as most morphemes are more than one letter long, to provide context in pairs does not provide sufficient statistical

[2]Obviously, a similar argument could be made for predicting the letter immediately *before* the current letter set as a way of identifying morphemes from the ends of the words.

coverage. Our system (MORPHEUS) described below uses a variety of context methods up to and including full-word contexts, and could easily be extended to full sentences.

Clearly, the claim that words consist solely of conjoined morphemes without overlap or linking letters is unsupported by the linguistic data. The next immediate step is to use statistical methods to compile *lists* of morphemes, with a confidence level expressed by their regularity in the input. The morphemes with the highest confidence level can then be used as a baseline, to segment other morphemes and determine such things as vowel harmony and possible overlap between/among morphemes. In this way, a multi-pass system can be built that divides words with high confidence into a set of morphemes without the direct intervention of humans. Our prototype attempt to construct the first phase of such a system is described below, along with some experimental results and their analysis.

4 MORPHEUS : The System

MORPHEUS is a fairly small system which consists of two distinct parts. One part of the system is responsible for parsing words from a user-defined text and building an internal database from the information available in the text. The second part of the system accepts a word as input which it then attempts to split into its individual morphemes.

The first part of the system is the smallest and simplest part of MORPHEUS. Individual words are parsed from the user-defined text and added to the internal database. The internal database consists of a simple N-gram model. Thus, given N-1 characters of information, our model returns the distributions for the next letter. Source words are divided into their N-character substrings (e.g. 'theater' would be divided into 'thea,' 'heat,' 'eate,' and 'ater' in a 4-gram model) and these substrings (and their prefixes) are added to the database. The reason for splitting the word in such a manner is that many morphemes do not occur as prefixes, but rather in the middle or at the end of a word. Thus we need to add the previously described substrings so that we are able to build statistics about morphemes which appear inside or at the ends of words.

The second part of the system has a job which is very different from that of the first part. MORPHEUS attempts to split the word into its component morphemes by trying to determine the morpheme boundaries. We theorize that these boundaries will occur at two possible locations: when the entropy of a substring increases dramatically, which we will refer to as increased unpredictability, or when the next letter in a word is unexpected, which we will refer to as high surprise. Each of these methods will be elaborated on below. Thus, to split a word MORPHEUS pulls successive letters from the beginning of the word we wish to split while appending those letters to a partial morpheme. As each letter is pulled from the word, the entropy of the partial morpheme is compared with the entropy of the partial morpheme with the new letter appended.

We continue this process until the new entropy is calculated to be much larger than the previous entropy, or when the next letter that we pull off of the word has a probability of occuring that is much smaller than some threshold value. MORPHEUS then asserts that we have found a morpheme boundary, and begins again with the next letter as a single-letter substring. Thus, if we have the word 'talked' and we determine that the entropy of 'talk' is smaller than the entropy of 'talke', we then store 'talk' as a morpheme, and continue trying to split the rest of the word ('d' in this case) with 'e' as our running string.

4.1 High Surprise

As discussed above, MORPHEUS splits words when the next letter in a word is relatively unexpected compared to some threshold value. As MORPHEUS builds a partial morpheme from successive letters of a word it computes the conditional probability that the Nth letter will occur after the partial morpheme formed by the previous letters. If the probability of occuring falls below some threshold value then MORPHEUS splits the word before the Nth letter and forms a new partial morpheme. Figure 1 contains an example of splitting a word with high surprise. In this case the probability of the *m* occuring after chess was almost 0 so the system split the word between the *s* and *m* forming the morphemes *chess* and *men* .

4.2 Increased Unpredictability

In addition to split on points of high surprise, MORPHEUS splits a word when there is a sharply increased unprectability between successive partial morphemes of a word. Thus, as it builds a partial morpheme from successive letters of a word, MORPHEUS checks the entropy of the partial morpheme before adding a letter to form a new partial morpheme. If the entropy increases sharply between two consecutive entropy measurements then it splits the word immediately after the second of the two partial morphemes. Figure 2 contains a sample segmentation of the word *psychology*. It is important to note that even though the entropy increased in going from *psyc* to *psych*, it did not increase sharply enough. i.e. the slope of the line was too small. However, the entropy increase between *psych* and *psycho* was much sharper so the word was then split between the first *o* and the *l* of psychology.

4.3 The Whole System

The two parts of our system come together when we calculate the entropy of a string. Implicit in the N-gram is the conditional probability that the Nth character occurred given the previous N-1 characters grouped together as a string. We then substitute as the Nth letter every legal character in our data set and use the calculated probabilities to calculate the entropy.

There are clearly many simple modifications that can be made to this system. For instance, the system described above will break morphemes at a *local* entropy minimum. This could easily result (and does, in some cases) in a number of false-positive morpheme boundaries being found. An obvious way to prevent this would be to break each word at the global entropy minimum, or to require a morpheme boundary to be a local minimum by mor e than a certain threshold (either in terms of a percentage or an absolute difference). Another obvious change is to express the entropy of a string in terms of the letters *succeeding* it, predicting the letters in the word from back-to-front, as it were. The results of our various experiments are represented below.

5 Experimental Results and Analysis

In principle, this approach should be language-independent so we have tested it with several different languages. Using various corpora we have tested it for several languages, including French, English, Finnish, and Spanish (among others not presented here). Each in case the entire

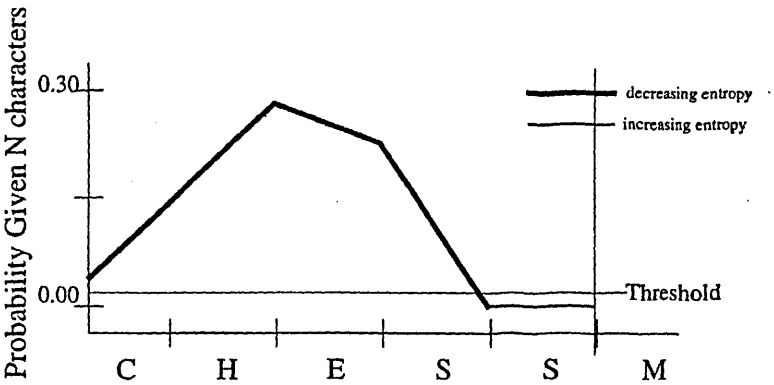

Figure 1 - Example of High Surprise

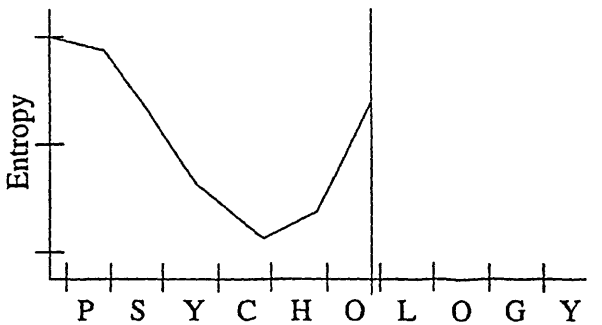

Figure 2 - Example of Sharply Increased Unpredictability

beaut-iful	beaut-y	child-'s
en-joy-ed	forgo-t	madl-y
no-how	run-ning	scornfu-lly
servan-t	slee-ves	some-times
thin-ner	travel	warm

Table 1: MORPHEUS : English sample results

Language	Wrong	Partial	Correct	Accuracy
English	40	8	31	44%
Spanish	64	4	17	22%
Finnish	371		149	28%
French	64	6	12	18%

Table 2: MORPHEUS : Performance study

corpus was pre-processed to eliminate case distinctions and most punctuation, and then each word was presented individually to MORPHEUS to develop the statistical database.

For the English testing, we asked the system to perform morphological extraction of 79 words selected randomly (with uniform distribution) from the list of words in *Through the Looking Glass*(Carroll 1988). Some of these words were common ('if'), some less so ('ebew', which appears twice), some morphologically complex ('bewildered', 'gradually'), and some were single morphemes. MORPHEUS's output was compared against the judgements of a native speaker of English and each word was classified as either correct, partially correct, or wrong.[3] Of the 79 words tested, xx were completely correct, yy were partially correct, and zz were wrong, an accuracy rate of pp%. This is an extremely significant result when one considers the paucity of the input data. Sample outputs from this system are presented here as Table 1.

The test results for the other languages (Spanish, French, Norwegian, and Finnish) are given in Table 2.

Table 2.

6 Extensions and Conclusions

Obviously, there are many additional experiments that could be done to more closely measure the accuracy of the system. The N-gram model is clearly an oversimplification and more powerful information theoretic techniques may be better able to extract morphemes. A method for combining several sorts of analysis to pick up the various morphemes found (such as "-ly" and "-ed", found by the reverse and forward analysis performed) is currently under development. And, of course, a multi-pass morphological extractor with confidence measures as described above would be useful to build and test.

[3]Correct and wrong should be self-explanatory. "Partially correct", in this case, means either that some morpheme boundaries were correct, others not, such as in 'che-erful-ly', or plausible but wrong, as in 'noth-ing', the gerund of the pseudoverb *'to noth.' For some languages, our native speakers did not use the "partially correct" category and results have been reported as they produced them.

Another interesting extension would be to take morphology more directly into account. Very little is known about the phonotactic shape of morphemes across languages, but the few results that exist can easily be incorporated. For example, morphemes tend to be added either to the beginnings or ends of stems, not in the middle. Furthermore, morphemes which are closer to their stems tend to be more closely bound and smaller than their more distant cousins. Bybee (1985) lists several results about the ordering of various sorts of tense and aspect morphemes on verbs; these, and similar results, could easily be provided to a more sophisticated system in the hopes of improving performance.

Considered as a proof-of-concept, the preliminary results from MORPHEUS are promising. Analyzing language at a morphological level is desirable for building of larger systems, and morphemes can be extracted with better than chance probability using very simple information theoretic techniques. Further improvements can only help these results, letting automatic morpheme extraction become a valuable and vital part of larger NLP projects.

References

BORIN, LARS. 1991. The automatic induction of morphological representation. Reports from Uppsala University, Linguistics (RUUL) 22, Department of Linguistics, Uppsala University, Uppsala, Sweden.

BYBEE, JOAN L. 1985. *Morphology*. Amsterdam: John Benjamin.

CARROLL, LEWIS. 1988. *Through the Looking Glass*. Champaign, Ill.: Project Gutenberg and Duncan Research, Millenium Falcon edition.

JUOLA, PATRICK. 1994a. A psycholinguistic approach to corpus-based machine translation. In *3rd International Conference on the Cognitive Science of Natural Language Processing (CSNLP-94)*, Dublin, Ireland.

——. 1994b. Self-organizing machine translation : Example-driven induction of transfer functions. Technical Report TR-722-94, Computer Science Department, University of Colorado. Also available as cmp-lg/9406012.

KEENE, CAROL A., 1990. *Document Retrieval Using Statistical Word Decomposition*. Department of Computer Science, University of Colorado dissertation.

OSWALD, JACQUES. 1991. *Diacritical Analysis of Systems : A Treatise on Information Theory*. Chichester, England: Ellis Horword, Ltd.

PETERS, ANN M., and LISE MENN. 1993. False starts and filler syllables : Ways to learn grammatical morphemes. *Language* 69.742–777.

RUMELHART, DAVID E., and JAMES L. MCCLELLAND. 1986. On learning the past tenses of English verbs. In *Parallel Distributed Processing : Explorations in the Microstructure of Cognition*, ed. by James L. McClelland, David E. Rumelhart, and the PDP Research Group, volume 2, chapter 18, 216–271. Cambridge, Mass.: MIT Press.

SHANNON, C. E. 1948. A mathematical theory of coding. In *Key Papers in the Development of Information Theory*, ed. by David Slepian, 5–29. New York: IEEE Press.

—— 1951. Prediction and entropy of printed English. In *Key Papers in the Development of Information Theory*, ed. by David Slepian, 42–46. New York: IEEE Press.