ParaMor: Minimally Supervised Induction of Paradigm Structure and Morphological Analysis

Christian Monson, Jaime Carbonell, Alon Lavie, Lori Levin
Language Technologies Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA, USA 15213
{cmonson, alavie+, jgc+, lsl+}@cs.cmu.edu

Abstract

Paradigms provide an inherent organizational structure to natural language morphology. ParaMor, our minimally supervised morphology induction algorithm, retrusses the word forms of raw text corpora back onto their paradigmatic skeletons; performing on par with state-of-the-art minimally supervised morphology induction algorithms at morphological analysis of English and German. ParaMor consists of two phases. Our algorithm first constructs sets of affixes closely mimicking the paradigms of a language. And with these structures in hand, ParaMor then annotates word forms with morpheme boundaries. To set ParaMor's few free parameters we analyze a training corpus of Spanish. Without adjusting parameters, we induce the morphological structure of English and German. Adopting the evaluation methodology of Morpho Challenge 2007 (Kurimo et al., 2007), we compare ParaMor's morphological analyses with Morfessor (Creutz, 2006), a modern minimally supervised morphology induction system. ParaMor consistently achieves competitive F_1 measures.

1 Introduction

Words in natural language (NL) have internal structure. Morphological processes derive new lexemes from old ones or inflect the surface form of lexemes to mark morphosyntactic features such as tense, number, person, etc. This paper address minimally supervised induction of productive natural language morphology from text. Minimally supervised induction of morphology interests us both for practical and theoretical reasons. In linguistic theory, the morpheme is often defined as the smallest unit of language which conveys meaning. And yet, without annotating for meaning, recent work on minimally supervised morphology induction from written corpora has met with some success (Creutz, 2006). We are curious how far this program can be pushed. From a practical perspective, minimally supervised morphology induction would help create morphological analysis systems for languages outside the traditional scope of NLP. However, to develop our method we induce the morphological structure of three well-understood languages, English, German, and Spanish.

1.1 Inherent Structure in NL Morphology

The approach we have taken to induce morphological structure has explicit roots in linguistic theory. Cross-linguistically, natural language organizes inflectional morphology into *paradigms* and *inflection classes*. A paradigm is a set of mutually exclusive operations that can be performed on a word form. Each mutually exclusive morphological operation in a paradigm marks a lexeme for some set or *cell* of morphosyntactic features. An inflection class, meanwhile, specifies the procedural details that a particular set of adherent lexemes follow to realize the surface form filling each paradigm cell. Each lexeme in a language adheres to a single inflection class for each paradigm the lexeme realizes. The lexemes belonging to an inflection class may have no relationship binding them together beyond an arbitrary morphological stipulation that they adhere to the same inflection class. But for this paper, an inflection class may

Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology, pages 117–125,
Prague, June 2007.

Paradigm Cells	Inflection Class	
	‘eat’	‘silent-e’
Unmarked	eat	dance, erase, …
Present, 3rd	eats	dances, erases, …
Past Tense	ate	danced, erased, …
Progressive	eating	dancing, erasing, …
Passive	eaten	danced, erased, …

Table 1: The English verbal paradigm, left column, and two inflection classes of the verbal paradigm. The verb *eat* fills the cells of its inflection class with the five surface forms shown in the second column. Verbs belonging to the ‘silent-e’ inflection class inflect following the pattern of the third column.

also refer to a set of lexemes that inflect similarly for phonological or orthographic reasons. Working with text we intentionally blur phonology and orthography.

A simple example will help illustrate paradigms, inflection classes, and the mutual exclusivity of cells. As shown in Table 1, all English verbs belong to a single common paradigm of five cells: One cell marks a verb for the morphosyntactic feature values *present tense 3rd person*, as in *eats*; another cell marks *past tense*, as in *ate*; a third cell holds a surface form typically used to mark *progressive aspect*, *eating*; a fourth produces a *passive participle*, *eaten*; and finally there is the unmarked cell, in this example *eat*.

Aside from inflection classes each containing only a few irregular lexemes, such as that containing *eat*, there are no English verbal inflection classes that arbitrarily differentiate lexemes on purely morphological grounds. There are, however, several inflection classes that realize surface forms only for verbs with particular phonology or orthography. The ‘silent-e’ inflection class is one such. To adhere to the ‘silent-e’ inflection class a lexeme must fill the unmarked paradigm cell with a form that ends in an unspoken character *e*, as in *dance*. The other paradigm cells in the ‘silent-e’ inflection class are filled by applying orthographic rules such as:

Progressive Aspect Cell – replace the final *e* of the unmarked form with the string *ing*, *dance* → *dancing*

Past Cell – substitute *ed*, *dance* → *danced*

Paradigm cells are mutually exclusive. In the English verbal paradigm, although English speakers can express progressive past actions with a grammatical construction, viz. *was eating*, there is no surface form of the lexeme *eat* that simultaneously fills both the *progressive* and the *past* cells of the verbal paradigm, **ateing*.

1.2 ParaMor

Paradigms and inflection classes, the inherent structure of natural language morphology, form the basis of ParaMor, our minimally supervised morphological induction algorithm. In ParaMor’s first phase, we find sets of mutually exclusive strings which closely mirror the inflection classes of a language—although ParaMor does not differentiate between syncretic word forms of the same lexeme filling different paradigm cells, such as *ed*-suffixed forms which can fill either the *past* or the *passive* cells of English verbs. In ParaMor’s second phase we employ the structured knowledge contained within the discovered inflection classes to segment word forms into morpheme-like pieces.

Languages employ a variety of morphological processes to arrive at grammatical word forms—processes including suffix-, prefix-, and infixation, reduplication, and template filling. Furthermore, the application of word forming processes often triggers phonological (or orthographic) change, such a as the dropped final *e* of the ‘silent-e’ inflection class, see Table 1. Despite the wide range of morphological processes and their complicating concomitant phonology, a large caste of inflection classes, and hence paradigms, can be represented as mutually exclusive substring substitutions. In the ‘silent-e’ inflection class, for example, the word-final strings *e.ed.es.ing* can be substituted for one another to produce the surface forms that fill the paradigm cells of lexemes belonging to this inflection class. In this paper we focus on identifying word final suffix morphology. While we focus on suffixes, the methods we employ can be straightforwardly generalized to prefixes and ongoing work seeks to model sequences of concatenative morphemes.

Inducing the morphology of a language from a naturally occurring text corpus is challenging. In languages with a rich morphological structure, surface forms filling particular cells of an inflection class may be relatively rare. In the Spanish newswire text over which we developed ParaMor there are 50,000 unique types. Among these types, instances of first and second person verb forms are few. The suffix *imos* which fills the *first person plural indicative present* cell for the *ir* verbal inflection class of Spanish occurs on only 77 unique lexemes. And yet we aim to identify candidate inflection classes which closely model the true inflection classes of a language, covering as many inflectional paradigm cells as possible.

Fortunately, we can leverage the paradigm structure of natural language morphology itself to retain many inflections which, because of data sparseness, might be missed if considered in isolation. ParaMor begins with a recall-centric search for partial candidate inflection classes. Many of the candidates which result from this initial search are incorrect. But intermingled with the false positives are candidates which collectively model significant fractions of true inflection classes. Hence, ParaMor's next step is to cluster the initial partial candidate inflection classes into larger groups. This clustering effectively uses the larger correct initial candidates as nuclei to which smaller correct candidates accrete. With as many initial true candidates as possible safely corralled with other candidates covering the same inflection class, ParaMor completes the paradigm discovery phase by discarding the large number of erroneous initially selected candidate inflection classes. Finally, with a strong grasp on the paradigm structure, ParaMor straightforwardly segments the words of a corpus into morphemes.

1.3 Related Work

In this section we highlight previously proposed minimally supervised approaches to the induction of morphology that, like ParaMor, draw on the unique structure of natural language morphology. One facet of NL morphological structure commonly leveraged by morphology induction algorithms is that morphemes are recurrent building blocks of words. Brent et al. (1995), Goldsmith (2001), and Creutz (2006) emphasize the building block nature of morphemes when they each use recurring word segments to efficiently encode a corpus. These approaches then hypothesize that those recurring segments which most efficiently encode a corpus are likely morphemes. Another technique that exploits morphemes as repeating sub-word segments encodes the lexemes of a corpus as a character tree, i.e. trie, (Harris, 1955; Hafer and Weis, 1974), or as a finite state automaton (FSA) over characters (Johnson, H. and Martin, 2003; Altun and M. Johnson, 2001). A trie or FSA conflates multiple instances of a morpheme into a single sequence of states. Because the choice of possible succeeding characters is highly constrained within a morpheme, branch points in the trie or FSA are likely morpheme boundaries. Often trie similarities are used as a first step followed by further processing to identify morphemes (Schone and Jurafsky, 2001).

The paradigm structure of NL morphology has also been previously leveraged. Goldsmith (2001) uses morphemes to efficiently encode a corpus, but he first groups morphemes into paradigm like structures he calls signatures. To date, the work that draws the most on paradigm structure is Snover (2002). Snover incorporates paradigm structure into a generative statistical model of morphology. Additionally, to discover paradigm like sets of suffixes, Snover designs and searches networks of partial paradigms. These networks are the direct inspiration for ParaMor's morphology scheme networks described in section 2.1.

2 ParaMor: Inflection Class Identification

2.1 Search

A Search Space: The first stage of ParaMor is a search procedure designed to identify partial inflection classes containing as many true productive suffixes of a language as possible. To search for these partial inflection classes we must first define a space to search over. In a naturally occurring corpus not all possible surface forms occur. In a corpus, each stem adhering to an inflection class will likely be observed in combination with only a subset of the suffixes in that inflection class. Each box in Figure 1 depicts a small portion of the empirical co-occurrence of suffixes and stems from a Spanish newswire corpus of 50,000 types. Each box in this figure contains a list of suffixes at the top in **bold**, together with the total number, and a few examples (in *italics*), of stems that occurred in separate word forms with each suffix in that box. For example, the box containing the suffixes **e**, **erá**, **ieron**, and **ió** contains the stems *deb* and *padec* because the word forms *debe*, *padece*, *deberá*, *padecerá*, etc. all occurred in the corpus. We call each possible pair of suffix and stem sets a *scheme*, and say that the **e.erá.ieron.ió** scheme covers the words *debe*, *padece,* etc. Note that a scheme contains both stems that occurred with exactly the set of suffixes in that scheme, as well as

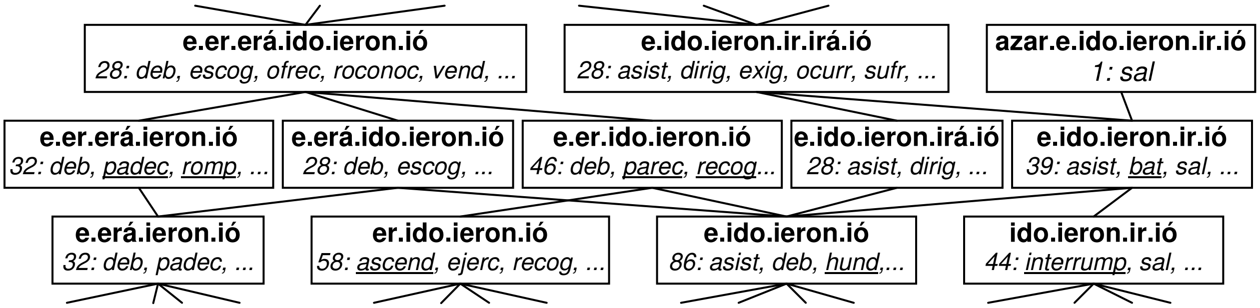

Figure 1: A small portion of a morphology scheme network—our search space of partial empirical inflection classes. This network was built from a Spanish Newswire corpus of 50,000 types, 1.26 million tokens. Each box contains a scheme. The suffixes of each scheme appear in **bold** at the top of each box. The total number of adherent stems for each scheme, together with a few exemplar stems, is in *italics*. Stems are underlined if they do not appear in any parent shown in this figure.

stems that occurred with suffixes beyond just those in the scheme. For example, in addition to the four suffixes **e**, **erá**, **ieron**, and **ió,** the stem *deb* occurred with the suffixes **er** and **ido**, as evident from the top left scheme **e.er.erá.ido.ieron.ió** which contains the stem *deb*. Intuitively, a scheme is a subset of the suffixes filling the paradigm cells of a true inflection class together with the stems that empirically occurred with that set of suffixes.

The schemes in Figure 1 cover portions of the *er* and the *ir* Spanish verbal inflection classes. The top left scheme of the figure contains suffixes in the *er* inflection class, while the top center scheme contains suffixes in the *ir* inflection class. The six suffixes in the top left scheme and the six suffixes in the top center scheme are just a few of the suffixes in the full *er* and *ir* inflection classes. As is fairly common for inflection classes across languages, the sets of suffixes in the Spanish *er* and *ir* inflection classes overlap. That is, verbs that belong to the *er* inflection class can take as a suffix certain strings of characters that verbs belonging to the *ir* inflection class can also take. The suffixes that are unique to the *er* verb inflection class in the top left scheme are **er** and **erá**; while the unique suffixes for the *ir* class in the top center scheme are **ir** and **irá**. In the third row of the figure, the scheme **e.ido.ieron.ió** contains only suffixes found in both the *er* and *ir* schemes.

While the example schemes in Figure 1 are correct and do occur in a real Spanish newswire corpus, the schemes are atypically perfect. There is only one suffix appearing in Figure 1 that is not a true suffix of Spanish—**azar** in the upper right scheme. In unsupervised morphology induction we do not know a priori the correct suffixes of a language. Hence, we form schemes by proposing candidate morpheme boundaries at every character boundary in every word, including the character boundary after the final character in each word form, to allow for empty suffixes.

Schemes of suffixes and their exhaustively co-occurring stems define a natural search space over partial inflection classes because schemes readily organize by the suffixes and stems they contain. We define a parent-child relationship between a parent scheme, P and a child scheme C, when P contains all the suffixes that C contains and when P contains exactly one more suffix than C. In Figure 1, parent child relations are represented by solid lines connecting boxed schemes. The scheme **e.er.erá.ido.ieron.ió**, for example, is the parent of three depicted children in Figure 1, one of which is **e.er.erá.ieron.ió**.

Our search strategy exploits a fundamental aspect of the relationship between parent and child schemes. Consider the number of stems in a parent scheme P as compared to the number of stems in any one of its children C. Since P contains all the suffixes which C contains, and because P only contains stems that occurred with every suffix in P, P can at most contain exactly the stems C contains and typically will contain fewer. In the Spanish corpus from which the scheme network of Figure 1 was built, 32 stems occur in forms with each of the five suffixes **e**, **er**, **erá**, **ieron**, and **ió** attached. But only 28 of these 32 stems occur in yet another form involving **ido**—the stem *deb* did but the stems *padec* and *romp* did not, for example.

A Search Strategy: To search for schemes which cover portions of the true inflection classes of a language, ParaMor's search starts at the bottom of the network. The lowest level in the scheme network consists of schemes which contain exactly one suffix together with all the stems that occurred in the corpus with that suffix attached. ParaMor considers each one-suffix scheme in turn beginning with that scheme containing the most stems, working toward schemes containing fewer. From each bottom scheme, ParaMor follows a single greedy upward path from child to parent. As long as an upward path takes at least one step, making it to a scheme containing two or more alternating suffixes, our search strategy accepts the terminal scheme of the path as likely modeling a portion of a true inflection class.

Each greedily chosen upward step is based on two criteria. The first criterion considers the number of adherent stems in the current scheme as compared to its parents' adherent sizes. A variety of statistics could judge the stem-strength of parent schemes: ranging from simple ratios through (dis)similarity measures, such as the dice coefficient or mutual information, to full fledged statistical tests. After experimenting with a range of such statistics we found, somewhat surprisingly, that measuring the ratio of parent stem size to child stem size correctly identifies parent schemes which contain only true suffixes just as consistently as more sophisticated tests. While a full report of our experiments is beyond the scope of this paper, the short explanation of this behavior is data sparseness. Many upward search steps start from schemes containing few stems. And when little data is available no statistic is particularly reliable.

Parent-child stem ratios have two additional computational advantages over other measures. First, they are quick to compute and second, the parent with the largest stem ratio is always that parent with the most stems. So, being greedy, each search step simply moves to that parent, P, with the most stems, as long as the parent-child stem ratio to P is large. The threshold above which a stem ratio is considered large enough to warrant an upward step is a free parameter. As the goal of this initial search stage is to identify schemes containing as wide a variety of productive suffixes as possible, we want to set the parent-child stem ratio threshold as low as possible. But a ratio threshold that is too small will allow search paths to schemes containing unproductive and spurious suffixes. In practice, for Spanish, we have found that setting the parent-child stem ratio cutoff much below 0.25 results in schemes that begin to include only marginally productive derivational suffixes. For this paper we leave the parent-child stem ratio cutoff parameter at 0.25.

Alone, stem strength assessments of parent schemes, such as parent-child stem ratios, falter as a search path nears the top of the morphology scheme network. Monotonically decreasing adherent stem size causes statistics that assess parents' stem-strength to become less and less reliable. Hence, the second criterion governing each search step helps to halt upward search paths before judging parents' worth becomes impossible. While there are certainly many possible stopping criteria, ParaMor's policy stops each upward search path when there is no parent scheme with more stems than it has suffixes. We devised this halting condition for two reasons. First, requiring each path scheme to contain more stems than suffixes attains high suffix recall. High recall results from setting a low bar for upward movement at the bottom of the network. Search paths which begin from schemes whose single suffix is rare in the text corpus can often take one or two upward search steps and reach a scheme containing the necessary three or four stems. Second, this halting criterion requires the top scheme of search paths that climb high in the network to contain a comparatively large number of stems. Reigning in high-reaching search paths before the stem count falls too far, captures path-terminal schemes which cover a large number of word types. In the second stage of ParaMor's inflection class identification phase these larger terminal schemes effectively vacuum up the useful smaller paths that result from the more rare suffixes. Figure 2 contains examples of schemes selected by ParaMor's initial search.

To evaluate ParaMor at paradigm identification, we hand compiled an answer key of the inflection classes of Spanish. This answer key contains nine productive inflection classes. Three contain the suffixes of the *ar*, *er*, and *ir* verbal inflection classes. There are two orthographically differentiated inflection classes for nouns in the answer key: one for nouns that form the plural by adding *s*, and one for nouns that take *es*. Adjectives in Spanish inflect for gender and number. Arguably, gender and number each constitute separate paradigms, each with two cells. But here we conflated these into a single inflection class with four cells. Finally, there are three inflection classes in our answer key covering Spanish clitics. Spanish verbal clitics behave orthographically as agglutinative sequences of suffixes.

1) **Ø.s** 5501 stems
2) **a.as.o.os** 892 stems
...
5) **a.aba.aban.ada.adas.ado.ados.an.ando.ar.aron.arse.ará.arán.ó** 25 stems
...
12) **a.aba.ada.adas.ado.ados.an.ando.ar.aron.ará.arán.e.en.ó** 21 stems
...
209) **e.er.ida.idas.ido.idos.imiento.ió** 9 stems
...
1590) **Ø.ipo** 4 stems
1591) **ido.idos.ir.iré** 6 stems
1592) **Ø.e.iu** 4 stems
1593) **iza.izado.izan.izar.izaron.izarán.izó** 8 stems
...

Figure 2: The suffixes of some schemes selected by the initial search over a Spanish corpus of 50,000 types. While some selected schemes contain large numbers of correct suffixes, such as the 1st, 2nd, 5th, 12th, 209th, and 1591st selected schemes; many others are incorrect collections of word final strings.

In a corpus of Spanish newswire text of 50,000 types and 1.26 million tokens, the initial search identifies schemes containing 92% of all ideal inflectional suffixes of Spanish, or 98% of the ideal suffixes that occurred at least twice in the corpus. There are selected schemes which contain portions of each of the nine inflection classes in the answer key. The high recall of the initial search comes, of course, at the expense of precision. While there are nine inflection-classes and 87 unique suffixes in the hand-built answer key for Spanish, 8339 schemes are selected containing 9889 unique candidate suffixes.

2.2 Clustering Partial Inflection Classes

While the third step of inflection class identification, discussed in Section 2.3, directly improves the initial search's low precision by filtering out bogus schemes, the second step, described here, conflates selected schemes which model portions of the same inflection class. Consider the fifth and twelfth schemes selected by ParaMor from our Spanish corpus, as shown in Figure 2. Both of these schemes contain a large number of suffixes from the Spanish *ar* verbal inflection class. And while each contains many overlapping suffixes, each possesses correct suffixes which the other does not. Meanwhile, the 1591st selected scheme contains four suffixes of the *ir* verbal inflection class, including the only instance of *iré* that occurs in any selected scheme. Containing only six stems, the 1591st scheme could accidentally be filtered out during the third phase of inflection class identification. Hence, the rationale for clustering initial selected schemes is two fold. First, by consolidating schemes which cover portions of the same inflection class we produce sets of suffixes which more closely model the paradigm structure of natural language morphology. And, second, corralling correct schemes safeguards against losing unique suffixes.

The clustering of schemes presents two unique challenges. First, we must avoid over-clustering schemes which model distinct inflection classes. As noted in Section 2.1, it is common, cross-linguistically, for the suffixes of inflection classes to overlap. Looking at Figure 2, we must be careful not to merge the 209th selected scheme, which models a portion of the *er* verbal inflection class, with the 1591st selected scheme, which models the *ir* class—despite these schemes sharing two suffixes, *ido* and *idos*. As the second challenge, the many small schemes which the search strategy produces act as distractive noise during clustering. While small schemes containing correct suffixes do exist, e.g. the 1591st scheme, the vast majority of schemes containing few stems and suffixes are incorrect collections of word final strings that happen to occur in corpus word forms attached to a small number of shared initial strings. ParaMor's clustering algorithm should, for example, avoid placing **Ø.s** and **Ø.ipo**, respectively the 1st and 1590th selected schemes, in the same cluster. Although **Ø.ipo** shares the null suffix with the valid nominal scheme **Ø.s**, the string 'ipo' is not a morphological suffix of Spanish.

To form clusters of related schemes while addressing both the challenge of observing a language's paradigm structure as well as the challenge of merging in the face of many small incorrectly selected schemes, ParaMor adapts greedy hierarchical agglomerative clustering. We modify vanilla bottom-up clustering by placing restrictions on which clusters are allowed to merge. The first restriction helps ensure that schemes modeling distinct but overlapping inflection classes remain separated. The restriction: do not place into the same cluster suffixes which share no stem in the corpus. This restriction retains separate clusters for separate inflection classes because a lexeme's stem occurring with suffixes unique to that lexeme's inflection class will not occur with suffixes unique to some other inflection class.

Alone, requiring all pairs of suffixes in a cluster to occur in the corpus with some common stem will not prevent small bogus schemes, such as **Ø.ipo** from attaching to correct schemes, such as **Ø.s**—the **ipo.s** scheme contains two 'stems,' the word form initial strings 'ma' and 't'. And so a second restriction is required. This second restriction employs a heuristic specifically adapted to ParaMor's initial search strategy. As discussed in Section 2.1, in addition to many schemes which contain only few suffixes, ParaMor's initial network search also identifies multiple overlapping schemes containing significant subsets of the suffixes in an inflection class. The 5th, 12th, and 209th selected schemes of Figure 2 are three such larger schemes. ParaMor restricts cluster merges heuristically by requiring at least one large scheme for each small scheme the cluster contains, where we measure the size of a scheme as the number of unique word forms it covers. The threshold size above which schemes are considered large is the second of ParaMor's two free parameters. The scheme size threshold is reused during ParaMor's filtering stage. We discuss the unsupervised procedure we use to set the size threshold when we present the details of cluster filtering in Section 2.3.

We have found that with these two cluster restrictions in place, the particular metric we use to measure the similarity of scheme-clusters does not significantly affect clustering. For the experiments we report here, we measure the similarity of scheme-clusters as the cosine between the sets of all possible stem-suffix pairs the clusters contain. A stem-suffix pair occurs in a cluster if some scheme belonging to that cluster contains both that stem and that suffix. With these adaptations, we allow agglomerative clustering to proceed until there are no more clusters that can legally be merged.

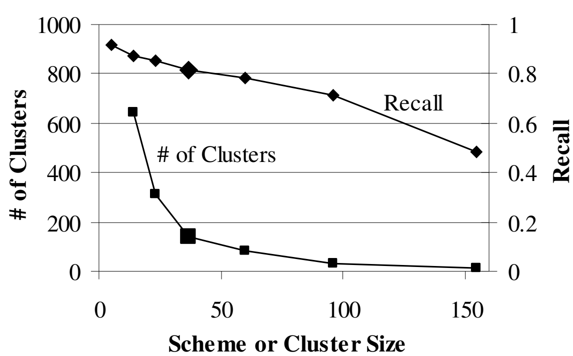

Figure 3: The # of clusters and their recall of unique Spanish suffixes as the scheme-cluster size cutoff is varied. The value of each function at the threshold we use in all experiments reported in this paper is that of the larger symbol.

2.3 Filtering of Inflection Classes

With most valid schemes having found a safe haven in a cluster with other schemes modeling the same inflection class, we turn our attention to improving scheme-cluster precision. ParaMor applies a series of filters, culling out unwanted scheme-clusters. The first filter is closely related to the cluster restriction on scheme size discussed in Section 2.2. ParaMor discards all unclustered schemes falling below the size threshold used during clustering. Figure 3 graphs the number of Spanish clusters which survive this size-based filtering step as the threshold size is varied. Figure 3 also contains a plot of the recall of unique Spanish suffixes as a function of this threshold. As the size threshold is increased the number of remaining clusters quickly drops. But suffix recall only slowly falls during the steep decline in cluster count, indicating ParaMor discards mostly bogus schemes containing illicit suffixes. Because recall is relatively stable, the exact size threshold we use should have only a minor effect on ParaMor's final morphological analyses. In fact, we have not fully explored the ramifications various threshold values have on the final morphological word segmentations, but have simply picked a reasonable setting, 37 covered word types. At this threshold, the number of scheme-clusters is reduced by more than 98%, while the number of unique candidate suffixes in any cluster is reduced by more than 85%. Note that the initial number of selected schemes, 8339, falls outside the scale of Figure 3.

Of the scheme-clusters which remain after size based filtering is complete, by far the largest category of incorrect clusters contains schemes which, like the 1593rd selected scheme, shown in Figure 2, incorrectly hypothesize morpheme boundaries one or more characters to the left of the true boundary. To filter out these incorrectly segmented clusters we use a technique inspired by Harris (1955). For each initial string common to all suffixes in the cluster, for each scheme in the cluster, we examine the network scheme containing the suffixes formed by stripping the initial string from the scheme's suffixes. We then measure the entropy of leftward trie characters of the stripped scheme. If the entropy is large, then the character stripped scheme is likely at a morpheme boundary and the original scheme is likely modeling an incorrect morpheme boundary. This algorithm would throw out the 1593rd selected scheme because the stems in the scheme **a.ado.an.ar.aron.arán.ó** end in a wide variety of characters, yielding high trie entropy, and signaling a likely morpheme boundary. Because we apply morpheme boundary filtering after we have clustered, the redundancy of the many schemes in the cluster makes this filter quite robust, letting us set the cutoff parameter as low as we like avoiding another free parameter.

2.4 Segmentation and Evaluation

Word segmentation is our final step of morphological analysis. ParaMor's current segmentation algorithm is perhaps the most simple paradigm inspired segmentation algorithm possible. Essentially, ParaMor strips off suffixes which likely participate in a paradigm. To segment any word, w, ParaMor identifies all scheme-clusters that contain a non-empty suffix that matches a word final string of w. For each such matching suffix, $f \in C$, where C is the cluster containing f, we strip f from w obtaining a stem t. If there is some second suffix $f' \in C$ such that $t.f'$ is a word form found in either of the training or the test corpora, then ParaMor proposes a segmentation of w between t and f. ParaMor, here, identifies f and f' as mutually exclusive suffixes from the same paradigm. If ParaMor finds no complex analysis, then we propose w itself as the sole analysis of the word. Note that for each word form, ParaMor may propose multiple separate segmentation analyses each containing a single proposed stem and suffix.

To evaluate ParaMor's morphological segmentations we follow the methodology of Morpho Challenge 2007 (Kurimo et al., 2007), a minimally supervised morphology induction competition. Word segmentations are evaluated in Morpho Challenge 2007 by comparing against hand annotated morphological analyses. The correctness of proposed morphological analyses is computed in Morpho Challenge 2007 by comparing pairs of word forms which share portions of their analyses. Recall is measured by first sampling pairs of words from the answer analyses which share a stem or morphosyntactic feature and then noting if that pair of word forms shares a morpheme in any of their proposed analyses. Precision is measured analogously, sampling morpheme-sharing pairs of words from the proposed analyses and noting if that pair of words shares a feature in any correct analysis of those words.

We evaluate ParaMor on two languages not examined during the development of ParaMor's induction algorithms: English and German. And we evaluate with each of these two languages at two tasks:

1. Analyzing inflectional morphology alone
2. Jointly analyzing inflectional and derivational morphology.

We constructed Morpho Challenge 2007 style answer keys for each language and each task using the Celex database (Burnage, 1990). The English and German corpora we test over are the corpora available through Morpho Challenge 2007. The English corpus contains nearly 385,000 types, while the German corpus contains more than 1.26 million types. ParaMor induced paradigmatic scheme-clusters over these larger corpora by reading just the top 50,000 most frequent types. But with the scheme-clusters in hand, ParaMor segmented all the types in each corpus.

We compare ParaMor to Morfessor v0.9.2 (Creutz, 2006), a state-of-the-art minimally supervised morphology induction algorithm. Morfessor has a single free parameter. To make for stiff competition, we report results for Morfessor at that parameter setting which maximized F_1 on each separate test scenario. We did not vary the two free parameters of ParaMor, but hold each of ParaMor's parameters at a setting which produced reasonable *Spanish* suffix sets, see sections 2.1-2.2. Table 2 contains the evaluation results. To estimate the variance of our experimental results we measured Morpho Challenge 2007 style precision, recall, and F_1 on multiple non-overlapping pairs of 1000 feature-sharing words.

Neither ParaMor nor Morfessor arise in Table 2 as clearly superior. Each algorithm outperforms the other at F_1 in some scenario. Examining precision and recall is more illuminating. ParaMor attains particularly high recall of inflectional affixes for both English and German. We conjecture that ParaMor's strong performance at identifying inflectional morphemes comes from closely modeling the natural paradigm structure of language. Conversely, Morfessor places its focus on precision and does not rely on any property exclusive to inflectional (or derivational) morphology. Hence,

	Inflectional Morphology Only								Inflectional & Derivational Morphology							
	English				German				English				German			
	P	R	F_1	σ	P	R	F_1	σ	P	R	F_1	σ	P	R	F_1	σ
Morfessor	53.3	47.0	**49.9**	1.3	38.7	44.2	41.2	0.8	**73.6**	34.0	46.5	1.1	**66.9**	37.1	**47.7**	0.7
ParaMor	33.0	**81.4**	47.0	0.9	42.8	**68.6**	**52.7**	0.8	48.9	53.6	**51.1**	0.8	60.0	33.5	43.0	0.7

Table 2: ParaMor segmentations compared to Morfessor's (Creutz, 2006) evaluated for **P**recision, **R**ecall, $\mathbf{F_1}$, and standard deviation of F_1, **σ**, in four scenarios. Segmentations over English and German are each evaluated against correct morphological analyses consisting, on the left, of inflectional morphology only, and on the right, of both inflectional and derivational morphology.

Morfessor attains high precision with reasonable recall when graded against an answer key containing both inflectional and derivational morphology.

We are excited by ParaMor's strong performance and are eager to extend our algorithm. We believe the precision of ParaMor's simple segmentation algorithm can be improved by narrowing down the proposed analyses for each word to the most likely. Perhaps ParaMor and Morfessor's vastly different strategies for morphology induction could be combined into a hybrid strategy more successful than either alone. And ambitiously, we hope to extend ParaMor to analyze languages with agglutinative sequences of affixes by generalizing the definition of a scheme.

Acknowledgements

The research reported in this paper was funded in part by NSF grant number IIS-0121631.

References

Altun, Yasemin, and Mark Johnson. "Inducing SFA with ϵ-Transitions Using Minimum Description Length." *Finite State Methods in Natural Language Processing Workshop at ESSLLI* Helsinki: 2001.

Brent, Michael R., Sreerama K. Murthy, and Andrew Lundberg. "Discovering Morphemic Suffixes: A Case Study in MDL Induction." *The Fifth International Workshop on Artificial Intelligence and Statistics* Fort Lauderdale, Florida, 1995.

Burnage, Gavin. *Celex—A Guide for Users*. Springer, Centre for Lexical information, Nijmegen, the Netherlands, 1990.

Creutz, Mathias. "Induction of the Morphology of Natural Language: Unsupervised Morpheme Segmentation with Application to Automatic Speech Recognition." Ph.D. Thesis in Computer and Information Science, Report D13. Helsinki: University of Technology, Espoo, Finland, 2006.

Goldsmith, John. "Unsupervised Learning of the Morphology of a Natural Language." *Computational Linguistics* 27.2 (2001): 153-198.

Hafer, Margaret A., and Stephen F. Weiss. "Word Segmentation by Letter Successor Varieties." *Information Storage and Retrieval* 10.11/12 (1974): 371-385.

Harris, Zellig. "From Phoneme to Morpheme." *Language* 31.2 (1955): 190-222. Reprinted in Harris 1970.

Harris, Zellig. *Papers in Structural and Transformational Linguists*. Ed. D. Reidel, Dordrecht 1970.

Johnson, Howard, and Joel Martin. "Unsupervised Learning of Morphology for English and Inuktitut." *Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*. Edmonton, Canada: 2003.

Kurimo, Mikko, Mathias Creutz, and Matti Varjokallio. "Unsupervised Morpheme Analysis – Morpho Challenge 2007." March 26, 2007. <http://www.cis.hut.fi/morphochallenge2007/>

Schone, Patrick, and Daniel Jurafsky. "Knowledge-Free Induction of Inflectional Morphologies." *North American Chapter of the Association for Computational Linguistics (NAACL)*. Pittsburgh, Pennsylvania: 2001. 183-191.

Snover, Matthew G. "An Unsupervised Knowledge Free Algorithm for the Learning of Morphology in Natural Languages." Sever Institute of Technology, Computer Science Saint Louis, Missouri: Washington University, M.S. Thesis, 2002.