

A Naive Theory of Affixation and an Algorithm for Extraction

Harald Hammarström
Dept. of Computing Science
Chalmers University of Technology
412 96, Gothenburg Sweden
harald2@cs.chalmers.se

Abstract

We present a novel approach to the unsupervised detection of affixes, that is, to extract a set of salient prefixes and suffixes from an unlabeled corpus of a language. The underlying theory makes no assumptions on whether the language uses a lot of morphology or not, whether it is prefixing or suffixing, or whether affixes are long or short. It does however make the assumption that 1. salient affixes have to be frequent, i.e occur much more often than random segments of the same length, and that 2. words essentially are variable length sequences of random characters, e.g a character should not occur in far too many words than random without a reason, such as being part of a very frequent affix. The affix extraction algorithm uses only information from fluctuation of frequencies, runs in linear time, and is free from thresholds and untransparent iterations. We demonstrate the usefulness of the approach with example case studies on typologically distant languages.

1 Introduction

The problem at hand can be described as follows:

Input : An unlabeled corpus of an arbitrary natural language

Output : A (possibly ranked) set of prefixes and suffixes corresponding to true prefixes and suf-

fixes in the linguistic sense, i.e well-segmented and with grammatical meaning, for the language in question.

Restrictions : We consider only concatenative morphology and assume that the corpus comes already segmented on the word level.

The theory and practice of the problem is relevant or even essential in fields such as child language acquisition, information retrieval and, of course, the fuller scope of computational morphology and its further layers of application (e.g Machine Translation).

The reasons for attacking this problem in an unsupervised manner include advantages in elegance, economy of time and money (no annotated resources required), and the fact that the same technology may be used on new languages.

An outline of the paper is as follows: we start with some notation and basic definitions, with which we describe the theory that is intended to model the essential behaviour of affixation in natural languages. Then we describe in detail and with examples the thinking behind the affix extraction algorithm, which actually requires only a few lines to define mathematically. Next, we present and discuss some experimental results on typologically different languages. The paper then finishes with a brief but comprehensive characterization of related work and its differences to our work. At the very end we state the most important conclusions and ideas on future components of unsupervised morphological analysis.

2 A Naive Theory of Affixation

Notation and definitions:

- $w, s, b, x, y, \dots \in \Sigma^*$: lowercase-letter variables range over strings of some alphabet Σ and are variously called words, segments, strings, etc.
- $s \triangleleft w$: s is a terminal segment of the word w i.e there exists a (possibly empty) string x such that $w = xs$
- $W, S, \dots \subseteq \Sigma^*$: capital-letter variables range over sets of words/strings/segments
- $f_W(s) = |\{w \in W | s \triangleleft w\}|$: the (suffix) frequency, i.e the number of words in W with terminal segment s
- $S_W = \{s | s \triangleleft w \in W\}$: all terminal segments of the words in W
- $u f_W(u) = |\{(x, y) | xuy = w \in W\}|$: the substring frequency of u , i.e the number times u occurs as a substring in the set of words W (x and y may be empty).
- $n f_W(u) = u f_W(u) - f_W(u)$: the non-final frequency of u , i.e. the substring frequency minus those in which it occurs as a suffix.
- $|\cdot|$: is overloaded to denote both the length of a string and the cardinality of a set

Assume we have two sets of random strings over some alphabet Σ :

- Bases $B = \{b_1, b_2, \dots, b_m\}$
- Suffixes $S = \{s_1, s_2, \dots, s_n\}$

Such that:

Arbitrary Character Assumption (ACA) Each character $c \in \Sigma$ should be equally likely in any word-position for any member of B or S .

Note that B and S need not be of the same cardinality and that any string, including the empty string, could end up belonging to both B and S . They need neither to be sampled from the same distribution; pace the requirement, the distributions

from which B and S are drawn may differ in how much probability mass is given to strings of different lengths. For instance, it would not be violation if B were drawn from a distribution favouring strings of length, say, 42 and S from a distribution with a strong bias for short strings.

Next, build a set of affixed words $W \subseteq \{bs | b \in B, s \in S\}$, that is, a large set whose members are concatenations of the form bs for $b \in B, s \in S$, such that:

Frequent Flyer Assumption (FFA) : The members of S are frequent. Formally: Given any $s \in S$: $f_W(s) \gg f_W(x)$ for all x such that 1. $|x| = |s|$; and 2. not $x \triangleleft s'$ for all $s' \in S$.

In other words, if we call $s \in S$ a *true suffix* and we call x an *arbitrary segment* if it neither a true suffix nor the terminal segment of a true suffix, then any true suffix should have much higher frequency than an arbitrary segment of the same length.

One may legitimately ask to what extent words of real natural languages fit the construction model of W , with the strong ACA and FFA assumptions, outlined above. For instance, even though natural languages often aren't written phonemically, it is not hard to come up with languages that have phonotactic constraints on what may appear at the beginning or end of a word, e.g, Spanish **st-* may not begin a word and yields *est-* instead. Another violation of ACA is that (presumably all (Ladefoged, 2005)) languages disallow or disprefer a consonant vs. a vowel conditioned by the vowel/consonant status of its predecessor. However, if a certain element occurs with *less* frequency than random (the best example would be click consonants which, in some languages e.g Eastern !Xõo (Traill, 1994), occur only initially), this will not be a practical problem.

As for FFA, we may have breaches such as Biblical Aramaic (Rosenthal, 1995) where an old $-ā$ element appears on virtually everywhere on nouns, making it very frequent, but no longer has any synchronic meaning. Also, one can doubt the requirement that an affix should need to be frequent; for instance, the Classical Greek inflectional (lacking synchronic internal segmentation) alternative medial 3p. pl. aorist imperative ending $-\sigma\theta\omega\nu$ (Blomqvist and Jastrup, 1998), is not common at all.

Positions	Distance
$\ p_1 - p_2\ $	0.47
$\ p_1 - p_3\ $	0.36
$\ p_1 - p_4\ $	0.37
$\ p_2 - p_3\ $	0.34
$\ p_2 - p_4\ $	0.23
$\ p_3 - p_4\ $	0.18

Table 1: Difference between character distributions according to word position.

Just how realistic the assumptions are is an empirical question, whose answer must be judged by experiments on the relevant languages. In the absence of fully annotated test sets for diverse languages, and since the author does not have access to the Hutmegs/CELEX gold standard sets for Finnish and English (Creutz and Lindén, 2004), we can only give some guiding experimental data.

ACA On a New Testament corpus of Basque (Leizarraga, 1571) we computed the probability of a character appearing in the initial, second, third or fourth position of the word. Since Basque is entirely suffixing, if it complied to ACA, we’d expect the distributions to be similar. However, if we look at the difference of the distributions in terms of variation distance between two probability distributions ($\|p - q\| = \frac{1}{2} \sum_x |p(x) - q(x)|$), it shows that they differ considerably – especially the initial position proves more special (see table 1).

FFA As for the FFA, we checked a corpus of bible portions of Warlpiri (Summer Institute of Linguistics, 2001). This was chosen because it is one of the few languages known to the author where data was available and which has a decent amount of frequent suffixes which are also long, e.g case affixes are typically bisyllabic phonologically and five-ish characters long orthographically. Since the orthography used marked segmentation, it was easy to compute FFA statistics on the words as removed from segmentation marking. Comparing with the lists in (Nash, 1980, Ch. 2) it turns out that FFA is remarkably stable for all grammatical suffixes occurring in the outermost layer. There are however the expected kind of breaches; e.g

a tense suffix *-ku* combined with a last vowel *-u* which is frequent in some frequent preceding affixes making the terminal segment *-uku* more frequent than some genuine three-letter suffixes.

The language known to the author which has shown the most systematic disconcord with the FFA is Haitian Creole (also in bible corpus experiments (American Bible Society, 1999)). Haitian creole has very little morphology of its own but owes the lion’s share of its words from French. French derivational morphemes abound in these words, e.g *-syon*, which have been carefully shown by (Lefebvre, 2004) not to be productive in Haitian Creole. Thus, the little morphology there is in Haitian creole is very difficult to get at without also getting the French relics.

3 An Algorithm for Affix Extraction

The key question is, if words in natural languages are constructed as W explained above, can we recover the segmentation? That is, can we find B and S , given only W ? The answer is yes, we can partially decide this. To be more specific, we can compute a score Z_W such that $Z_W(x) > Z_W(y)$ if $x \in S$ and $y \notin S$. In general, the converse need not hold, i.e if both $x, y \in S$, or both $x, y \notin S$, then it may still be that $Z_W(x) > Z_W(y)$. This is equivalent to constructing a ranked list of all possible segments, where the true members of S appear at the top, and somewhere down the list the junk, i.e non-members of S , start appearing and fill up the rest of the list. Thus, it is not said *where* on the list the true-affixes/junk border begins, just that there is a consistent such border.

Now, how should this list be computed? All terminal segments are contained in the set S_W , the question is just to order them. We shall now define three properties that we argue will be enough to put the S -belonging affixes at the top. For a terminal segment s , define:

Frequency The frequency $f_W(s)$ of s (as a terminal segment).

Curve Drop First, for s , define its curve $C_s(c)$ which is a probability distribution on Σ :

$$C_s(c) = \frac{f_W(cs)}{f_W(s)}$$

Next, more importantly, define its *curve drop* $\overline{C}(s)$ which is a value in $[0, 1]$:

$$\overline{C}(s) = \frac{1 - \max_c(C_s(c))}{1 - \frac{1}{|\Sigma|}}$$

Random Adjustment First, for s , define its probability as:

$$P_W(s) = \frac{f_W(s)}{\sum_{s' \in S_W} f_W(s')}$$

Second, equally straightforwardly, for an arbitrary segment u , define its non-final probability as:

$$nP_W(u) = \frac{nf_W(u)}{\sum_{u'} nf_W(u')}$$

Finally, for a terminal segment s , define its *random adjustment* $RA(s)$ which a value in Q^+ :

$$RA(s) = \begin{cases} \frac{P_W(s)}{nP_W(s)} & \text{if } nP_W(s) > 0 \\ 1.0 & \text{otherwise} \end{cases}$$

It is appropriate now to show the intuition behind the definitions. There isn't much to comment on frequency, so we'll go to curve drop and random adjustment. All examples in this section come from the Brown corpus (Francis and Kucera, 1964) of one million tokens ($|W| = 47178$ and $|S_W| = 154407$).

The curve drop measure is meant to predict when a suffix is well-segmented to the left. Consider a suffix s , in all the words on which it appears, there is a preceding character c . Figure 1 shows examples of the frequency distribution on preceding character for example suffixes *-ing* and *-ng*. The reasoning is as follows. If s is a true suffix and is well-segmented to the left, then its curve-drop value should be high. Frequent true suffixes that attach to bases whose last character is random should have a close to uniform curve. On the other hand, if the curve drop value is low it means there is a character that suspiciously often precedes s . However, if s weren't a true suffix to

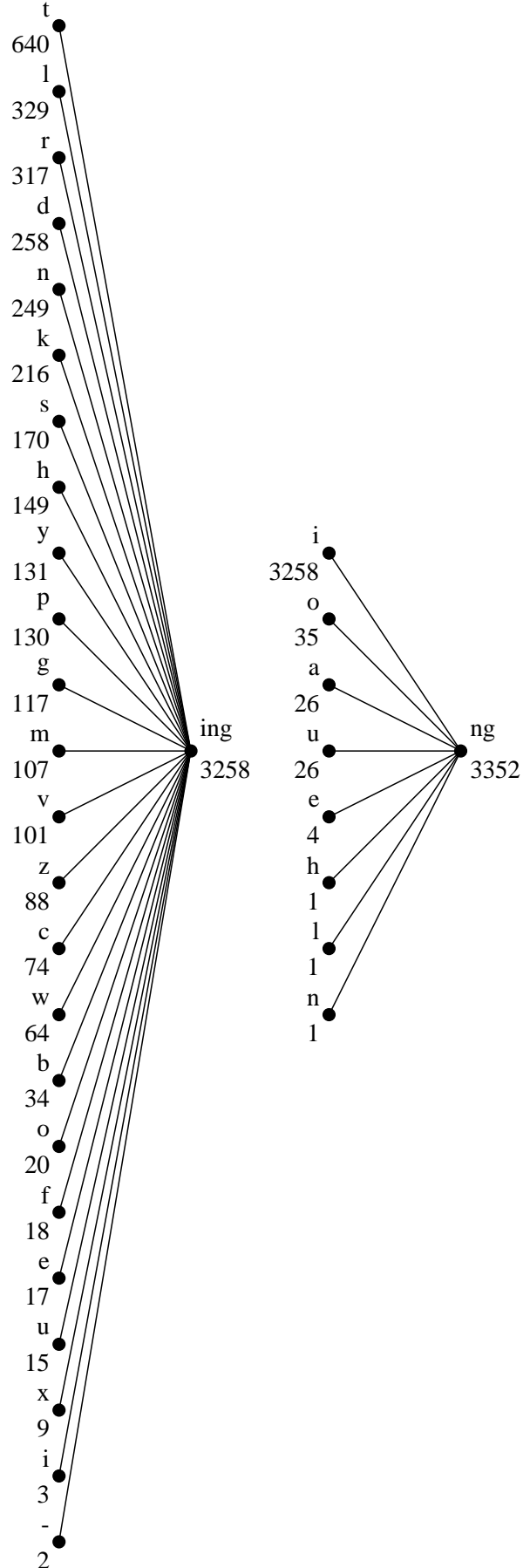


Figure 1: The curve frequencies giving rise to the curves C_{ing} and C_{ng} respectively.

begin with, perhaps just a frequent but random character, then we expect it's curve drop value to be high too! To exemplify this, we have $\overline{C}(ing) \approx 0.833$, $\overline{C}(ng) \approx 0.029$ and $\overline{C}(a) \approx 0.851$.

The random adjustment measure it precisely to distinguish what a "frequent but random segment" is, that is, discriminate e.g *-a* versus *-ing* as well as *-a* versus *-ng*. Now, how does one know whether something is random or not? One approach would be to say the shorter the segment the more random. Although it's possible to get this to work reasonably well in practice, it has some drawbacks. First, it treats all segments of the same length the same, which may be too brutal, e.g should *-s* be penalized as much as *-a*? Second, it might be considered too vulnerable to orthography. For example if a language has an odd trigraph for some phoneme, we are clearly going to introduce an error source. Instead we propose that a segment is random iff it has similar probability in any position of the word. This avoids the "flat length"-problems but has others, which we think are less harmful. First, we might get sparse data which can either be back-off smoothed or, like here, effectively ignored (where we lack occurrence we set the *RA* to 1.0). Second, phonotactic or orthographic constraints may cause curiosities, e.g. English *y* is often spelled *i* when medial as in *fly* vs. *flies*.

To put it all together, we propose the characterization of suffixes in terms of the three properties as shown in table 2. The terms high and low are of course idealized, as they are really gradient properties.

As seen from the table, we hold that true suffixes (and only true suffixes) are those which have a high value for all three properties. Therefore, we define our final ranking score, the $Z_W : S_W \rightarrow \mathbf{Q}$:

$$Z_W(s) = \overline{C}(s) \cdot RA(s) \cdot f_W(s) \quad (1)$$

Thus we are deliberate saying that if you have a not-so-high relative value for one of the properties, you can compensate to some extent by having very high relative values for the other properties (relative here means relative to the corresponding values of other suffixes). It is instructive to look at what happens in a few interesting cases:

1. We have two suffixes such that one is an en-

largement of the other by a random segment, e.g *-ting* versus *-ing*, where the true suffix is the shorter one.

In this case, we expect both to have similar high \overline{C} , the longer one should have higher *RA* and, by necessity, the shorter one should have significantly higher frequency. Example values for *-ing* versus *-ting* are shown in table 3.

Here, we see that the shorter wins out and we can use that fact to weed out the longer one (cf. purging below). (One might think that in a "perfect" situation, the f_W and *RA* would cancel out, leaving the situation a tie. However, *RA* will not cancel f_W in a language which, like all language I know of, has more non-final than final "positions of segments in words", and also, ceteris paribus, we expect a higher frequency to yield a more reliable curve drop value.)

2. We have two suffixes such that one is a tail of the other, but both are true suffixes, and they just happen to share a segment e.g *-ly* versus *-y*.

In this case, we succeed in keeping both if the longer wins out on a better curve-drop and random adjustment. In fact, as shown in table 3 this is exactly what happens with *-ly* versus *-y*.

3. We have two true suffixes which incidentally share an ending which is not a true suffix. Although easy to find in other languages, I failed to find an example of this in English without confounding factors, but we can imagine one, for example *-xz* versus *-yz*. Given the assumption that *-z* itself is not a true suffix, $f_W(z)$ should not be many times higher than $f_W(xz) + f_W(yz)$, thus its curve-drop not many percent, if at all, higher than 0.5, and of course, $RA(z) \approx 1$. On the other hand, by assumption of being true suffixes, *-xz* and *-yz* should have high curve-drop values, and outperform *-z* on *RA*.

Empirically, the prediction is wrong in the case *-est* versus *-est*, as shown in 3 but *-ist* and *-est* can hardly be said to satisfy FFA.

4. We have two true stacked suffixes which share an ending and this ending is also true suffix, e.g

f_W	\bar{C}	RA	Example	Label
high	high	high	<i>-ing</i>	True suffix
high	high	low	<i>-a</i>	Frequent random segment
high	low	high	<i>-ng</i>	Tail of true suffix
high	low	low	N/A	Second part of a digraph
low	high	high	<i>-oholic</i>	Infrequent true suffix
low	high	low	<i>-we</i>	Happenstance low RA-segment?
low	low	high	<i>-icz</i>	Tail of foreign personal name ending
low	low	low	<i>-ebukadnessar</i>	Infrequent segment

Table 2: The logically possible configurations of the three suffix properties, accompanied by an appropriate linguistically inspired label and an example from English.

s	$f_W(s)$	$\bar{C}(s)$	$RA(s)$	$Z_W(s)$
<i>-ing</i>	3258	0.83	19.6	53309.3
<i>-ting</i>	640	0.69	31.5	13929.5
<i>-y</i>	3931	0.63	5.8	14402.7
<i>-ly</i>	1532	0.76	23.4	27282.2
<i>-t</i>	2796	0.74	0.50	1040.6
<i>-st</i>	561	0.64	0.68	246.3
<i>-ist</i>	202	0.81	1.29	213.9
<i>-est</i>	213	0.88	1.82	341.4
<i>-s</i>	11220	0.80	2.49	22514.8
<i>-ings</i>	205	0.89	60.5	11034.2
<i>-ations</i>	215	0.86	110.9	20482.1

Table 3: Values for some borderline cases.

-ations versus *-ings*.

As opposed to the above case, *-s* will appear in a lot of other places than after *-ing* and *-ation*, and is consequently given a higher score as shown in table 3.

As these considerations exemplify, the formal criterion mostly conforms to linguistic analysis, but as noted as noted in the third example, the outcomes occasionally disconcords with linguistic analysis.

A theoretical weakness with the RA -value as computed at present is when applied to languages which stack suffixes after each other. English does this to a small extent, as in *-ing* vs. *-ings*. In such cases, when calculating the non-final frequency of *-ing* one would like to count an occurrence of *-ing* in *-ings* as a final occurrence. But this would require knowing beforehand that *-s* is a true suffix as opposed to *-ings*. Fortunately, the impact of this drawback, also in other languages such as Turkish, ap-

pears not to be crucial. Even if suffixes occur when they are “almost” final, they still don’t occur in the when initial or in the mid-span of the word.

As a last discussion note, it is tempting to leave out the f_W -component in the calculation of the rank- ing . The frequency is really only needed when deciding between suffixes which are tails of each other – it plays no crucial role in ranking between suffixes which don’t share a tail. If frequencies are used only to purge out losers in tail-indexed sets of suffixes, the resulting list will also contain some non-FFA true suffixes but also too many spurious things, such as foreign personal name endings.

To sum up, the final Z_W -score in equation 1 is the one that purports to have the property that $Z_W(x) > Z_W(y)$ if $x \in S_W$ and $y \notin S_W$ – at least if purged (see below). We cannot give a formal proof that languages satisfying ACA and FFA should get a faultless ranking list because this is true only in a heuristic sense. To set bounds on the probability for it to hold is also depends on a lot of factors that are hard, or at least inelegant, to characterize. We hope, however, to have sketched the how the ACA and FFA assumptions are used.

A summary of the algorithm described in this section is displayed in table 4.

The time-complexity bounding factor is the number of (final and non-final) segments, which is linear (in the size of the input) if words are bounded in length by a constant and quadratic in the (really) worst case if not.

Input: A text corpus C

Step 1. Extract the set of words W from C (thus all contextual and word-frequency information is discarded)

Step 2. Calculate $f_W(s)$, $\overline{C}(s)$ and $RA(s)$ for each $s \in S_W$

Step 3. Combine $Z_W(s) = \overline{C}(s) \cdot RA(s) \cdot f_W(s)$

Table 4: Summary of affix-extraction algorithm.

4 Experimental Results

For an English bible corpus (King James, 1977) we get the top 30 plus bottom 3 suffixes as shown in table 5.

English has little affixation compared to e.g Turkish which is at the opposite end of the typological scale (Dryer, 2005). The corresponding results for Turkish on a bible corpus (American Bible Society, 1988) is shown in table 6.

The results largely speak for themselves but some comments are in order. As is easily seen from the lists, some suffixes are suffixes of each other so one could *purge* the list in some way to get only the most “competitive” suffixes. One purging strategy would be to remove x from the list if there is a z such that $x = yx$ and $Z_W(z) > Z_W(x)$ (this would remove e.g *-ting* if *-ing* is above it on the list). A more sophisticated purging method is the following, which does slightly more. First, for a word $w \in W$ define its best segmentation as: $Segment(w) = \operatorname{argmax}_{s \leftarrow w} Z_W(s)$. Then purge by keeping only those suffixes which are the best parse for at least one word: $S'_W = \{s \in S_W | \exists w Segment(w) = s\}$.

Such purging kicks out the bulk of “junk” suffixes. Table 7 shows the numbers for English, Turkish and the virtually affixless Maori (Bauer et al., 1993). It should be noted that “junk” suffixes still remain after purging – typically common stem-final characters – and that there is no simple relation between the number of suffixes left after purging and the amount of morphology of the language in question. Otherwise we would have expected the morphology-less Maori to be left with no, or 28-ish,

<i>-ed</i>	15448.4	<i>-s</i>	3407.3
<i>-eth</i>	12797.1	<i>-ions</i>	2684.5
<i>-ted</i>	11899.4	<i>-est</i>	2452.6
<i>-iah</i>	11587.5	<i>-sed</i>	2313.7
<i>-ly</i>	10571.2	<i>-y</i>	2239.2
<i>-ings</i>	8038.9	<i>-leth</i>	2166.3
<i>-ing</i>	7292.8	<i>-nts</i>	2122.6
<i>-ity</i>	6917.6	<i>-ied</i>	1941.7
<i>-edst</i>	6844.7	<i>-ened</i>	1834.9
<i>-ites</i>	5370.2	<i>-ers</i>	1819.5
<i>-seth</i>	5081.6	<i>-ered</i>	1796.7
<i>-ned</i>	4826.7	<i>-ded</i>	1582.2
<i>-s'</i>	4305.2	<i>-neth</i>	1540.0
<i>-nded</i>	3833.8
<i>-ts</i>	3783.1	<i>-ig</i>	0.0
<i>-ah</i>	3766.9	<i>-io</i>	0.0
<i>-ness</i>	3679.3	<i>-ti</i>	0.0

Table 5: Top 30 and bottom 3 extracted suffixes for an English bible corpus. The high placement of English *-eth* and *-iah* are due to the fact that the bible version used has *drinketh*, *sitteth* etc and a lot of personal names in *-iah*.

suffixes or at least less than English.

A good sign is that the purged list and its order seems to be largely independent of corpus size (as long as the corpus is not very small) but we do get some significant differences between bible English and newspaper English.

We have chosen to illustrate using affixes but the method readily generalizes to prefixes as well and even prefixes and suffixes at the same time. As an example of this, we show top-10 purged prefix-suffix scores in the same table also for some typologically differing languages in table 8. Again, we use bible corpora for cross-language comparability (Swedish (Svenska Bibelsällskapet, 1917) and Swahili (British and Foreign Bible Society, 1953)). The scores have been normalized in each language to allow cross-language comparison – which, judging from the table, seems meaningful. Swahili is an exclusively prefixing language but verbs tend to end in *-a* (whose status as a morpheme is the linguistic sense can be doubted), whereas Swedish is suffixing, although some prefixes are or were productive in word-formation.

Language	Corpus	Tokens	$ W $	$ S_W $	$ S'_W $
Maori	(The British & Foreign Bible Society, 1996)	1101665	8354	23007	78
English	(King James, 1977)	917634	12999	39845	63
Turkish	(American Bible Society, 1988)	574592	56881	175937	122

Table 7: Figures for different languages on the effects on the size of the suffix list after purging.

<i>-larına</i>	71645.4	<i>-adılar</i>	16587.9
<i>-larından</i>	47941.9	<i>-lerinden</i>	15201.1
<i>-lerinin</i>	43917.3	<i>-nden</i>	14082.2
<i>-lerden</i>	36294.0	<i>-sinin</i>	13493.9
<i>-inden</i>	35258.2	<i>-nin</i>	12340.9
<i>-iyorlardı</i>	28716.2	<i>-yorsunuz</i>	12135.0
<i>-arak</i>	27774.1	<i>-larla</i>	12069.7
<i>-iyorsunuz</i>	25403.1	<i>-en</i>	11513.5
<i>-inin</i>	25045.5	<i>-ten</i>	11424.0
<i>-dılar</i>	20718.7	<i>-sınız</i>	11043.0
<i>-lere</i>	20718.2	<i>-madılar</i>	10958.9
<i>-ip</i>	20431.2	<i>-lardan</i>	10428.1
<i>-dan</i>	19468.4	<i>-sınız</i>	10391.1
<i>-ndan</i>	18556.3	<i>-...</i>	<i>...</i>
<i>-ından</i>	18226.3	<i>-ist</i>	0.0
<i>-yorlardı</i>	18097.1	<i>-iy</i>	0.0
<i>-acaksınız</i>	16751.1	<i>-yo</i>	0.0

Table 6: Top 30 and bottom 3 extracted suffixes for Turkish. Most of these are really compounds of two suffixes, showing that some adaptation to multi-layer suffixing languages is appropriate.

A full discussion of further aspects such as a more informed segmentation of words, peeling of multiple suffix layers and purging of unwanted affixes requires, is beyond the scope of this paper.

5 Related Work

For reasons of space we cannot cite and comment every relevant paper even in the narrow view of highly unsupervised extraction of affixes from raw corpus data, but we will cite enough to cover each line of research. The vast fields of word segmentation for speech recognition or for languages which do not mark word boundaries will not be covered. In our view, segmentation into lexical units is a different problem than that of affix extraction since the frequencies of lexical items are different, i.e occur much more sparsely. Results from this area which

Swedish		English		Swahili	
<i>för-</i>	0.097	<i>-ed</i>	0.132	<i>-a</i>	0.100
<i>-en</i>	0.086	<i>-eth</i>	0.109	<i>wa-</i>	0.095
<i>-na</i>	0.036	<i>-iah</i>	0.099	<i>ali-</i>	0.065
<i>-ade</i>	0.035	<i>-ly</i>	0.090	<i>nita-</i>	0.059
<i>-a</i>	0.034	<i>-ings</i>	0.068	<i>aka-</i>	0.049
<i>-ar</i>	0.033	<i>-ing</i>	0.062	<i>ni-</i>	0.046
<i>-er</i>	0.033	<i>-ity</i>	0.059	<i>ku-</i>	0.044
<i>-as</i>	0.032	<i>-edst</i>	0.058	<i>ata-</i>	0.042
<i>-s</i>	0.031	<i>-ites</i>	0.046	<i>ha-</i>	0.032
<i>-de</i>	0.031	<i>-s'</i>	0.036	<i>a-</i>	0.031
...

Table 8: Comparative figures for prefix vs. suffix detection.

have been carried over or overlap with affix detection will however be taken into account. A lot of the papers cited have a wider scope and are still useful even though they are criticized here for having a non-optimal affix detection component.

Many authors trace their approaches back to two early papers by Zellig Harris (Harris, 1955; Harris, 1970) which count *letter successor varieties*. The basic procedure is to ask how many different phonemes occur (in various utterances e.g a corpus) after the first n phonemes of some test utterance and predict that segmentation(s) occur where the number of successors reaches a peak. For example, if we have *play, played, playing, player, players, playground* and we wish to test where to segment *plays*, the successor count for the prefix *pla* would be 1 because only *y* occurs after whereas the number of successors of *play* peak at three (i.e $\{e, i, g\}$). Although the heuristic has had some success it was shown (in various interpretations) as early as (Hafer and Weiss, 1974) that it is not really sound – even for English. A slightly better method is to compile a set of words into a *trie* and predict boundaries at nodes with high activity (e.g (Johnson and Martin, 2003; Schone

and Jurafsky, 2001; Kazakov and Manandhar, 2001) and earlier papers by the same authors), but this not sound either as non-morphemic short common character sequences also show significant branching.

The algorithm in this paper is differs significantly from the Harris-inspired varieties. First, we do not record the number of phonemes/character of a given prefix/suffix but their frequency distribution. In the example above, that would be the distribution $\{ e:3, i:1, g:1 \}$ rather than a uniform three-member set. Secondly, segmentation of a given word is not the immediate objective and what amounts to identification of the end of a lexical (thus generally low-frequency) item is not within the direct reach of the model. Thirdly, and most importantly, the algorithm in this paper looks at the *relative drop* of the frequency curve not at peaks in absolute frequency.

A different approach, sometimes used in complement of other sources of information, is to select *aligned pairs* (or sets) of strings that share a long character sequence (work includes (Jacquemin, 1997; Yarowsky and Wicentowski, 2000; Baroni et al., 2002; Clark, 2001)). A notable advantage is that one is not restricted to concatenative morphology.

Many publications (Ćavar et al., 2004; Brent et al., 1995; Goldsmith et al., 2001; Déjean, 1998; Snover et al., 2002; Argamon et al., 2004; Goldsmith, 2001; Creutz and Lagus, 2005; Neuvel and Fulop, 2002; Baroni, 2003; Gaussier, 1999; Sharma et al., 2002; Wicentowski, 2002; Oliver, 2004), and various other works by the same authors, describe strategies that use frequencies, probabilities, and optimization criteria, often Minimum Description Length (MDL), in various combinations. So far, all these are unsatisfactory on two main accounts; on the theoretical side, they still owe an explanation of why compression or MDL should give birth to segmentations coinciding with morphemes as linguistically defined. On the experimental side, thresholds, supervised/developed parameters and selective input still cloud the success of reported results, which, in any case, aren't wide enough to sustain some too rash language independence claims.

To be more specific, some MDL approaches aim to minimize the description of the set of words in the input corpus, some to describe all tokens in the corpus, but, none aims to minimize, what one would otherwise expect, the set of possible words

in the language. More importantly, none of the reviewed works allow any variation in the description language ("model") during the minimization search. Therefore they should be more properly labeled "weighting schemes" and it's an open question whether their yields correspond to linguistic analysis. Given an input corpus and a traditional linguistic analysis, it is trivial to show that it is possible to decrease description length (according to the given schemes) by stepping away from linguistic analysis. Moreover, various forms of codebook compression, such as Lempel-Ziv compression, yield shorter description but without any known linguistic relevance at all. What is clear, however, apart from whether it is theoretically motivated, is that MDL approaches are *useful*.

A systematic test of segmentation algorithms over many different types of languages has yet to be published. For three reasons, it will not be undertaken here either. First, as e.g already Manning (1998) notes for sandhi phenomena, it is far from clear what the gold standard should be (even though we may agree or disagree to disagree on some familiar European languages). Secondly, segmentation algorithms may have different purposes and it might not make good sense to study segmentation in isolation from induction of paradigms. Lastly, and most importantly, all of the reviewed techniques (Wicentowski, 2004; Wicentowski, 2002; Snover et al., 2002; Baroni et al., 2002; Andreev, 1965; Ćavar et al., 2004; Snover and Brent, 2003; Snover and Brent, 2001; Snover, 2002; Schone and Jurafsky, 2001; Jacquemin, 1997; Goldsmith and Hu, 2004; Sharma et al., 2002; Clark, 2001; Kazakov and Manandhar, 1998; Déjean, 1998; Oliver, 2004; Creutz and Lagus, 2002; Creutz and Lagus, 2004; Hirsimäki et al., 2003; Creutz and Lagus, 2005; Argamon et al., 2004; Gaussier, 1999; Lehmann, 1973; Langer, 1991; Flenner, 1995; Klenk and Langer, 1989; Goldsmith, 2001; Goldsmith, 2000; Hu et al., 2005b; Hu et al., 2005a; Brent et al., 1995), as they are described, have threshold-parameters of some sort, explicitly claim **not** to work well for an open set of languages, or require noise-free all-form input (Albright, 2002; Manning, 1998; Borin, 1991). Therefore it is not possible to even design a fair test.

In any event, we wish to appeal to the merits of developing a theory in parallel with experimentation

– as opposed to only ad hoc result chasing. If we have a theory and we don't get the results we want, we may scrutinize the assumptions behind the theory in order to modify or reject it (understanding why we did so). Without a theory there's no telling what to do or how to interpret intermediate numbers in a long series of calculations.

6 Conclusion

We have presented a new theory of affixation and a parameter-less efficient algorithm for collecting affixes from raw corpus data of an arbitrary language. Depending on one's purposes with it, a cut-off point for the collected list is still missing, or at least, we do not consider that matter here. The results are very promising and competitive but at present we lack formal evaluation in this respect. Future directions also include a more specialized look into the relation between affix-segmentation and paradigmatic variation and further exploits into layered morphology.

7 Acknowledgements

The author has benefited much from discussions with Bengt Nordström. The author is also grateful to Bob Carpenter for pointing out a grave technical error in an earlier version of this paper.

References

- Adam C. Albright. 2002. *The Identification of Bases in Morphological Paradigms*. Ph.D. thesis, University of California at Los Angeles.
- American Bible Society. 1988. *Turkish Bible*. American Bible Society, Tulsa, Oklahoma.
- American Bible Society. 1999. *Bib La*. American Bible Society. This edition from 2003.
- Nikolai Dmitrievich Andreev, editor. 1965. *Statistiko-kombinatornoe modelirovanie iazykov*. Akademia Nauk SSSR, Moskva.
- Shlomo Argamon, Navot Akiva, Amihod Amit, and Oren Kapah. 2004. Efficient unsupervised recursive word segmentation using minimum description length. In *COLING-04, 22-29 August 2004, Geneva, Switzerland*.
- Marco Baroni, Johannes Matiassek, and Harald Trost. 2002. Unsupervised discovery of morphologically related words based on orthographic and semantic similarity. In *Proceedings of the Workshop on Morphological and Phonological Learning of ACL/SIGPHON-2002*, pages 48–57.
- Marco Baroni. 2003. Distribution-driven morpheme discovery: A computational/experimental study. *Yearbook of Morphology*, pages 213–248.
- Winifred Bauer, William Parker, and Te Kareongawai Evans. 1993. *Maori*. Descriptive Grammars. Routledge, London & New York.
- Jerker Blomqvist and Poul Ole Jastrup. 1998. *Grekisk Grammatik: Graesk grammatik*. Akademisk Forlag, København, 2 edition.
- Lars Borin. 1991. *The Automatic Induction of Morphological Regularities*. Ph.D. thesis, University of Uppsala.
- Michael R. Brent, S. Murthy, and A. Lundberg. 1995. Discovering morphemic suffixes: A case study in minimum description length induction. In *Fifth International Workshop on Artificial Intelligence and Statistics, Ft. Lauderdale, Florida*.
- British and Foreign Bible Society. 1953. *Maandiko matakatifu ya Mungu yaitwaya Biblia, yaani Agano la kale na Agano jipya, katika lugha ya Kiswahili*. British and Foreign Bible Society, London, England.
- Damir Čavar, Joshua Herring, Toshikazu Ikuta, Paul Rodrigues, and Giancarlo Schrementi. 2004. On induction of morphology grammars and its role in bootstrapping. In Gerhard Jäger, Paola Monachesi, Gerald Penn, and Shuly Wintner, editors, *Proceedings of Formal Grammar 2004*, pages 47–62.
- Alexander Clark. 2001. Learning morphology with pair hidden markov models. In *ACL (Companion Volume)*, pages 55–60.
- Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the 6th Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON), Philadelphia, July 2002*, pages 21–30. Association for Computational Linguistics.
- Mathias Creutz and Krista Lagus. 2004. Induction of a simple morphology for highly-inflecting languages. In *Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, pages 43–51. Barcelona.
- Mathias Creutz and Krista Lagus. 2005. Unsupervised morpheme segmentation and morphology induction from text corpora using morfessor 1.0. Technical report, Publications in Computer and Information Science, Report A81, Helsinki University of Technology, March.

- Mathias Creutz and Krister Lindén. 2004. Morpheme segmentation gold standards for Finnish and English. publications in computer and information science, report a77, Helsinki University of Technology. Technical report, Publications in Computer and Information Science, Report A77, Helsinki University of Technology, October.
- Hervé Déjean. 1998. *Concepts et algorithmes pour la découverte des structures formelles des langues*. Ph.D. thesis, Université de Caen Basse Normandie.
- Matthew S. Dryer. 2005. Prefixing versus suffixing in inflectional morphology. In Bernard Comrie, Matthew S. Dryer, David Gil, and Martin Haspelmath, editors, *World Atlas of Language Structures*, pages 110–113. Oxford University Press.
- Gudrun Flenner. 1995. Quantitative morphsegmentierung im spanischen auf phonologischer basis. *Sprache und Datenverarbeitung*, 19(2):63–78.
- Nelson W. Francis and Henry Kucera. 1964. Brown corpus. Department of Linguistics, Brown University, Providence, Rhode Island. 1 million words.
- Éric Gaussier. 1999. Unsupervised learning of derivational morphology from inflectional lexicons. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-1999)*. Association for Computational Linguistics, Philadelphia.
- John Goldsmith and Yu Hu. 2004. From signatures to finite state automata. Technical report TR-2005-05, Department of Computer Science, University of Chicago.
- John Goldsmith, Derrick Higgins, and Svetlana Soglasnova. 2001. Automatic language-specific stemming in information retrieval. In Carol Peters, editor, *Cross-Language Information Retrieval and Evaluation: Proceedings of the CLEF 2000 Workshop*, Lecture Notes in Computer Science, pages 273–283. Springer-Verlag, Berlin.
- John Goldsmith. 2000. Linguistica: An automatic morphological analyzer. In A. Okrent and J. Boyle, editors, *Proceedings from the Main Session of the Chicago Linguistic Society's thirty-sixth Meeting*.
- John Goldsmith. 2001. Unsupervised learning of the morphology of natural language. *Computational Linguistics*, 27(2):153–198.
- Margaret A. Hafer and Stephen F. Weiss. 1974. Word segmentation by letter successor varieties. *Information and Storage Retrieval*, 10:371–385.
- Zellig S. Harris. 1955. From phoneme to morpheme. *Language*, 31(2):190–222.
- Zellig S. Harris. 1970. Morpheme boundaries within words: Report on a computer test. In Zellig S. Harris, editor, *Papers in Structural and Transformational Linguistics*, volume 1 of *Formal Linguistics Series*, pages 68–77. D. Reidel, Dordrecht.
- Teemu Hirsimäki, Mathias Creutz, Vesa Siivola, and Mikko Kurimo. 2003. Unlimited vocabulary speech recognition based on morphs discovered in an unsupervised manner. In *Proceedings of Eurospeech 2003, Geneva*, pages 2293–2996. Geneva, Switzerland.
- Yu Hu, Irina Matveeva, John Goldsmith, and Colin Sprague. 2005a. Refining the SED heuristic for morpheme discovery: Another look at Swahili. In *Proceedings of the Workshop on Psychocomputational Models of Human Language Acquisition*, pages 28–35, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Yu Hu, Irina Matveeva, John Goldsmith, and Colin Sprague. 2005b. Using morphology and syntax together in unsupervised learning. In *Proceedings of the Workshop on Psychocomputational Models of Human Language Acquisition*, pages 20–27, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Christian Jacquemin. 1997. Guessing morphology from terms and corpora. In *Proceedings, 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '97)*, Philadelphia, PA.
- Howard Johnson and Joel Martin. 2003. Unsupervised learning of morphology for English and Inuktitut. In *HLT-NAACL 2003, Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, May 27 - June 1, Edmonton, Canada*, volume Companion Volume - Short papers.
- Dimitar Kazakov and Suresh Manandhar. 1998. A hybrid approach to word segmentation. In C. D. Page, editor, *Proceedings of the 8th International Workshop on Inductive Logic Programming (ILP-98) in Madison, Wisconsin, USA*, volume 1446 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlin.
- Dimitar Kazakov and Suresh Manandhar. 2001. Unsupervised learning of word segmentation rules with genetic algorithms and inductive logic programming. *Machine Learning*, 43:121–162.
- King James. 1977. *The Holy Bible, containing the Old and New Testaments and the Apocrypha in the authorized King James version*. Thomas Nelson, Nashville, New York.
- Ursula Klenk and Hagen Langer. 1989. Morphological segmentation without a lexicon. *Literary and Linguistic Computing*, 4(4):247–253.

- Peter Ladefoged. 2005. *Vowels and Consonants*. Blackwell, Oxford, 2 edition.
- Hagen Langer. 1991. *Ein automatisches Morphsegmentierungsverfahren für deutsche Wortformen*. Ph.D. thesis, Georg-August-Universität zu Göttingen.
- Claire Lefebvre. 2004. *Issues in the study of Pidgin and Creole languages*, volume 70 of *Studies in Language Companion Series*. John Benjamins, Amsterdam.
- Hubert Lehmann. 1973. *Linguistische Modellbildung und Methodologie*. Max Niemeyer Verlag, Tübingen. Pp. 71-76 and 88-93.
- Joanes Leizarraga. 1571. *Iesus Krist Gure Iaunaren Testamentu Berria*. Pierre Hautin, Inprimizale, Roxellan. NT only.
- Christopher D. Manning. 1998. The segmentation problem in morphology learning. In Jill Burstein and Claudia Leacock, editors, *Proceedings of the Joint Conference on New Methods in Language Processing and Computational Language Learning*, pages 299–305. Association for Computational Linguistics, Somerset, New Jersey.
- David G. Nash. 1980. *Topics in Warlpiri Grammar*. Ph.D. thesis, Massachusetts Institute of Technology.
- Sylvain Neuvel and Sean A. Fulop. 2002. Unsupervised learning of morphology without morphemes. In *Workshop on Morphological and Phonological Learning at Association for Computational Linguistics 40th Anniversary Meeting (ACL-02)*, July 6-12, pages 9–15. ACL Publications.
- A. Oliver. 2004. *Adquisició d'informació lèxica i morfosintàctica a partir de corpus sense anotar: aplicació al rus i al croat*. Ph.D. thesis, Universitat de Barcelona.
- Franz Rosenthal. 1995. *A grammar of biblical Aramaic*, volume 5 of *Porta linguarum Orientalium*. Harrassowitz, Wiesbaden, 6 edition.
- Patrick Schone and Daniel Jurafsky. 2001. Knowledge-free induction of inflectional morphologies. In *Proceedings of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, PA, 2001*.
- Utpal Sharma, Jugal Kalita, and Rajib Das. 2002. Unsupervised learning of morphology for building lexicon for a highly inflectional language. In *Proceedings of the 6th Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON)*, Philadelphia, July 2002, pages 1–10. Association for Computational Linguistics.
- Matthew G. Snover and Michael R. Brent. 2001. A bayesian model for morpheme and paradigm identification. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001)*, pages 482–490. Morgan Kaufmann Publishers.
- Matthew G. Snover and Michael R. Brent. 2003. A probabilistic model for learning concatenative morphology. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, pages 1513–1520. MIT Press, Cambridge, MA.
- Matthew G. Snover, Gaja E. Jarosz, and Michael R. Brent. 2002. Unsupervised learning of morphology using a novel directed search algorithm: Taking the first step. In *Workshop on Morphological and Phonological Learning at Association for Computational Linguistics 40th Anniversary Meeting (ACL-02)*, July 6-12. ACL Publications.
- Matthew G. Snover. 2002. An unsupervised knowledge free algorithm for the learning of morphology in natural languages. Master's thesis, Department of Computer Science, Washington University.
- Summer Institute of Linguistics. 2001. *Bible: selections in Warlpiri*. Document 0650 of the Aboriginal Studies Electronic Data Archive (ASEDA), AIATSIS (Australian Institute of Aboriginal and Torres Strait Islander Studies), Canberra. Translated in portions 1968–2001.
- Svenska Bibelsällskapet. 1917. *Gamla och Nya testamentet: de kanoniska böckerna*. Norstedt, Stockholm.
- The British & Foreign Bible Society. 1996. *Maori Bible*. The British & Foreign Bible Society, London, England.
- Anthony Traill. 1994. *A !Xõ Dictionary*, volume 9 of *Quellen zur Khoisan-Forschung/Research in Khoisan Studies*. Rüdiger Köppe Verlag, Köln.
- Richard Wicentowski. 2002. *Modeling and Learning Multilingual Inflectional Morphology in a Minimally Supervised Framework*. Ph.D. thesis, Johns Hopkins University, Baltimore, MD.
- Richard Wicentowski. 2004. Multilingual noise-robust supervised morphological analysis using the word-frame model. In *Proceedings of the ACL Special Interest Group on Computational Phonology (SIGPHON)*, pages 70–77.
- David Yarowsky and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000)*, pages 207–216.